



2019 10 al 13 de septiembre - Cartagena de Indias, Colombia

RETOS EN LA FORMACIÓN
DE INGENIEROS EN LA
ERA DIGITAL

DISEÑO Y VALIDACIÓN DE UN INSTRUMENTO DE RECOLECCIÓN DE INFORMACIÓN Y AUTO-REPORTE REFERENTE A LAS ESTRATEGIAS DE APRENDIZAJE DE PROGRAMACIÓN DE COMPUTADORES

Stephanie Torres Jiménez, Jhon Jairo Ramírez Echeverry, Felipe Restrepo Calle, Fabio A. González

**Universidad Nacional de Colombia
Bogotá, Colombia**

Resumen

El cuestionario propuesto por Pintrich, Smith, García y McKeachie en el año 1991 denominado *Motivated Strategies for Learning Questionnaire (MSLQ)* es un instrumento de auto-reporte que permite recolectar información asociada a la motivación y a las estrategias de aprendizaje (cognitivas y metacognitivas) que utilizan los estudiantes. Debido a su alcance y a su gran acogida, el cuestionario ha sido adaptado, traducido y aplicado a estudiantes de diferentes lugares del mundo como Pakistán, Sudáfrica, Colombia, Uruguay, entre otros, con el fin de responder a diferentes circunstancias y necesidades de los estudiantes de cada país o idioma. Así mismo, se han realizado adaptaciones para considerar aspectos específicos de diferentes áreas del conocimiento como ingeniería y física. No obstante, se ha visto un bajo nivel de aplicabilidad e intervención del cuestionario en el estudio de las estrategias del aprendizaje de la programación de computadores. Esto se debe en gran parte a que el instrumento, por la época en la que se desarrolló, no considera aspectos que hoy son relevantes, como el uso de tecnologías de la información para el aprendizaje. Dado que el desarrollo de software y la resolución de problemas a través de la programación han tomado un lugar importante en la ciencia y en la sociedad, es necesario llevar a cabo el diseño y validación de un instrumento basado en el MSLQ, que permita medir y caracterizar las estrategias de aprendizaje utilizadas por los estudiantes de programación de computadores. Es así como este artículo presenta la propuesta de trabajo en el que se construirá un nuevo instrumento de auto-reporte para caracterizar las estrategias de aprendizaje de los estudiantes de programación de computadores.

Palabras clave: programación de computadores; aprendizaje autorregulado; instrumento de auto-reporte, MSLQ

Abstract

The questionnaire proposed by Pintrich, García y McKeachie in 1991 denominated Motivated Strategies for Learning Questionnaire (MSLQ) is a self-report instrument, which allows collect information associated to the motivation and learning strategies (cognitive and metacognitive) that are used by students. Due to its scope and great reception, the questionnaire has been adapted, translated and applied to students from different places of the world such as Pakistan, South Africa, Colombia and Uruguay, with the goal to answer different circumstances and needs of each country or language. In the same way, several adaptations have been done to consider specific aspects from different knowledge areas like Engineering and Physics. Nevertheless, a low level of applicability and intervention of the questionnaire has been seen in the study of learning strategies of computer programming. It is due to the instrument does not consider aspects that are relevant today, such as the use of Information Technologies for learning. Because software development and problem-solving tasks through programming have played an important role in science and society. It is necessary to carry out the design and validation of content of an instrument based on MSLQ. It must measure and characterize the learning strategies used by computer programming students. It allows to give recommendations about content subject related to programming and formulate learning strategies. In such manner, this paper presents the work proposal, which consists in the construction of a new instrument to characterize student learning strategies of computer programming students.

Keywords: computer programming; self-regulated learning, self-report instrument, MSLQ

1. Introducción

Programar se ha convertido en una de las habilidades más importantes en los últimos años, considerada por diferentes autores como un área cognitiva que requiere el dominio de diferentes habilidades, entre ellas, la resolución de problemas. La programación se puede definir como la escritura de una secuencia de instrucciones que son procesadas por un computador con el fin de resolver un problema, y que se compone por cuatro elementos: el dominio del problema, donde se establece el enunciado con sus restricciones; el dominio del lenguaje de programación, el cual define la sintaxis y la semántica; el dominio del paradigma de programación, que define el estilo de la construcción del programa; y por último, la orquestación de la solución (Juarez-Ramirez, et al, 2019). Para autores como Tek et al., (2018) y Malik et al.,(2019), programar es una de las tareas más difíciles que deben enfrentar los estudiantes de Ciencias de la Computación. Esta dificultad podría deberse a la falta de motivación y a un bajo nivel en la habilidad para resolver problemas, aspectos que son necesarios en la tarea de programación (Drini, 2018).

Diversas investigaciones sobre el aprendizaje de la programación de computadores han hecho uso de la teoría sobre la autorregulación para poder entender cómo se comporta un estudiante de programación durante su proceso de aprendizaje, con dos objetivos principalmente: predecir su

éxito en el curso y proponer mejoras en los contenidos y estrategias de enseñanza. Por otro lado, los instrumentos o escalas se convierten en la principal fuente de información para medir los aspectos de autorregulación que la literatura enuncia, entre ellos, las estrategias de aprendizaje.

Autores como Chen y Whitesel (2012) ven la necesidad de diseñar instrumentos capaces de evaluar las estrategias de aprendizaje que son utilizadas en una tarea y dominio particulares, como en el caso de la programación de computadores. En consecuencia, es necesario tener en cuenta que un gran número de las estrategias de aprendizaje consideradas en los instrumentos existentes no son comúnmente utilizadas por los estudiantes de programación de computadores y por lo tanto, aportan en baja medida en su aprendizaje. Así mismo, éstas tampoco consideran el uso de herramientas modernas, como los cursos virtuales, los blogs, las comunidades en línea, entre otras. De ahí surge la necesidad de construir un instrumento que considere las estrategias de aprendizaje actuales, lo que permitiría identificar cuáles son las más relevantes y utilizadas por los estudiantes. Este artículo plantea el marco de trabajo que será utilizado para obtener un nuevo instrumento válido, cuyo objetivo es estudiar las estrategias de aprendizaje que usan los estudiantes de programación, considerando la evolución de las estrategias de autorregulación originales con las actuales.

El resto de este artículo está estructurado de la siguiente manera: en la Sección 2 se revisan los conceptos más importantes sobre la definición de la autorregulación en el aprendizaje y sus formas de medirla. En la Sección 3 se detallan los trabajos realizados y los instrumentos utilizados en el área de investigación, dando especial atención al MSLQ (Motivated Strategies for Learning Questionnaire), debido a su extenso uso. La Sección 4 describe el marco de trabajo propuesto para el diseño y validación de contenido del instrumento. Finalmente, en la Sección 5 se concluye con los aspectos más importantes.

2. Antecedentes

2.1. Aprendizaje autorregulado

La autorregulación en el aprendizaje es el proceso donde el estudiante controla y regula su propio aprendizaje, a través del establecimiento de objetivos y del seguimiento de éstos. Este concepto no sólo comprende el conocimiento de una asignatura o habilidad, sino que tiene en cuenta tres aspectos importantes: la autoconciencia, la automotivación y las habilidades de comportamiento de los estudiantes para implementar dicho conocimiento de manera adecuada (Zimmerman, 2002). Zimmerman (1989) define las estrategias de aprendizaje como los procesos encaminados para la adquisición de información o habilidades para el aprendizaje. Entre las cuales se incluye la búsqueda, la organización y la transformación de la información, así como el ensayo o la repetición de conceptos claves. Así mismo, determina que todo estudiante autorregulado es aquel que utiliza una serie de estrategias de aprendizaje con el objetivo de alcanzar sus metas académicas, como calificaciones altas, valoración social u oportunidades de empleo posteriores a la graduación.

2.2. Instrumentos de auto-reporte

En la literatura se pueden encontrar principalmente dos tipos de instrumentos para medir las estrategias de aprendizaje autorregulado en los estudiantes (Rovers, et al., 2019): Instrumentos en

línea e instrumentos fuera de línea. El primer tipo se aplica durante la realización de una tarea; los métodos más utilizados son: las observaciones sistemáticas, los protocolos en voz alta (o más conocidos por su nombre en inglés *think-aloud protocols*) y los registros de actividades, que son recolectados principalmente por los sistemas de gestión del aprendizaje (LMS por sus siglas en inglés). El segundo tipo, fuera de línea, son los conocidos como instrumentos de auto-reporte, que se emplean antes o después de la ejecución de la tarea. Usualmente estos últimos constan de una estructura de cuestionario o escalas tipo Likert, compuesto por una serie de sentencias o ítems con el objetivo de que las personas determinen su nivel de acuerdo o desacuerdo con cada una de éstas (Jacobse, et al., 2012).

Los instrumentos de auto-reporte son los métodos tradicionales para medir las estrategias de aprendizaje autorregulado. Algunos de los más conocidos son: MSLQ (Motivated Strategies for Learning Questionnaire) (Pintrich, et al., 1993), LASSI (Learning and Study Strategies Inventory) (Zimmermann et al., 1988), MAI (Metacognitive Awareness Inventory) (Schraw y Dennison, 1994), SPQ, (Study Process Questionnaire) (Biggs, 1987), entre otros. Estos instrumentos son utilizados por numerosos investigadores porque presentan ventajas sobre otros métodos, como en la recolección y análisis de información y en su aplicación, ya que los estudiantes no son interrumpidos durante sus actividades de aprendizaje (Schellings y Van Hout-Wolters, 2011).

3. La autorregulación en la programación de computadores

3.1 Uso y diseño de instrumentos para medir la autorregulación en el aprendizaje de la programación de computadores

Los instrumentos de auto-reporte se han hecho muy populares en la práctica y en la investigación. Cabe resaltar que muchos de los trabajos encontrados se basan en el uso de instrumentos fuera de línea, y algunos de ellos se mencionan a continuación.

Se evidencia el extenso uso del MSLQ en el contexto de la programación de computadores, el cual es adaptado, o aplicado, en su totalidad o subescalas del mismo, como lo hicieron (Tsai, 2018) y (Bergin, et al., 2005). En el primer estudio, se adaptó la subescala de autoeficacia en un entorno de aprendizaje de VPL (*Visual Programming Language*), que tenía el objetivo de apoyar la enseñanza de conceptos básicos de programación. Como resultado de la investigación, el autor encontró que la autoeficacia desempeña un papel importante en el aprendizaje del concepto de la programación. En la segunda investigación se aplicó el cuestionario en su totalidad para conocer el nivel de incidencia de la autorregulación en el rendimiento académico. Cabe destacar que las dos investigaciones concluyeron que las estrategias motivacionales tenían un gran impacto sobre el desempeño de los estudiantes, mientras que las estrategias como la repetición de conceptos clave y la elaboración de ideas no eran muy utilizadas por los estudiantes. Dicha conclusión podría ser un indicio de que las estrategias de aprendizaje consideradas en el MSLQ no son las más apropiadas para aprender a programar o no son tenidas en cuenta por los estudiantes.

Así mismo, Castellanos, et al., (2017) y Ramírez, et al., (2018) aplicaron el MSLQ-Colombia (Ramírez, et al., 2016), que es una adaptación del MSLQ al idioma español, a diferentes grupos de estudiantes de un curso de programación de computadores, denominado Estructuras de Datos

impartido en la Universidad Nacional de Colombia. En la primera investigación resaltan el papel que desempeña el análisis de código fuente en el estudio de la motivación y las estrategias de aprendizaje. Por otro lado, en la segunda se evidenció que sólo las variables motivacionales tenían correlación en el desempeño académico, y únicamente se obtuvieron resultados significativos estadísticamente relacionados con los aspectos de motivación y no con las estrategias de aprendizaje.

Ramalingam y Wiedenbeck, (2005) proponen una escala de medición de autoeficacia en los estudiantes que están aprendiendo a programar en C++ con el paradigma orientado a objetos. Ésta se enfoca en tareas de programación como: diseñar, escribir, comprender, modificar y reutilizar programas. Su propósito principal era calcular el grado de motivación y persistencia del estudiante ante la adversidad en el aprendizaje. Los autores muestran que aunque existen otros factores como: la habilidad para aprender, la experiencia previa en matemáticas y computación y el uso de estrategias de aprendizaje, la autoeficacia parece tener un nivel alto de influencia sobre los procesos de aprendizaje.

3.2 Autorregulación en las estrategias de aprendizaje de la Programación de Computadores

Diferentes autores han intentado medir o caracterizar las estrategias de aprendizaje autorregulado en los estudiantes de programación de computadores. El trabajo realizado por Loksa y Ko (2016) es uno de los más novedosos, ya que proponen un marco de trabajo de autorregulación en el área de la computación, que tiene en cuenta aspectos como la metacognición, la planeación y la motivación. Está compuesto por cinco actividades que soportan la resolución de problemas mostradas:

- **Planeación:** el estudiante debe reflexionar acerca del siguiente paso cuando está resolviendo un problema.
- **Monitoreo del proceso:** el estudiante debe hacer un seguimiento acerca de cómo va la resolución del problema, preguntándose si la tarea aún está incompleta.
- **Monitoreo de la comprensión:** los estudiantes deben razonar sobre las limitaciones de su memoria al resolver el problema.
- **Reflexión sobre la cognición:** los estudiantes deben razonar sobre sus limitaciones de memoria cuando resuelven un problema.
- **Autoexplicación:** el estudiante debe ser capaz de auto explicarse acerca del proceso que le permitió llegar a una solución.

Otro estudio que también sugiere las escalas de autorregulación para el aprendizaje de programación es la revisión sistemática de literatura llevada a cabo por García, et al., (2018). Los autores redefinen las categorías de la taxonomía de estrategias de aprendizaje autorregulado propuesta por Zimmerman y Martínez-Pons (1986), ya que afirman que éstas no consideran las herramientas online para el aprendizaje en las Ciencias de la Computación, debido a la fecha en que se propuso dicha taxonomía. Para este trabajo, el elemento de valor lo dan las definiciones adaptadas sobre autorregulación que da García, et al. Por esta razón, a continuación, se mencionan sólo las que se consideraron pertinentes para el tema de estudio:

- **Autoevaluación:** el estudiante se autoevalúa para validar ejercicios de programación.
- **Organización y transformación:** los estudiantes diseñan planes antes de codificar sus programas. Los diagramas de flujo desempeñan un papel importante en esta tarea.
- **Establecimiento de objetivos y planeación:** el estudiante establece los objetivos del programa a desarrollar, así como el tiempo destinado para ello o para estudiar.
- **Búsqueda de información:** los estudiantes deciden consultar en sitios online para comprender mejor lo que no entiende. Los video tutoriales, blogs y comunidades en línea son los sitios más relevantes para esta actividad.
- **Registros y monitoreo:** esfuerzo del estudiante para guardar resultados o materiales de aprendizaje.
- **Auto-consecuencias:** los estudiantes establecen premios o castigos por el éxito o falla durante su proceso de aprendizaje. Aquí se consideran los ambientes competitivos donde utilicen estrategias como gamificación.
- **Ensayo y memorización:** se considera que el estudiante memorice conceptos a través de la práctica. Una estrategia es la demostración de cómo resolver un problema. Los tutoriales son los más afines a este concepto.
- **Búsqueda de ayuda:** se tiene en cuenta la ayuda de los compañeros, profesor y la obtenida de las comunidades como foros en línea.
- **Revisión de registros:** se considera hacer una revisión detallada del material de estudio como notas de clase, tutoriales, páginas web, cursos en línea, entre otros, con el objetivo de preparar las clases y futuros exámenes.

4. Propuesta del diseño del instrumento

La mayoría de los estudios realizados sobre la autorregulación han dado como resultado que las estrategias consideradas en los instrumentos como el MSLQ, son menos utilizadas por los estudiantes de programación de computadores, puesto que las empleadas podrían tener características particulares, como el uso de las tecnologías de la información. La Tabla I muestra algunas de las estrategias recopiladas por García et al. (2018) que no son consideradas en los instrumentos actuales.

Tabla I. Estrategias de aprendizaje en Programación de Computadores

Escala de autorregulación	Actividades
Autoevaluación	Procesos de revisión de código fuente Debug sobre los programas
Organización y transformación	Uso de diagramas de flujo
Búsqueda de información y ayuda	Búsqueda en blogs y comunidades en línea Búsqueda de asesoría de profesores
Auto-consecuencias	Incursión en ambientes competitivos, ejemplo: gamificación
Ensayo y memorización	Uso de video-tutoriales
Revisión de registros	Revisión de notas de clase, cursos en línea, tutoriales y manuales de referencia para preparar la clase y exámenes

Dadas las actividades anteriormente mencionadas, se propone el diseño y validación de contenido de un instrumento de auto-reporte que permita medir las estrategias de aprendizaje de los estudiantes de programación de computadores asociadas a dichas actividades. Para lo cual, planteamos el siguiente marco de trabajo mostrado en la Figura 1.

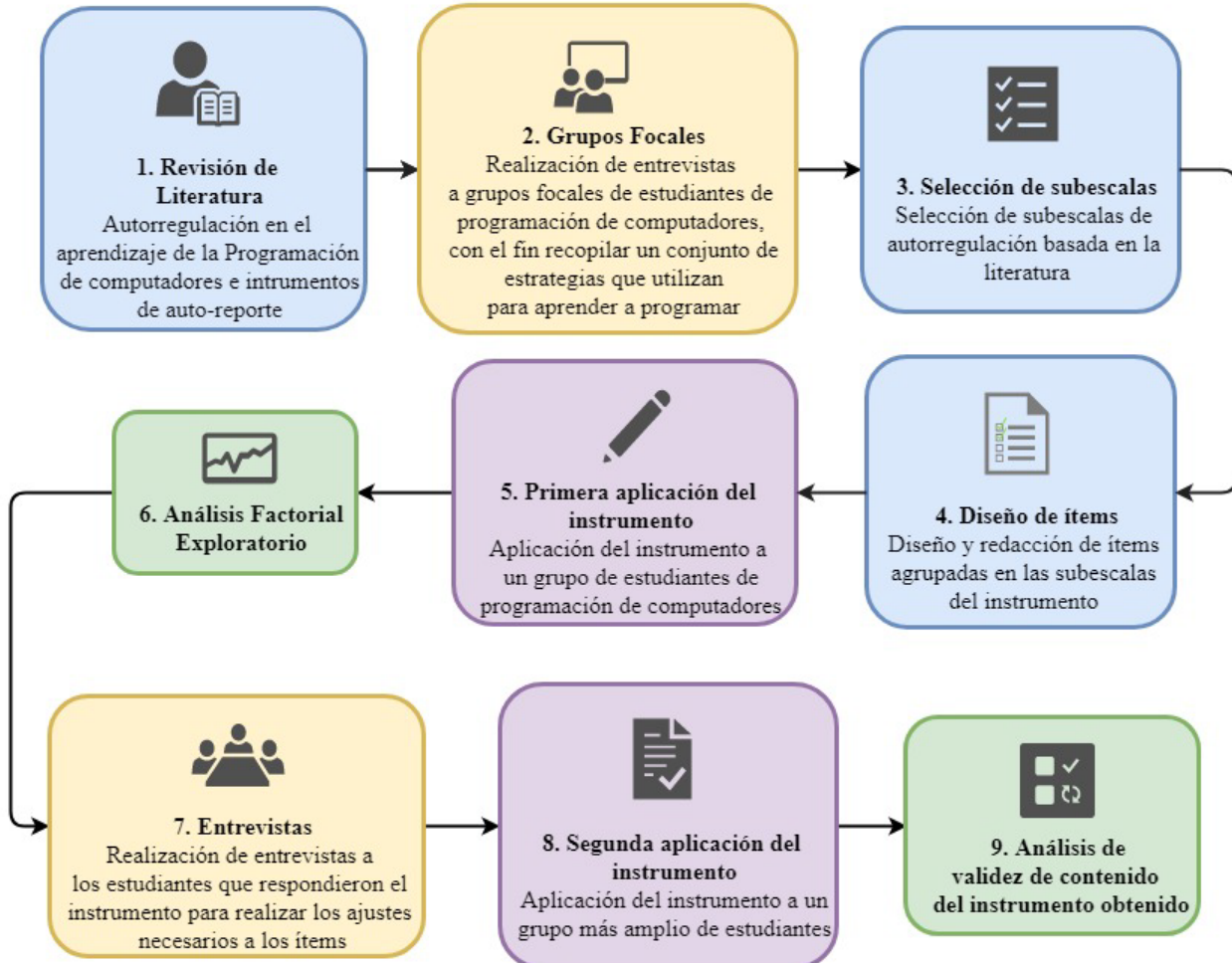


Figura 1. Marco de trabajo propuesto para el diseño del instrumento de auto-reporte

5. Conclusiones

Los instrumentos de autorreporte se han convertido en una fuente de información relevante en el estudio de la autorregulación de los estudiantes de programación de computadores, ya que, entre otras cosas, permiten descubrir correlaciones entre el desempeño académico con la motivación y el uso de estrategias de aprendizaje. Sin embargo, aún falta indagar este aspecto. Diversas investigaciones apuntan a la necesidad de construir y validar instrumentos específicos para determinar las estrategias de aprendizaje utilizadas en áreas de estudio específicas, como, por ejemplo, la programación de computadores. Esto debido en gran parte a los avances tanto en su aprendizaje como en su enseñanza. Dada esta necesidad, surgió la motivación de detallar la

propuesta del marco de trabajo para el diseño del instrumento como una investigación a corto plazo.

6. Referencias

- Bergin, S., Reilly, R., & Traynor, D. (2005). Examining the role of self-regulated learning on introductory programming performance. In *First International Workshop on Computing Education Research* (pp. 81–86).
- Castellanos, H., Restrepo-Calle, F., González, F. A., & Echeverry, J. J. R. (2017). Understanding the relationships between self-regulated learning and students source code in a computer programming course. In *Proceedings - Frontiers in Education Conference, FIE* (Vol. 2017-Oct, pp. 1–9). IEEE.
- Chen, C., & Whitesel, J. (2012). The Validity and Reliability Study of a Revised Motivated Strategy for Learning Questionnaire (MSLQ) for Assessing Computer Software Learning Strategies. *Int. Journal of E-Adoption*, 4(2), 28–51.
- Drini, M. (2018). Using new methodologies in teaching computer programming. In *ISEC-2018 - Proceedings of the 8th IEEE Integrated STEM Education Conference* (Vol. 2018-Jan, pp. 120–124).
- Garcia, R., Falkner, K., & Vivian, R. (2018). Systematic literature review: Self-Regulated Learning strategies using e-learning tools for Computer Science. *Computers and Education*, 123, 150–163.
- Garcia, A., & Ramirez, J. J. (2016). Adaptation and Validation of the Motivated Strategies for Learning Questionnaire -MSLQ- in Engineering Students in Colombia. *Int. Journal of Engineering Education*, 32(4), 1774–1787.
- Jacobse, A. E., & Harskamp, E. G. (2012). Towards efficient measurement of metacognition in mathematical problem solving. *Metacognition and Learning*, 7(2), 133–149.
- Juarez-Ramirez, R., Navarro, C. X., Tapia-Ibarra, V., Macias-Olvera, R., & Guerra-Garcia, C. (2019). What is Programming? Putting all Together - A Set of Skills Required. In *2018 6th International Conference in Software Engineering Research and Innovation (CONISOFT)* (pp. 11–20).
- Loksa, D., & Ko, A. J. (2016). The Role of Self-Regulation in Programming Problem Solving Process and Success. In *Proceedings of the 2016 ACM Conference on International Computing Education Research - ICER '16* (pp. 83–91).
- Malik, S. I., Mathew, R., Al-Nuaimi, R., Al-Sideiri, A., & Coldwell, J. (2019). Learning problem solving skills: Comparison of E-learning and M-learning in an introductory programming course. *Education and Information Technologies*.
- Pintrich, P. R., Smith, D. A. F., Garcia, T., & Mckeachie, W. J. (1993). Reliability and Predictive Validity of the Motivated Strategies for Learning Questionnaire (Mslq). *Educational and Psychological Measurement*, 53(3), 801–813.
- Ramalingam, V., & Wiedenbeck, S. (2005). Development and Validation of Scores on a Computer Programming Self-Efficacy Scale and Group Analyses of Novice Programmer Self-Efficacy. *Journal of Educational Computing Research*, 19(4), 367–381.

- Ramírez Echeverry, J. J., Rosales Castro, L. F., Restrepo Calle, F., & González, F. A. (2018). Self-regulated learning in a computer programming course. *IEEE Revista Iberoamericana de Tecnologías Del Aprendizaje*, 13(2), 75–83.
- Rovers, S. F. E., Clarebout, G., Savelberg, H. H., de Bruin, A. B. H., & van Merriënboer, J. J. G. (2019). Granularity matters: comparing different ways of measuring self-regulated learning. *Metacognition and Learning*, 1–19.
- Schellings, G., & Van Hout-Wolters, B. (2011). Measuring strategy use with self-report instruments: theoretical and empirical considerations. *Metacognition and Learning*, 6(2), 83–90.
- Schraw, G., & Dennison, R. S. (1994). Assessing metacognitive awareness. *Contemporary Educational Psychology*, 19, 460–475
- Tek, F. B., Benli, K. S., & Deveci, E. (2018). Implicit Theories and Self-Efficacy in an Introductory Programming Course. *IEEE Transactions on Education*, 61(3), 218–225.
- Tsai, C. Y. (2019). Improving students' understanding of basic programming concepts through visual programming language: The role of self-efficacy. *Computers in Human Behavior*, 95 (May 2018), 224–232.
- Weinstein, C. E., Zimmermann, S. A., & Palmer, D. R. (1988). Assessing Learning Strategies: The design and development of the LASSI. In *Learning and Study Strategies* (pp. 25–40).
- Zimmerman, B. (2002). Becoming a Self-Regulated Learner: An Overview. *Theory into Practice*, 41(2), 64–67.
- Zimmerman, B. (1989). A social cognitive view of self-regulated academic learning. *Journal of Educational Psychology*, 81(3), 329–339.
- Zimmerman, B., & Pons, M. M. (1986). Development of a Structured Interview for Assessing Student Use of Self-Regulated Learning Strategies. *American Educational Research Journal*.

Sobre los autores

- **Stephanie Torres Jiménez**, Ingeniera de Sistemas y Computación, Estudiante de Maestría en Ingeniería de Sistemas y Computación de la Universidad Nacional de Colombia. sttorresji@unal.edu.co.
- **Jhon Jairo Ramírez Echeverry**, Ph.D., Profesor asociado, Ingeniería Eléctrica y Electrónica, Universidad Nacional de Colombia, jjramireze@unal.edu.co.
- **Felipe Restrepo Calle**, Ph.D., Profesor asociado, Ingeniería de Sistemas e Industrial, Universidad Nacional de Colombia. ferestrepoca@unal.edu.co.
- **Fabio Augusto González Osorio**, Ph.D., Profesor Titular, Ingeniería de Sistemas e Industrial, Universidad Nacional de Colombia. fagonzalezo@unal.edu.co.

Los puntos de vista expresados en este artículo no reflejan necesariamente la opinión de la Asociación Colombiana de Facultades de Ingeniería.

Copyright © 2019 Asociación Colombiana de Facultades de Ingeniería (ACOFI)