

# Incorporación de un *software* educativo como estrategia pedagógica para la enseñanza de la lógica computacional

---

Incorporation of an educational software as a pedagogical strategy for the teaching of computational logic

[Innovación]

**Edwin Quintero Monsalve\***

Secretaría de Educación del Valle del Cauca, Colombia

✉ [edwquinmon@gmail.com](mailto:edwquinmon@gmail.com)

 <https://orcid.org/0000-0002-2998-8759>

**Alexandra María Silva Monsalve\*\***

Universidad Santo Tomás, Colombia

✉ [alexandrasilva@ustadistancia.edu.co](mailto:alexandrasilva@ustadistancia.edu.co)

 <https://orcid.org/0000-0001-7554-0237>

Recibido: 10 de agosto del 2021

Aceptado: 18 de septiembre del 2021

Citar como:

Quintero Monsalve, E. y Silva Monsalve, A. M. (2021). Incorporación de un *software* educativo como estrategia pedagógica para la enseñanza de la lógica computacional. *CITAS*, 7(1).

<https://doi.org/10.15332/24224529.6846>



---

\* Magister en Gestión de la Tecnología Educativa y docente adscrito a la Secretaría de Educación del Valle del Cauca.

\*\* Ph. D. en Educación y docente investigadora de la Universidad Santo Tomás.

## Resumen

Las dificultades en el diseño y construcción de algoritmos computacionales en los estudiantes y su efecto en el bajo rendimiento académico originaron el proyecto de investigación que dio vida a este artículo. En la investigación, se propuso como objetivo la implementación de una estrategia pedagógica apoyada en el *Software* educativo *App Inventor* para el mejoramiento de la lógica computacional en la implementación de algoritmos con los estudiantes del Técnico en Análisis y Desarrollo de Sistemas en Tulua, Valle del Cauca. Metodológicamente, se desarrolló una investigación de tipo cuantitativo descriptivo. Inicialmente, se aplicó una prueba diagnóstica inicial con el fin de recolectar datos y realizar un análisis; posteriormente, se diseñaron sesiones de trabajo con el apoyo del *software App Inventor*, para luego, finalmente, aplicar una prueba final. La propuesta pedagógica se orientó desde los principios del constructivismo, generando estrategias de aprendizaje significativo. Los resultados reflejaron un aporte satisfactorio al proceso de enseñanza-aprendizaje de los estudiantes, lo que les permitió construir su propio conocimiento y alcanzar un aprendizaje significativo. Por otra parte, el desempeño de los estudiantes en las clases evidenció su interés por el aprendizaje y la actitud crítica frente a la resolución de las diferentes actividades propuestas.

**Palabras claves:** algoritmo, aprendizaje, aprendizaje activo, enseñanza, programación informática, programa de ordenador, *software* didáctico.

## Abstract

The difficulties in the design and construction of computational algorithms in students and their effect on poor academic performance originated the research project that led to this article. The objective of this research project was the implementation of a pedagogical strategy supported by the educational *software App Inventor* for the improvement of computational logic in the implementation of algorithms with the Technician in Systems Analysis and Development students in Tulua, Valle del Cauca. Methodologically, a descriptive quantitative research was developed. Firstly, an initial diagnostic test was applied in order to collect data and perform an analysis; afterwards, work sessions were designed with the support of the *software App Inventor*, and then, a final test was applied. The results identified a satisfactory contribution to the teaching-learning process of the students, which allowed them to build their own knowledge and achieve significant learning. On the other hand, the performance of the students in the classes showed their interest in learning and critical attitude towards the resolution of the different activities proposed.

**Keywords:** algorithms, learning, active learning, teaching, computer programming, computer *software*, didactic software.

## Introducción

El diseño curricular del programa Técnico en Programación de *Software* en sus competencias y resultados de aprendizaje incorpora la enseñanza de algoritmos computacionales. El anterior

programa es articulado en la cadena de formación de estudiantes de grado décimo y once, se les imparte formación en dos áreas técnicas denominadas análisis de sistemas de información (ASI) y desarrollo de sistemas de información (DSI). En el transcurso del segundo periodo académico (2018), de los diecinueve (19) estudiantes de grado décimo, el 74 % de los evidenciaron desempeños básicos y bajos en las competencias procedimentales y solamente el 26 % evidenciaron desempeños altos y avanzados. De esta manera, los niveles de desempeño se establecieron de acuerdo con la siguiente escala de valoración: de 0,0 a 3,4, desempeño bajo; de 3,5 a 4,2, desempeño básico; de 4,3 a 4,7, desempeño alto y de 4,8 a 5,0, desempeño superior.

Los resultados sobre los niveles de desempeño de los estudiantes (19) de grado décimo muestran que seis (6) estudiantes obtuvieron un nivel desempeño bajo, equivalente al 32 %; que ocho (8) estudiantes obtuvieron un nivel de desempeño básico, equivalente al 42 %; que tres (3) estudiantes, un nivel de desempeño alto, equivalente al 16 %, y que dos (2) estudiantes, un nivel de desempeño superior, equivalente al 10 %. De acuerdo con los resultados, se puede afirmar que el número de estudiante que obtuvieron un desempeño bajo y básico fueron catorce (14) estudiantes, equivalentes al 74 %; esto indica que más de la mitad de los estudiantes de grado décimo presentaron dificultad en la apropiación de los conceptos y temáticas relacionados con el diseño y construcción de algoritmos computacionales. Por otra parte, revisando los desempeños de los estudiantes en el 2017, de los treinta y cinco (35) estudiantes de grado décimo, treinta y dos (32) estudiantes, equivalentes al 91.4 %, lograron un desempeño básico. Esto evidencia que la gran mayoría de estudiantes de este grado presentaron dificultades relacionadas con los algoritmos computacionales; en este sentido, surge un reto para la labor docente: ¿Cómo mejorar la lógica computacional en la implementación de algoritmos con los estudiantes?, justamente, teniendo en cuenta que esta habilidad es fundamental para el desarrollo de la competencia en la formulación y planteamiento de problemas a nivel computacional.

### **Algoritmos computacionales**

Se definen como un método procedimental para la resolución de un problema, una serie de datos precisos, definidos y finitos. Un algoritmo es una secuencia de pasos lógicos y ordenados con los cuales se da solución a un problema en específico (Colombia Aprende, 2018). Los algoritmos son la base de la programación de *softwares*, teniendo en cuenta que para el manejo y operación de un lenguaje de programación es importante desarrollar un conocimiento lógico-procedimental, que se adquiera apropiando y desarrollando las competencias del saber y el “saber hacer”, relacionados con los algoritmos computacionales. De esta manera, los estudiantes de la modalidad de la media técnica, a partir del grado décimo y hasta el grado once, comienzan una etapa de formación técnica denominada Técnico en Programación de *Software*. Para adelantar las competencias de este programa, es importante que los estudiantes desarrollen habilidades o competencias en algoritmos y programación. Los estudiantes aprenden a realizar un análisis de un problema y a diseñar su solución de forma ordenada y secuencial, lo que los lleva a un aprendizaje significativo en el que se produce una interacción entre los conocimientos más relevantes, de sus estructuras cognitivas, y los nuevos conocimientos, de tal manera que estos adquieran un significado y se integren a su estructura cognitiva.

## ***Los algoritmos computacionales y la resolución de problemas***

Todas las actividades y tareas de nuestro diario vivir son procesos rutinarios en los que sin darnos cuenta ejecutamos un algoritmo; una secuencia de pasos para realizar una actividad: levantarnos, bañarnos, cepillarnos los dientes, hacer una actividad deportiva, hacer las tareas del colegio, etc. Por costumbre, todos los días se hacen de forma mecánica, pero no de manera lógica. No obstante, se adolece de un pensamiento lógico-matemático, para los estudiantes es fundamental porque permite fortalecer las habilidades procedimentales y lógicas permitiendo analizar un problema a través de una solución algorítmica, mediante la cual llegan a la solución de un determinado problema (Ausubel, 1983). Los procesos de enseñanza-aprendizaje de los algoritmos computacionales ayudarán a los estudiantes a fortalecer su pensamiento lógico-matemático y, además, les permitirán entender y analizar diferentes tipos de problemáticas, que, a través de una secuencia lógica y ordenada de pasos, los puede llevar a la solución. Esto contribuye al fortalecimiento de habilidades cognitivas y procedimentales de los estudiantes relacionados con la lógico-matemática (Botero y Trefftz, 2015).

### **Fundamentos teóricos**

Las bases conceptuales que sustentan el desarrollo teórico de la investigación provienen del constructivismo y el aprendizaje significativo, teorías de aprendizaje que permiten evidenciar la importancia de la apropiación cognitiva por parte del estudiante.

#### ***Aprendizaje significativo***

El aprendizaje de la algoritmia computacional es un proceso de adquisición de conocimientos estructurados, lógicos y secuenciales, en los que, cada día, el estudiante aprende nuevas teorías y plantea nuevas formas en pro de darle solución a un problema específico. A medida que el estudiante analiza los enunciados de un problema en un nivel algorítmico, va construyendo un aprendizaje significativo. Según Piaget (2000) se puede dar a través de tres posibilidades: la asimilación, que se consolida por medio de una base sólida de conocimientos e ideas previas; la acomodación, donde se apropian los nuevos conocimientos sin ningún proceso de reflexión y adaptación previa, y la adaptación inteligente, que constituye el equilibrio entre los conocimientos previos y los nuevos, buscando que el conocimiento, y su estructura, responda óptimamente a las necesidades internas y externas (Rivas, 2008). Por otra parte, Ausubel propuso originalmente el concepto de aprendizaje significativo de la siguiente forma: “el aprendizaje ocurre cuando el material se presenta en su forma final y se relaciona con los conocimientos anteriores de los alumnos” (Romero, 2009, p. 2). Así, el aprendizaje significativo ocurre cuando el estudiante relaciona y comprende la información nueva para él y la relaciona con el conocimiento previo que tiene; reajustando y reestructurando ambas informaciones en este proceso y, así, logrando actualizar el conocimiento de conceptos y teorías que él había apropiado previamente con un nuevo conocimiento. El cuál aprendió y apropió, para, así, construir aprendizaje sólido y válido.

En el aprendizaje significativo, el aprendiz debe hacer uso de los significados que ya internalizó para poder captar los significados de los materiales educativos, en este caso, está progresivamente diferenciando su estructura cognitiva y haciendo una reconciliación integradora que busca

identificar semejanzas y diferencias y reorganizar su conocimiento (Moreira, 2005). Así mismo, el aprendizaje por descubrimiento guiado parte de unos principios metodológicos activos e investigadores. El estudiante desarrollará sus aprendizajes a partir de sus experiencias, pero siempre guiado y orientado por un docente, quien orientará los distintos tipos aprendizaje, cognitivos, procedimentales o actitudinales, etc. (Coloma y Tafur, 1999). Aunado a lo anterior, el fortalecimiento de las competencias matemáticas desde edades tempranas se convertirá en una fortaleza cognitiva para el estudiante (Palma y Sarmiento, 2015). En un proyecto adelantado por Taborda y Medina (2012), estos autores relacionan el fortalecimiento de las habilidades matemáticas y la lógica computacional en niños y niñas.

### ***El constructivismo***

El proceso de enseñanza y aprendizaje de algoritmos computacionales es un ejemplo tangible del constructivismo, en este proceso, el docente facilita las herramientas didácticas, pedagógicas y tecnológicas para que el estudiante visualice, lea y manipule, con el fin de construir su propio conocimiento. Hoy en día, las tecnologías de la información y la comunicación (TIC) se están convirtiendo en una herramienta didáctica muy utilizada en los ambientes de aprendizaje, porque hacen que los conceptos sean más visuales y fáciles de apropiar, cobren sentido y sean menos memorísticos y más didácticos. De esta manera, el constructivismo sustentó la estrategia pedagógica propuesta en esta investigación, ya que, como teoría del conocimiento, apunta a que la persona no es una obra que genera el ambiente, sino una construcción propia, que se construye a diario como consecuencia de la relación que esta tiene con su entorno. Aunque existen varias corrientes constructivistas, hay una idea común en todas: el aprendizaje es un proceso activo, diferente para cada individuo y construido de manera individual y social a partir de los conocimientos que se tienen sobre el mundo. A partir de lo anterior, los principios generales del constructivismo se centran en la interacción del hombre con el entorno, los aprendizajes previos como requerimiento del conocimiento a construir, la significación que se da a la información nueva y la relación entre el conocimiento y la realidad (Araya et al., 2007).

Piaget afirma que el proceso de aprendizaje es una práctica basada en el descubrimiento, en la que la motivación para aprender es la clave de los nuevos aprendizajes. Dicha motivación crea un equilibrio entre la organización, integración y transformación de la información que recibe el individuo, es decir, entre esquemas, asimilación y acomodación (Serrano y Pons, 2011). Por otra parte, Ausubel define el aprendizaje como una actividad significativa para la persona que aprende, asociándolo a la presencia de relaciones entre el nuevo conocimiento y los conocimientos previos adquiridos por esta (Carretero, 2009). El constructivismo aporta herramientas y teorías a los objetivos de la educación contemporánea, los cuales buscan formar estudiantes autónomos, críticos y librepensadores, quienes construyen su propio conocimiento (Tünnermann, 2011). En este sentido, las TIC juegan un papel importante al momento de crear, compartir y transformar el conocimiento, ya que se constituyeron en piezas fundamentales de la construcción de la estrategia pedagógica para la enseñanza y el aprendizaje, en este artículo, mediada por el *software* educativo *App Inventor*.

## ***Incorporación de las TIC en la enseñanza y aprendizaje***

Es inminente la incursión de las TIC en los procesos de enseñanza y aprendizaje. En este panorama, el docente tiene un reto frente a la formulación y apropiación de nuevos escenarios educativos. No obstante, el compromiso frente a estas nuevas apuestas debe consolidarse desde los diferentes estamentos educativos. En este sentido, Cabero nos dice lo siguiente: “[...] la incorporación de las TIC en los centros educativos repercutirá sobre las estructuras organizativas, el conocimiento que los sujetos tengan de la organización, el nivel de participación, las relaciones de poder, la horizontalidad y verticalidad de la información” (Cabero, 2010, p. 38). De acuerdo con lo anterior, se evidencia que desde el rol de docentes se puede contribuir a la formulación de nuevos escenarios educativos, existe una infinidad de propuestas en todas las disciplinas (Ainzua, 2014), especialmente, el uso de la herramienta conocida como *scratch*, con la que se han adelantado intervenciones didácticas en diferentes áreas del conocimiento, como las hechas por Escobar (2013), López (2014), Palma y Mora (2014) y Peña (2016). No cabe duda que para el desarrollo de la presente investigación es fundamental el uso de la tecnología para la apropiación de los conceptos de la lógica computacional. También, es importante destacar los aportes de la inteligencia artificial a las diferentes herramientas en línea, que permiten la creación de materiales educativos para la enseñanza y aprendizaje (Sánchez et al., 2020).

## **Metodología**

Esta es una investigación de tipo cuantitativo con un enfoque descriptivo: se analizaron las ventajas que tiene enseñar algoritmos y programación con el *software App Inventor* respecto a una metodología enseñanza tradicional (Hernández et al., 2014). Se inició con la aplicación de una prueba diagnóstica para evaluar el nivel de apropiación cognitivo relacionado con los algoritmos computacionales. Posteriormente, se implementó la estrategia pedagógica sobre la enseñanza y el aprendizaje de algoritmos computacionales relacionada con el *software App Inventor*. Finalmente, se aplicó una prueba final para medir la incidencia de la estrategia pedagógica.

A partir de lo anterior, se establecieron dos variables: independiente, asociada a la exposición de los estudiantes a actividades pedagógicas mediadas por *App Inventor*, y dependiente, competencia lógico-matemática relacionada con la comprensión cognitiva y procedimental de los algoritmos computacionales. Por otra parte, la población para esta investigación fue de diecinueve (19) estudiantes, los cuales cursaban el grado décimo en el momento del trabajo de campo, esta constituye una muestra no probabilística. Para adelantar la investigación, se propuso como objetivo mejorar la lógica computacional para la implementación de algoritmos mediada por la incorporación del *software App Inventor*. Las fases de la investigación fueron las siguientes: caracterización y diagnóstico de la población; diseño e implementación de la estrategia pedagógica y validación de la propuesta pedagógica.

## **Resultados**

Los resultados se presentan de acuerdo con las fases propuestas para el diseño metodológico mencionadas en el párrafo anterior.

## Caracterización y diagnóstico de la población

Con el propósito de caracterizar la población, se les aplicó a los estudiantes una prueba inicial de diagnóstico (a los 19 estudiantes de grado décimo). A continuación, se presentan los resultados del diagnóstico a los estudiantes (tabla 1).

Tabla 1. Resultados de la prueba diagnóstica

Criterios	Correcta	Incorrecta	No contestaron
Identificación de los elementos que componen un algoritmo	74 %	16 %	10 %
Identificación de los tipos de datos de entrada del algoritmo	89 %	0 %	11 %
Identificación de las estructuras simples para la construcción de un algoritmo	74 %	16 %	10 %
Identificación de las estructuras compuestas para la construcción de un algoritmo	89 %	0 %	11 %
Resolución de ejercicios de estructura condicional simple	16 %	63 %	21 %
Resolución de ejercicios de estructura condicional compuesta	21 %	21 %	58 %
Resolución de ejercicios de estructuras cíclicas	5 %	21 %	74 %
Formulación y resolución de un problema	0 %	0 %	100 %

Fuente: elaboración propia.

Los resultados de la tabla 1 muestran que, en los primeros criterios, en los que se evaluaron los aspectos conceptuales, los estudiantes evidencian un nivel de desempeño superior, con altos porcentajes de preguntas correctas (74 % y 89 %). No obstante, cuando se revisan los criterios de resolución y formulación de problemas, las respuestas correctas son mínimas: del 5 %, el 16 % y el 21 %. Lo anterior evidencia que los estudiantes asimilan los conceptos, pero no los aplican o apropian, es decir, existe una desarticulación entre la teoría y la práctica.

## Diseño e implementación de la estrategia pedagógica

Para empezar, se establece un plan de trabajo que consiste en diseñar, organizar y articular diferentes actividades apoyadas en el *software App Inventor*. Asimismo, cada una de las actividades está enfocada en fortalecer las debilidades evidenciadas en la prueba inicial. Para lograr el diseño y desarrollo de algoritmos computacionales, se hace necesario que el estudiante apropie conceptos, características, representación, operadores, variables, constantes, contadores, acumuladores, tipos de datos, estructura secuencial, estructura condicional y estructura iterativa de los algoritmos (Granada et al., 2019). Es así como se relacionan las sesiones de clase y los contenidos que han sido articulados a través de los referentes teóricos y los objetivos de esta investigación. La siguiente tabla (tabla 2) muestra de forma general las sesiones en las que los estudiantes trabajaron con el *software App Inventor*. Este trabajo se desarrolló en distintas clases, que duraron, en promedio, de una a dos horas.

Tabla 2. Diseño de las sesiones con el software *App Inventor*

Sesión	Título de la sesión	Temas a tratar
1	Presentación del software App Inventor	Creación de un nuevo proyecto: modo de diseño, editor de bloques y modos de ejecución (emulador, <i>al companion</i> y USB)
2	Vista del diseñador y editor de bloques en App Inventor	Componentes de interfaz de usuario, creación de <i>screen</i> y manejo de bloques de control, lógica, matemáticas, listas y variables
3	Estructura de control secuencial de un algoritmo	Declaración de variables, asignación de valores, entrada de datos, procesamiento de datos (operaciones) y reporte de resultados
4	Estructura condicional	Estructura condicional de selección simple, selección doble y selección múltiple.
5	Estructura iterativa (repetición)	Estructura iterativa para y mientras

Fuente: elaboración propia.

El software *App Inventor* tiene dos (2) componentes denominados “vista diseñador” y “editor de bloques”. En el componente denominado “vista diseñador”, se encuentran la *screen* activa y los diferentes componentes de interfaz del usuario —como botones, etiquetas, campos de texto, selector de lista y campos de contraseñas—, los cuales se utilizaron para el diseño de los algoritmos computacionales. En el componente denominado “editor de bloques”, se encuentran los bloques referentes a las variables, operadores matemáticos de texto, las estructuras condicionales e iterativas, que hicieron parte del desarrollo de los algoritmos computacionales. A continuación, en la figura 1 se presentan las interfaces del *software* educativo.

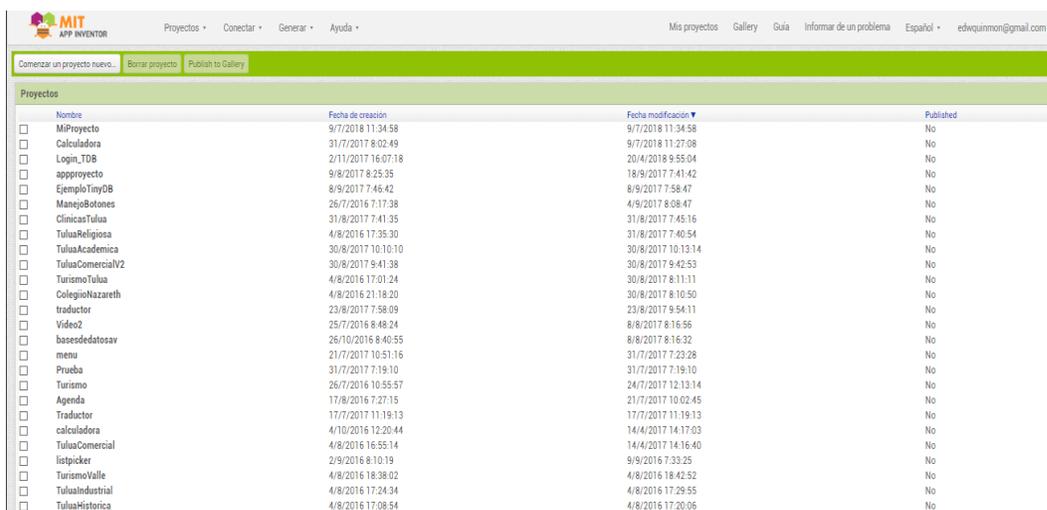


Figura 1. Pantalla de inicio del software App Inventor.

Fuente: elaboración propia.

De igual manera, la propuesta pedagógica para cada sesión presentó componentes como tema, tiempo, recursos, objetivos, descripción de la actividad y evaluación.

## Implementación

App Inventor es una herramienta TIC muy útil y fácil de usar, que permite, de forma dinámica, elaborar las actividades propuestas a lo largo del desarrollo de un proyecto, con el objetivo de que el estudiante pueda apropiarse y realizar la transferencia entre el diseño, la construcción y programación de algoritmos computacionales. De igual manera, se utilizará *Edmodo*, donde se creó un grupo académico mediante el cual se compartieron cada una de las sesiones de clase (figura 2).

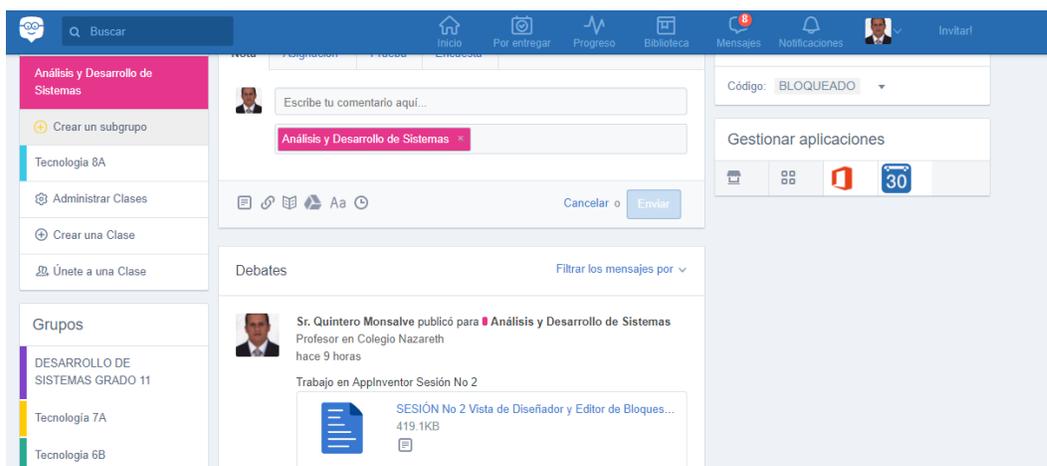


Figura 2. Aula de clase en *Edmodo*.

Fuente: elaboración propia.

Para complementar la estrategia pedagógica por cada sesión, el estudiante debió adelantar un proyecto. Por ejemplo, en esta sesión se trabajó un algoritmo sencillo sobre el cálculo de la suma de dos números, se desarrolló en 2 horas de clases con el apoyo de una guía de trabajo enviada a los estudiantes por la red social académica *Edmodo* y el software *App Inventor*. En esta sesión, se trabajó el tema correspondiente a la estructura de control secuencial de un algoritmo y se ejecutaron los tres procesos de tratamiento de información que son: leer datos, ejecutar procesos y reportar resultados. En la figura 3, se presenta uno de los proyectos adelantados por los estudiantes.



Figura 3. Proyecto “suma” ejecutado en el emulador de *App Inventor*.

Fuente: elaboración propia.

### **Validación de la estrategia pedagógica**

A partir del desarrollo de la estrategia pedagógica, la cual se fundamenta en las teorías del constructivismo, postulase plantea la existencia y prevalencia de procesos activos en la construcción del conocimiento (Díaz y Hernández, 2015). A partir de un plan de clases integrado por sesiones, incorporando el uso de las TIC, se propuso que cada uno de los estudiantes superara las dificultades presentadas en la prueba diagnóstica. Asimismo, con el apoyo del *software App Inventor* se desarrollaron cada una de las sesiones que integraron la estrategia pedagógica. Luego, con la prueba final y con el análisis presentado a continuación, se buscó verificar la incidencia del software didáctico, como estrategia pedagógica, en la apropiación de un aprendizaje significativo (Beltrán y Cerrero, 2018).

Con el fin de medir el desempeño de los estudiantes después de aplicar la estrategia pedagógica, la validación, análisis e interpretación de los datos obtenidos se realizó aplicando una prueba final. Los resultados obtenidos muestran un avance significativo en el diseño y construcción de algoritmos computacionales: el porcentaje general de repuestas correctas en la prueba inicial se situó en un 62,5 % y, en la prueba final, los resultados correctos fueron del 100 % adicionalmente, se presenta el análisis correlacional entre la variable independiente, exposición de los estudiantes a actividades pedagógicas mediadas por *App Inventor*, y la variable dependiente, competencia lógico-matemática.

De igual manera, el coeficiente de correlación entre estas dos variables es de 0,36, lo que significa que hay una correlación positiva entre las dos variables (figura 4).

#### CITAS

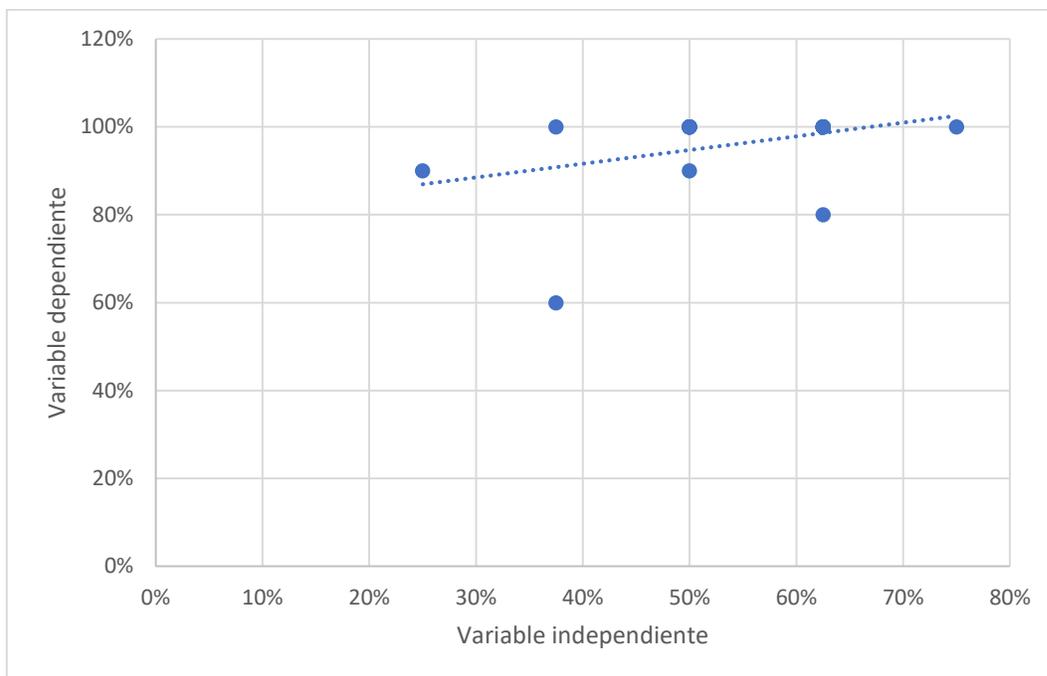


Figura 4. Coeficiente de correlación entre las variables independiente y dependiente.

Fuente: elaboración propia.

Se evidencia que el *software App Inventor* ayudó al estudiante en la construcción de su propio conocimiento, ya que le permitió visualizar, analizar, construir y corregir los errores en el diseño y construcción de algoritmos computacionales, de esta manera, y como lo menciona Piaget, se construye conocimiento cuando hay reestructuración y reconstrucción de aprendizajes previos, lo que es fundamental en esta investigación, pues la estrategia pedagógica permitió que el estudiante recordara y reforzara conocimientos relacionados con conceptos básicos de algoritmos, estructura secuencial algorítmica, estructuras condicionales simples y dobles y estructuras iterativas (repetición): temas fundamentales para lograr desarrollar el diseño y la construcción de algoritmos computacionales, objeto de la presente investigación. Así, también le permitió relacionar de manera correcta los conceptos teóricos con la práctica. Por tal razón, el constructivismo aporta desde diferentes teorías, como es el caso del aprendizaje significativo, que quiere hacer del aprendizaje una experiencia significativa para la vida cotidiana del estudiante mediante la correlación de los temas a tratar y apoyado el conocimiento previo. Para esta investigación, los resultados muestran que el estudiante consiguió apropiarse de la competencia lógico-matemática y de la relación de esta con situaciones cotidianas; ante esto, decide adoptar una actitud de concientización, frente a sus debilidades, y de responsabilidad y entrega, para superarlas. Esto se sustenta desde los resultados de respuestas correctas en la prueba inicial y en la prueba final, en las que el porcentaje de respuestas correctas siempre fue superado por parte de cada estudiante de la muestra (Díaz y Hernández, 1999).

## Discusión y conclusiones

La estrategia pedagógica buscó superar las debilidades y reforzar algunos conceptos, especialmente, en el saber hacer, ubicado en el diseño y construcción de algoritmos computacionales en los estudiantes de grado décimo con cadena de formación en el Técnico de Programación. Desde los objetivos formulados para el desarrollo de la investigación, y teniendo en cuenta específicamente el momento en el que se realizó la aplicación de la prueba inicial, se detectaron debilidades en la parte conceptual y procedimental relacionadas con el diseño y construcción de algoritmos computacionales: se evidenciaron debilidades en cuanto al análisis de los enunciados de los problemas, la construcción de la estructura secuencial del algoritmo, el uso de las variables y constantes, la aplicación de las estructuras condicionales e iterativas y la formulación y resolución de problemas de tipo práctico.

A partir del anterior análisis y los hallazgos evidenciados en la prueba inicial, se diseñaron sesiones de clase apoyadas por el *software App Inventor* como estrategia pedagógica. En la primera sesión, se realizó la presentación de este *software* educativo, sesión que tuvo una excelente acogida por parte de los estudiantes con el 100 % de asistencia. La segunda sesión trabajó el tema de las vistas: el diseñador y el editor de bloques en la *App Inventor*. En esta sesión, la asistencia fue del 95 %. La tercera sesión desarrolló el tema de estructura de control secuencial de un algoritmo, en la cual la asistencia fue del 95 %, a partir del reconocimiento de las debilidades en este tema. En esta sesión, los estudiantes fueron muy dedicados y aplicados en el desarrollo de la clase. La cuarta sesión, la cual tuvo una asistencia del 100 %, trabajó el tema de estructuras condicionales de un algoritmo y se realizaron la solución a algoritmos relacionados con estructura condicional simple y estructura condicional doble. En esta última sesión, los estudiantes presentaron un poco de dificultad, no obstante, siempre tuvieron una actitud asertiva crítica a superar las debilidades. En la quinta sesión sobre estructura iterativa (repetición), la cual conto con una asistencia del 95 % (en las dos clases), los estudiantes desarrollaron los ejercicios propuestos especificados en la guía de trabajo, con el apoyo del *software App Inventor*.

Finalmente, la validación de la estrategia pedagógica se hizo por medio de la prueba final, que evaluó los conceptos básicos de algoritmos, estructura de control secuencial de un algoritmo, estructuras condicionales (simples y compuestas) y estructura iterativa (repetición). Los resultados, a partir de la aplicación de esta prueba, fueron significativamente altos y prueban que todos los estudiantes obtuvieron un porcentaje más alto de respuestas correctas que los obtenidos en la prueba diagnóstica. De igual manera, la implementación del *software App Inventor*, como herramienta mediadora en el proceso de enseñanza-aprendizaje, les permitió a los estudiantes fortalecer sus habilidades lógico-matemáticas y promover una cultura de trabajo autónomo, hecho que significativamente se reflejó en los puntajes obtenidos en la prueba final.

También, se puede concluir, desde la incorporación de las TIC, que la aplicación móvil y gratuita del *App Inventor* fue una ventaja provechosa para el desarrollo de las actividades propuestas, ya que permitió descargar los archivos de los códigos fuentes y ejecutables y probarlos en los celulares y tabletas de los estudiantes, validando el funcionamiento de las soluciones creadas en *App Inventor*.

Finalmente, se puede concluir que la estrategia pedagógica implementada mediante el uso del *software App Inventor* en el proceso de enseñanza-aprendizaje de los algoritmos computacionales dinamizó las actividades académicas en el aula de clases, ofreciendo a los estudiantes la oportunidad de construir su propio conocimiento y de utilizar sus conocimientos previos como base para desarrollar los nuevos. Además, en el ámbito tecnológico, los estudiantes ampliaron sus saberes sobre diferentes aplicaciones y herramientas que fortalecen las competencias lógico-matemáticas y el uso pedagógico que se hace de los recursos tecnológicos que posee una institución educativa. También, es importante destacar el papel que cumplen actividades lúdicas en contextos educativos (Silva y Montañez, 2019). De cierta manera, el uso del *App* incentiva en los estudiantes la motivación al aprendizaje, por lo novedoso de la práctica, y les permite superar, pensemos en la población que fue objeto de estudio de esta investigación, dificultades relacionadas a la teoría y la práctica.

## Referencias

- Ainzua, J. I. (2014). *Calentamiento global con scratch y escuelas eficientes con Arduino* [tesis de maestría inédita]. Universidad Pública de Navarra. <http://bit.ly/2uOhbOi>
- Araya, V., Alfaro, M. y Andonegui, M. (2007). Constructivismo: orígenes y perspectivas. *Red de Revistas Científicas de América Latina y el Caribe, España y Portugal*. Obtenido de <http://www.redalyc.org/articulo.oa?id=76111485004>
- Ausubel, N. (1983). *Psicología educativa: un punto de vista cognoscitivo*. Editorial Trillas. <http://bit.ly/2u6OXhu>
- Beltrán, E. y Cerrero, P. (2018). *La Tecnología como herramienta para la enseñanza de la probabilidad simple en eventos probables y equiprobables, por medio de una aplicación para android* [tesis de maestría inédita]. Universidad Distrital Francisco José de Caldas. <http://bit.ly/2Ula62t>
- Botero, R. D. y Trefftz, H. (2015). MEDRA para el aprendizaje en lógica de programación orientada a objetos mediante un juego. *Memorias de Investigación en Ingeniería de Sistemas e Informática*, 81-92.
- Cabero, J. (2010). Los retos de la integración de las TICs en los procesos educativo. Límites y posibilidades. *Perspectiva Educativa*, 49, 32-61.
- Carretero, M. (2009). *Constructivismo y educación*. Ediciones Paidós.
- Coloma, C. R. y Tafur, R. M. (1999). El Constructivismo y sus Implicancias en Educación. *Educación*, 8(16), 217-244. <http://bit.ly/2OgTgxl>
- Colombia Aprende. (2018). *Currículos exploratorios*. Colombia Aprende. <http://bit.ly/38YdRPq>
- Díaz, A., & Hernández, A. (2015). *Constructivismo y aprendizaje significativo*. McGraw Hill.
- Escobar, D. F. (2013). *Incidencia de la Implementación del Ambiente de Programación Scratch, en los Estudiantes de Media Técnica, Para el Desarrollo de la Competencia Laboral General de Tipo Intelectual Exigida por el Ministerio de Educación Nacional Colombiano* [tesis inédita de maestría]. Universidad de Santander; Tecnológico de Monterrey. <http://bit.ly/2Ua6b8u>
- Granada, R., Barwaldt, R. y Espindola, D. (2019). *Glossary of computational terms as a stimulus to programming logic: A case study with deaf students(Conference Paper)* [conferencia]. IEEE
- Hernández, R., Fernández, C. y Baptista, M. D. (2014). *Metodología de la investigación*. Editorial McGraw Hill.
- López, J. C. (2008). *Algoritmos y programación en la educación escolar*. Fundación Gabriel Piedrahita Uribe.
- López, J. C. (2014). *Actividades de Aula con Scratch que favorecen el uso del Pensamiento Algorítmico, el Caso del Grado 3° en el INSA* [tesis inédita de maestría]. Universidad Icesi. <http://bit.ly/2uOikW6>

- Moreira, M. A. (2005). Aprendizaje significativo crítico (Critical meaningful learning) Indivisa. *Boletín de Estudios e Investigación*, (6), 83-102
- Palma, I. y Mora, M. A. (2014). Aplicación para dispositivos móviles en el Robot Lego NXT como herramienta didáctica en educación secundaria. *Revista de Computación y Sistemas (CyS)* 1-5. <http://bit.ly/2GJZX7q>
- Palma, C. A. y Sarmiento, R. E. (2015). Estado del arte sobre experiencias de enseñanza de programación a niños y jóvenes para el mejoramiento de las competencias matemáticas en primaria. *Revista Mexicana de Investigación Educativa*, 20(65), 607-641. <http://www.redalyc.org/pdf/140/14035408013.pdf>
- Peña, E. D. (2016). *Estrategia didáctica para el mejoramiento de los proyectos de tecnología e informática de los estudiantes de los grados 6° y 7° de la IE. Escuela Normal Superior Pedro Justo Berrio de Santa Rosa de Osos* [tesis de maestría inédita]. Universidad Pontificia Bolivariana <http://bit.ly/38VZI53>
- Piaget, J. (2000). *La equilibración de las estructuras cognitivas*. Siglo XXI.
- Rivas, M. (2008). *Procesos cognitivos y aprendizaje significativo*. Organización Educativa de la Comunidad de Madrid. Obtenido de <http://bit.ly/37Odfvn>
- Romero Trenas, F. (2009). Aprendizaje significativo y constructivismo. *Revista Digital Para Profesionales de la Enseñanza*, (3). <http://bit.ly/31cJzWp>
- Sánchez, L., Alor Hernández, G., Rosales, V., Cortes, C. y Sánchez, J. (2020). Generating educational mobile applications using UIDPs identified by artificial intelligence techniques. *Computer Standards and Interfaces*, 70. <https://doi.org/10.1016/j.csi.2019.103407>
- Serrano, J. M. y Pons Parra, R. M. (2011). El Constructivismo hoy: enfoques constructivistas en educación. *Revista Electrónica de Investigación Educativa*, 13(1). <http://bit.ly/2RJWwEO>
- Silva, A. y Montañez, L. (2019). Aprendizaje psicomotriz en el área de Educación Física, Recreación y Deportes mediado por el uso de software educativo (Psychomotor learning in the area of Physical Education, Recreation, and Sports, mediated by the use of educational software). *Retos*, 36, 302-309. <https://doi.org/10.47197/retos.v36i36.67131>
- Taborda, H. y Medina, D. (2012). *Programación de computadores y desarrollo de habilidades de pensamiento en niños escolares: fase exploratoria*. Universidad Icesi. <http://bit.ly/2RLRSWO>
- Tünnermann, C. (2011). El constructivismo y el aprendizaje de los estudiantes. *Universidades*, (48), 21-32. <https://www.redalyc.org/pdf/373/37319199005.pdf>