

Aprendizaje de la programación a partir de los proyectos de grado

Hernández Yasnó, Bibiana*
Forero Másmela, Alejandra**
Flórez Saldaña, Jaime***

1. Objetivo

Caracterizar el proceso de aprendizaje de programación a partir de los proyectos de grado en una institución de educación superior en Cali.

2. Revisión teórica

La enseñanza de programación dentro de las carreras tecnológicas y profesionales en el área de ingeniería es parte elemental de la formación, y la motivación de conocer el proceso en que aprenden, aplican sus conocimientos y cómo se reflejan en el desarrollo de los proyectos de grado son el enfoque de esta investigación. Entendiendo que la programación es un proceso esquemático y ordenado para la resolución de problemas en el que se usan herramientas informáticas como editores y compiladores este se aplica desde la resolución de ecuaciones matemáticas, desarrollo de páginas web, creación de video juegos, control de procesos y monitoreo en una máquina, entre tantas otras. Con base a esto se pretende mediante la investigación hacer una aproximación a cuál es el enfoque que tiene el autor en el proyecto de grado, el uso de los conocimientos adquiridos en la carrera sobre programación (sus múltiples complejidades y estructuras) y los criterios que usan en la selección del tipo de lenguaje de programación.

El proceso de formación en carreras del área de ingeniería informática y electrónica en la temática programación es muy variado de acuerdo a la naturaleza del

programa con su perfil ocupacional; por tanto, se enfoca el aprendizaje desde tres referentes: el psico-social y el técnico. En lo psico-social se toman en cuenta dos aspectos: el primero es la autoconfianza, teniendo en cuenta que los autores de los proyectos de grado vivieron una formación académica no sólo de conocimientos sino que debieron afrontar sus realidades sociales y confrontarlas con sus capacidades empleando la práctica al rendimiento, situaciones que le exigieron determinación y desempeño (Bandura, 1986, citado por Wiedenbeck, LaBelle & Kain, 2004). A través del ejercicio durante la carrera académica son capaces de alcanzar destrezas y dominio del tema (Ausbel, 1986, citado por Trejos, 2015).

La representación cognitiva que el conocimiento va desarrollando en habilidades de abstracción y comprensión para desarrollar la programación, de acuerdo a la lógica en la concepción de diseño de programar y la agilidad para comprender las herramientas empleadas en el área, es la estrategia para la construcción de metodología en la enseñanza y aprendizaje activo que incorpora herramientas para aprender áreas como la programación generando un apoyo de retroalimentación y esquemas que permitan su comprensión, pero sobre todo adaptación del nuevo conocimiento (López, Hernández & Farran, 2011).

Para el segundo factor, se toman en cuentas los modelos mentales, enfoque que emplea la enseñanza de programación como una actividad cognitiva que requiere el desarrollo de representaciones abstractas y estructura lógica como el conocimiento sobre sintaxis y la semántica del entorno de programación que se aplica (Heggstad & Kanfer, 2005, citado por Arévalo, Muñoz & Gómez, 2018). La importancia en los modelos mentales son empleados en los

* Especialista en telecomunicaciones. Ingeniero en sistemas. Correo electrónico: bibiana.hernandez@aunarcali.edu.co

** Tecnólogo en mecatrónica. Correo electrónico: alejandra.forero@aunarcali.edu.co

*** Ingeniero en sistemas. Correo electrónico: jaimesaldana@aunarcali.edu.co

esquemas y planos de manera abstracta que se toman para tener la comprensión de la problemática, su sistematización y su posterior aplicación en la realidad (Señas & Moroni, 2002, citado por Pérez & Roig-Vila, 2015).

Se toma como tercer referente la algoritmia como recurso en la comprensión para abordar la solución de la problemática del programa empleando el análisis, la definición, la delimitación y su desarrollo en el entorno de cualquier lenguaje de programación que se acerque más a su aplicación. Para el cuarto referente, se consideran los paradigmas de programación aplicados de acuerdo al tipo el uso del lenguaje en el desarrollo del proyecto de grado. La complejidad de los programas que se desarrollan actualmente produce la necesidad de encaminar a los alumnos a utilizar técnicas efectivas de programación. Es importante enfatizar en el diseño previo y una estrategia para lograrlo es comenzar a enseñar programación utilizando los algoritmos como recursos esquemáticos para plasmar el modelo de la resolución de un problema (Levy, 1994, citado por Moroni & Señas, 2005).

La perspectiva teórica empleada para desarrollar la programación está dada en los llamados paradigmas. Ha sido la evolución de la tecnología lo que ha implicado cambios en la comprensión y desarrollo de los lenguajes de programación, entre los más empleados se encuentran el imperativo, el funcional y el orientado a objetos, también llamado POO. El paradigma imperativo está enfocado en asociar una serie de instrucciones y condiciones que son guardados en un espacio de memoria; estos pueden ser llamados entornos de programación de bajo nivel. Para el paradigma funcional, la creación de funciones e instrucciones, que pueden albergar elementos de entrada y salida, permite dividir el programa de forma que cada uno realice

una única función; también pueden encontrarse híbridos de lenguajes de programación con paradigma imperativos. En el caso del paradigma orientado a objetos, está enfocado en la creación de espacios u objetos empleando atributos y métodos desde la lógica; puede encontrarse de forma híbrida en lenguajes de programación con estructuras funcionales (Trejos, 2015).

Teniendo en cuenta esto se han planteado estrategias haciendo uso de la teoría del aprendizaje significativo el cual permite el descubrimiento y el uso de aplicaciones learning para la resolución de problemas heredados de las matemáticas y en las cuales la programación de computadores a través del paradigma funcional proporciona caminos lógicos y alcanzables (Trejos, 2016). Se hace muy útil en estos procesos mitigativos que los estudiantes puedan verificar que la implementación de los algoritmos y su posterior codificación en un lenguaje de programación se ajustan a los planteamientos teóricos y formales que se hicieron desde el momento en que se escogió (Trejos, 2016).

Por otro lado, para la construcción de sistemas de visualización de software se deben estudiar las capacidades humanas de captación, asimilación y nacionalización de la información para luego adaptar los recursos computacionales en un mayor beneficio de la comprensión de eventos. Los sistemas de animación bien diseñados, contemplando los aspectos de la percepción humana y redundan en una mayor efectividad en su aplicación (Pérez & Roig-Vila, 2015).

En síntesis, la experiencia piloto aporta evidencias en torno al potencial de la modalidad de taller para incluir TIC dentro de la formación práctica de futuros ingenieros, permitiendo el desarrollo de diferentes habilidades destacadas por las políticas universita-

rias vigentes. Así mismo, dentro de la gran variedad de TIC disponibles, la construcción de simulaciones con entornos de programación por bloques ha sido una elección acertada en cuanto ofrece a los estudiantes un primer contacto con la programación, colaborando en la generación de un piso tecnológico común entre aquellos con formación técnica y conocimientos previos y aquellos que no. (Monjelay y Rodríguez, 2016).

3. Resultados esperados

La presente propuesta estará enfocado en tres puntos claves, la primera, encontrar la pertinencia del proceso de aprendizaje con las temáticas tratadas en programación para el desarrollo del egresado aplicando la programación en sus diferentes paradigmas con la temática para el desarrollo del proyecto aplicando las competencias y herramientas que permiten su construcción. La segunda, evidenciar como la aplicación de nuevas tecnologías de la información y la comunicación en el aula permiten fortalecer o mejorar el proceso académico en las asignaturas de programación para promocionar mejoras en las competencias y habilidades que permita fortalecer el impacto en futuros proyectos de grado.

La tercera, es la relación del perfil ocupacional que se tiene en los programas de ingeniería informática y electrónica con las temáticas desarrolladas en los proyectos de grados en una institución universitaria del norte de Cali empleando la recolección de datos, la búsqueda y el análisis dentro de su desarrollo metodológico en la realización de los proyectos de grados.

4. Impacto social, económico y ambiental

En lo social, contar con una herramienta de procesamiento de datos que permita a la dirección del programa conocer las estructuras de aprendizaje que se emplean en programación y las temáticas frecuentes en los proyectos de grado y compararlo con los contenidos programáticos de las asignaturas ofrecidas para así verificar el perfil ocupacional en los programas de ingeniería informática y electrónica. A largo plazo, generar una base de datos de todos los programas académicos de una facultad que apoye la construcción de un sistema de información, la actualización de temáticas en los contenidos programáticos y mejora de las temáticas en el área de programación. En lo económico, esta propuesta de investigación permitirá tener un estudio para la toma de decisión sobre la realidad que tiene los proyectos de grado con respecto a la pertinencia de las áreas de formación que ofertan el perfil ocupacional del programa académico y que están apoyadas en la malla curricular.

Referencias Bibliográficas

- Arévalo, C.; Muñóz, E. & Gómez, J. (2018). El efecto de la autoeficacia y el trabajo colaborativo en estudiantes novatos de programación. *Investigación y ciencia de la universidad autónoma de Aguascalientes*, 74, 73-80.
- López, J.; Hernández, C. & Farran, Y. (2011). Una plataforma de evaluación automática con una metodología efectiva para la enseñanza/aprendizaje en programación de computadores. *Ingeniare. Revista chilena de ingeniería*, 19(2), 265-277.
- Monjelay, N. & Rodríguez, G. (2016). *Repensando la programación como formación práctica en Ingeniería: un estudio de caso en primer año*. Ingenia-

re. *Revista chilena de ingeniería*, 26(1), 172-183.

Moroni, N. & Señas P. (2005). *Estrategias para la enseñanza de la programación. Jornadas de educación en informática y TICS en Argentina. Universidad Nacional del Sur. Argentina.*

Pérez H., & Roig-Vila, R. (2015). *Entornos de programación no mediados simbólicamente para el desarrollo del pensamiento computacional. Una experiencia en la formación de profesores de Informática de la Universidad Central del Ecuador. Revista RED de educación a distancia*, 46(9),1-22.

Trejos, O. (2015). *Modelo metodológico para simplificar la comprensión de tres paradigmas de programación basándose en aprendizaje significativo. Ventana informática*, 33, 125-142.

Trejos, O. (2016). *Metodología y algoritmo para encontrar números perfectos en un rango determinado de números naturales usando programación funcional. Entre Ciencia e Ingeniería*, 11(21), 66-72.

Wiedenbeck, S.; LaBelle, D. & Kain, V. (2004). *Factors affecting course outcomes in introductory programming. En E. Dunican, & T. R. G. Green (Eds.), Proceedings of the Psychology of programming interest group* 16, 97-110.
