

APRENDIZAJE DE STEM EN LA INFANCIA A TRAVÉS DE ENTORNOS DE
PROGRAMACIÓN VISUAL Y DISPOSITIVOS DE HARDWARE INTERACTIVO

FERNANDO JOSÉ VIZCAINO LINERO
SIXTA VERÓNICA SAMBONI BUESAQUILLO

OSCAR LEONARDO ACEVEDO PABON

INSTITUCIÓN UNIVERSITARIA POLITÉCNICO GRANCOLOMBIANO
FACULTAD DE INGENIERÍA, DISEÑO E INNOVACIÓN
MAESTRÍA EN EDUCACIÓN STEM
2026

Tabla de contenido

INTRODUCCIÓN	2
CAPÍTULO I. CONTEXTUALIZACIÓN DEL PROBLEMA DE INVESTIGACIÓN	5
1.1 Planteamiento del problema	5
1.2 Pregunta de investigación	7
1.3 Justificación	8
1.4 Objetivos de investigación	9
1.4.1 Objetivo general	9
1.4.2 Objetivos específicos	9
CAPÍTULO II. MARCO REFERENCIAL	1
2.1 Antecedentes	1
2.2 Marco teórico	2
2.2.1 Teoría constructivista	2
2.3 Marco conceptual	3
2.3.1 Educación STEM	3
2.3.2 Hardware interactivo	3
2.3.3 Pensamiento Computacional (PC)	3
2.3.4 Pensamiento Sistémico (PS)	5
2.3.5 Motivación	8
CAPÍTULO III. METODOLOGÍA	1
3.1 Paradigma y enfoque que enmarcan el estudio	1
3.2 Diseño y métodos	1
3.2.1. Instrumentos	2
3.2.2 Validación de instrumentos	3
3.3 Población, muestra y escenario escolar	7
3.4 Estructura de la intervención didáctica	8
3.4.1 Descripción general de la intervención	8
3.4.2 Fase 1. Programación visual con Scratch	8
3.4.3 Fase 2. Programación con Flowcode y montaje en hardware	9
3.4.4 Materiales, recursos y componentes electrónicos	10
3.4.5 Sustento teórico de las decisiones de diseño	10
3.5 Fases de la investigación	11
3.6 Metodología de interpretación de resultados	12

3.7. Criterios de calidad y consideraciones éticas	14
CAPÍTULO IV. ANÁLISIS Y DISCUSIÓN DE RESULTADOS.....	1
4.1. Resultados cuantitativos	1
4.1.1. Resultados pretest–postest de habilidades STEM	1
4.1.2. Resultados prueba de motivación	6
4.2 Resultados cualitativos	9
4.2.1. Resultados de entrevistas y observación	9
CAPÍTULO V. CONCLUSIONES Y RECOMENDACIONES.....	1
5.1 Conclusiones	1
5.3 Implicaciones para el diseño curricular y trabajos futuros	3
5.4 Limitaciones del estudio.....	4
5.5 Recomendaciones	5
REFERENCIAS BIBLIOGRÁFICAS	6
ANEXOS	1
ANEXO A. CUESTIONARIO PRETEST – DIAGNÓSTICO DE COMPETENCIAS STEM	1
ANEXO B. CUESTIONARIO POSTEST – EVALUACIÓN DE COMPETENCIAS STEM	4
ANEXO C. FORMATO GUIA ENCUESTA DE MOTIVACIÓN HACIA STEM.....	7
ANEXO D. FORMATO GUÍA ENTREVISTA SEMIESTRUCTURADA A ESTUDIANTES	9
ANEXO E. FORMATO GUÍA DE OBSERVACIÓN DE CLASE	11
ANEXO F. EVIDENCIAS DE LA IMPLEMENTACIÓN	1
ANEXO G. CRONOGRAMA DE INTERVENCIÓN DIDÁCTICA	1
<i>Aplicación Cuestionario Postest – Evaluación de competencias STEM.....</i>	<i>3</i>
Formato guía Guía de observación de clase	4
ANEXO H. FORMATO DE AUTORIZACIÓN INSTITUCIONAL PARA EL DESARROLLO DE LA INVESTIGACIÓN	1
ANEXO I. CONSENTIMIENTO INFORMADO DE REPRESENTANTE LEGAL.....	1
ANEXO J. CONSENTIMIENTO INFORMADO.....	1

Lista de Figuras

Figura 1 Porcentaje de acierto en las preguntas de prueba de habilidades STEM antes y después de la intervención didáctica.	1
Figura 2 Respuestas de los estudiantes a los ítems de la prueba de motivación en escala de Likert.	6

Lista de Tablas

Tabla 1 Componentes electrónicos del semáforo diseñado en Proteus	10
Tabla 2 Resultados de la prueba de Wilcoxon-Pratt sobre la mejora en los resultados de la prueba de habilidades STEM.....	4

INTRODUCCIÓN

La enseñanza de las áreas STEM en la educación básica colombiana enfrenta un rezago persistente respecto a las competencias que demanda el siglo XXI. Las pruebas nacionales e internacionales ubican al país en posiciones medias-bajas en matemáticas y ciencias, y los informes del Ministerio de Educación Nacional documentan una integración todavía incipiente del pensamiento computacional en la educación primaria rural. Wing (2006) delimitó hace dos décadas el pensamiento computacional como una competencia básica para ciudadanos del mundo contemporáneo; Arnold y Wade (2015) situaron en el mismo plano el pensamiento sistémico, entendido como la capacidad de comprender relaciones, retroalimentaciones y comportamientos emergentes. La Institución Educativa Agropecuaria José Dolores Daza, ubicada en el corregimiento de Los Milagros, comparte los rasgos habituales de la escuela rural colombiana: infraestructura tecnológica limitada, exposición escasa de los estudiantes a entornos de programación y ausencia de experiencias previas con *hardware* programable. Sobre ese telón de fondo se inscribe el problema que aborda este trabajo.

La observación sistemática del proceso escolar en sexto grado reveló que los estudiantes llegaban con un nivel bajo-medio de pensamiento computacional y sistémico, y sin un canal pedagógico que articulara la teoría con la producción de artefactos tecnológicos. De ahí surge la pregunta que orienta el estudio: ¿cómo contribuye una intervención basada en entornos de programación visual y dispositivos de *hardware* interactivo al desarrollo del pensamiento computacional y sistémico en estudiantes de sexto grado de la Institución Educativa Agropecuaria José Dolores Daza? El objetivo general que responde a esa pregunta consiste en evaluar el aporte

de una estrategia pedagógica fundamentada en *Scratch*, *Flowcode* y montaje de *hardware* al desarrollo de ambas formas de pensamiento en la población señalada.

Los resultados organizan el aporte de la investigación en cuatro aspectos. El primero es de orden pedagógico: la intervención resultó viable e implementable en un aula rural, con los recursos disponibles y dentro del calendario escolar ordinario; los estudiantes recorrieron la secuencia *Scratch* → *Flowcode* → *hardware* sin abandonos y produjeron el artefacto integrador. El segundo aspecto es conocimiento cuantitativo: el contraste pretest–posttest mostró mejoras superiores a veinte puntos porcentuales en descomposición y abstracción, mejoras intermedias en modelado y mejoras inferiores a diez puntos en retroalimentación y justificación de decisiones; la escala Likert de motivación registró puntuaciones mayoritariamente positivas, con picos en interés con placer, trabajo en grupo y percepción del pensamiento computacional. El tercer aspecto es conocimiento cualitativo: las entrevistas y la observación de aula documentaron entusiasmo sostenido, superación del temor inicial ante el *hardware*, aumento del trabajo colaborativo y aparición de prácticas espontáneas de ensayo y corrección, aunque también persistencia de dificultades para verbalizar la causa de un error. El cuarto aspecto es, quizá, el de mayor alcance prospectivo: la investigación identificó que retroalimentación y justificación de decisiones son habilidades que no maduran plenamente en sexto grado, lo que abre oportunidades concretas para el diseño curricular en secundaria y para futuras investigaciones en grados superiores.

El documento se organiza en cinco capítulos. El Capítulo I contextualiza el problema de investigación con su planteamiento, pregunta, justificación y objetivos. El Capítulo II construye el marco referencial con antecedentes, marco teórico y conceptos centrales del estudio. El Capítulo III presenta la metodología, los instrumentos empleados, su validación y la estructura detallada de la intervención didáctica. El Capítulo IV expone el análisis y la discusión de los resultados

cuantitativos y cualitativos. El Capítulo V articula las conclusiones por objetivo específico, las implicaciones para el diseño curricular y los trabajos posteriores, las limitaciones del estudio y las recomendaciones dirigidas a la institución, al profesorado y a la comunidad investigadora. Al final, se incluyen las referencias bibliográficas y los anexos que soportan cada instrumento y evidencia.

CAPÍTULO I. CONTEXTUALIZACIÓN DEL PROBLEMA DE INVESTIGACIÓN

1.1 Planteamiento del problema

La educación basada en ciencias, tecnología, ingeniería y matemática, denominada STEM por sus siglas en inglés (National Science & Technology Council NSTC, 2024) se ha logrado consolidar como un enfoque importante dentro de la formación académica en diferentes niveles educativos, desde escuelas primarias y preescolares hasta educación superior de pregrado y posgrado. La meta central de esta educación consiste en que los estudiantes sean capaces de desenvolverse en un mundo que cada día es más desarrollado en los ámbitos de la tecnología, la innovación y la globalización.

A pesar del continuo crecimiento de la importancia y la presencia de la educación de tipo STEM, todavía se tiene que lidiar con retos evidentes en su introducción en los niveles primarios, en particular cuando la meta es la de trabajar de manera explícita la potenciación del pensamiento computacional y el pensamiento sistémico, habilidades cruciales en el manejo de la tecnología y en la participación en innovación actuales.

La vigencia del pensamiento computacional como aprendizaje escolar se sostiene en un itinerario que va del ensayo seminal de Wing (2006), quien postula esta habilidad como competencia básica del siglo XXI junto a la lectura y la aritmética, hasta la revisión de Grover y Pea (2013), que sitúan su incorporación al currículo K-12 como respuesta a la expansión de la cultura digital en la escolaridad obligatoria. El informe del Joint Research Centre de la Comisión Europea recoge esa trayectoria y argumenta la inclusión sistemática del pensamiento computacional en los currículos nacionales (Bocconi et al., 2016). La OCDE (2019) incorpora tales capacidades al Learning Compass 2030 como aprendizajes orientados a la agencia estudiantil

y a la resolución de problemas complejos, las competencias digitales condicionan hoy las oportunidades educativas de la población rural (UNESCO, 2023). Dicho marco justifica que estudiantes de sexto grado de Los Milagros accedan a Scratch y Flowcode como mediaciones formativas.

El interés contemporáneo por el pensamiento sistémico responde a la necesidad de comprender fenómenos interconectados de índole socioambiental, tecnológica y educativa. Senge (2006) postula la quinta disciplina como aprendizaje organizacional que rehúye las explicaciones lineales frente a problemas de largo alcance. Una taxonomía posterior distingue niveles progresivos de razonamiento sistémico y orienta su enseñanza en contextos escolares (Stave y Hopper, 2007). Desde la didáctica STEM, York et al. (2019) identifican aplicaciones específicas en química, biología, ingeniería y educación ambiental, donde los aprendices reconocen componentes, relaciones y retroalimentaciones dentro de totalidades organizadas. Para cultivar y evaluar esta habilidad en cursos iniciales de ciencia e ingeniería (Lavi y Bertel, 2024). Tal acumulado teórico respalda que un semáforo didáctico, pese a su aparente sencillez, introduzca a los estudiantes de sexto grado en el análisis de entradas, procesos y salidas como sistema cerrado y observable

La enseñanza a partir de modelos tradicionales es en muchas ocasiones predominante, y estos modelos son poco aptos para el desarrollo de habilidades y procesos asociados con el pensamiento computacional (abstracción y descomposición) y con el pensamiento sistémico (modelado y retroalimentación), además de prácticas de depuración, verificación y ajuste (Aravena et al., 2024). Al ser la infancia un periodo en el que se deben consolidar estas habilidades, es urgente implementar modelos más adecuados: las metodologías activas como la programación visual y el *hardware* interactivo son, a este respecto, una excelente opción. Sin embargo, la práctica

docente queda a menudo limitada por aspectos como los costos, la disponibilidad de recursos, la formación del profesorado y la resistencia al cambio.

La ausencia de estrategias que generen una integración efectiva de metodologías innovadoras provoca una limitación en la capacidad de desarrollo de competencias asociadas con la sociedad digital, en particular aquellas relacionadas con el pensamiento computacional y sistémico. Según Esquer y Fernández (2021), el acceso temprano a la metodología STEM con recursos tecnológicos es capaz de fomentar de forma significativa el pensamiento con base computacional y la resolución de problemas, aunque la falta de capacitación docente, escasos recursos y las metodologías poco atractivas generan un efecto negativo en su implementación. En este sentido, el uso de programación visual y *hardware* interactivo puede convertirse en un medio para dinamizar el aprendizaje STEM en la infancia, orientando el diseño, prueba y ajuste de soluciones y fortaleciendo habilidades desde edades tempranas.

1.2 Pregunta de investigación

La pregunta central de investigación de estudio es la siguiente:

¿Cómo contribuye la integración de entornos de programación visual y dispositivos de *hardware* interactivo al aprendizaje STEM en niños de grado sexto?

Esta pregunta se sistematiza en la siguientes preguntas más específicas:

¿Qué impacto tiene la programación visual en el desarrollo del pensamiento computacional (abstracción y descomposición) y del pensamiento sistémico (modelado y retroalimentación)?

¿El uso de *hardware* interactivo favorece la comprensión de los principios básicos de STEM?

¿Es posible implementar una estrategia pedagógica que articule la programación visual y el *hardware* en experiencias de aprendizaje significativo?

¿Cuáles son los beneficios y limitaciones de aplicar estas metodologías en contextos educativos de la infancia?

1.3 Justificación

Con el continuo avance y desarrollo tecnológico y científico que la especie humana ha experimentado con el paso de los años, se ha creado una creciente necesidad de adaptar los procesos educativos a los desafíos de la sociedad contemporánea. La aceleración en los procesos de transformación tecnológica ha provocado una creciente demanda del desarrollo en las competencias del enfoque STEM. Con lo anterior, se tiene que, desde edades tempranas, todos y cada uno de los niños requieren de una formación capaz de fortalecer las habilidades cognitivas superiores tales como como el razonamiento lógico y el razonamiento cuantitativo, así como la capacidad de solución de problemas creativos. Este estudio se enfoca en el desarrollo del pensamiento computacional y el pensamiento sistémico (Videla et al., 2021).

Los entornos de programación visual y dispositivos de *hardware* interactivo en la enseñanza de STEM constituyen una oportunidad para dinamizar el aprendizaje, haciéndolo más atractivo, accesible y práctico. La programación visual, al basarse en lenguajes de bloques intuitivos, permite a los niños iniciarse en el pensamiento computacional sin la complejidad sintáctica de los lenguajes tradicionales, favoreciendo la comprensión de la lógica y la secuenciación (Marín et al., 2023). Por otro lado, el *hardware* interactivo, como microcontroladores, sensores y actuadores, complementa la programación visual al permitir que los estudiantes observen resultados tangibles en tiempo real y comprendan cómo las partes del sistema se relacionan entre sí, lo que fortalece el pensamiento sistémico, el aprendizaje activo y la experimentación (Guanotuña et al., 2024).

Este estudio adquiere importancia tanto en el plano académico como en el social. En el plano académico, este trabajo produce tanto teorías como aplicaciones que se pueden utilizar para futuras investigaciones y enriquecer la discusión sobre la enseñanza de STEM en la infancia. En el plano

social, el proyecto se suma a la tarea de enseñar a los estudiantes a ser capaces de entender, analizar y mirar con otros ojos los desafíos a los que se enfrentan en la actualidad tecnológica, a través de la enseñanza integral que les acerca a la diversidad y a la complejidad de los ambientes en los que deberán desenvolverse.

1.4 Objetivos de investigación

1.4.1 *Objetivo general*

Analizar cómo la integración de la programación visual y el uso de *hardware* interactivo contribuyen al aprendizaje de competencias STEM en 30 niños de grado sexto de la Institución Educativa Agropecuaria José Dolores Daza, en el municipio Bolívar, departamento Cauca.

1.4.2 *Objetivos específicos*

Diseñar instrumentos de evaluación para el pensamiento computacional y el pensamiento sistémico en estudiantes de grado sexto, en el marco de una intervención con programación visual y *hardware* interactivo.

Analizar el nivel de desarrollo del pensamiento computacional y el pensamiento sistémico en estudiantes de grado sexto antes y después de la intervención pedagógica.

Evaluar la pertinencia y los alcances de la estrategia pedagógica implementada, considerando la motivación y la experiencia de aprendizaje en el aula.

CAPÍTULO II. MARCO REFERENCIAL

2.1 Antecedentes

Estudios anteriores muestran evidencia convergente sobre el valor de integrar programación y *hardware* en experiencias STEM y sobre la necesidad de medir competencias con instrumentos adecuados. Bajo esta perspectiva, Çetin y Demircan (2020) señalan que la robótica programable favorece el pensamiento computacional, la resolución de problemas y la colaboración cuando existe una secuencia estructurada. Por otro lado, Feijoo y Rodríguez (2022) reportan mejoras en comunicación, trabajo colaborativo e interés por tecnología e ingeniería en actividades tipo reto con plataformas *hardware-software*. Por su parte, Stolpe y Hallström (2023) explican que la programación visual por bloques facilita la comprensión de conceptos y la integración con circuitos y robótica, con aportes al pensamiento sistémico cuando se trabaja con sistemas y relaciones entre componentes. Ahora bien, Marín et al. (2023) advierten que, en Colombia, la implementación requiere estrategias situadas por limitaciones de recursos e integración curricular, aspecto que justifica una propuesta viable en contexto rural.

Respecto a medición y validez, Ribarić et al. (2025) aportan un referente de validación psicométrica de escalas de motivación STEM en población escolar, útil para justificar dimensiones e ítems de la medición motivacional. Así, Davis et al. (2023) muestran que el autoinforme y las evaluaciones basadas en escenarios capturan facetas distintas del pensamiento sistémico, por ello conviene apoyar la medición en tareas de desempeño y triangulación. Finalmente, Guggemos et al. (2023) respaldan el uso de pruebas de desempeño y criterios psicométricos para interpretar pensamiento computacional con mayor claridad, lo que sustenta la elección de instrumentos comparables y una lectura orientada a cambios por intervención.

2.2 Marco teórico

2.2.1 Teoría constructivista

La perspectiva constructivista aporta un marco para comprender el aprendizaje como un proceso activo mediado por la interacción social y el uso de herramientas culturales. En línea con lo anterior, Vygotsky (1995) plantea que la enseñanza favorece la construcción de conocimiento cuando organiza experiencias donde el estudiante participa, comunica y actúa con apoyo del entorno. En este estudio, la orientación constructivista se utiliza como encuadre para interpretar el trabajo con programación visual y *hardware* interactivo como una situación de aula donde se observan colaboración, comunicación técnica y resolución de problemas durante la ejecución de retos. Por otro lado, Kanobel et al. (2019) señalan que estrategias centradas en participación y trabajo colaborativo pueden favorecer motivación y comprensión en contextos STEM. Por ello, el enfoque constructivista se retoma únicamente para situar el análisis de dinámicas observables, sin desplazar el foco conceptual que definen los instrumentos del estudio.

Desde esta perspectiva, la intervención didáctica se concibió como una secuencia de retos progresivos en la que el estudiantado construyó soluciones, contrastó resultados y ajustó procedimientos mediante programación visual y *hardware* interactivo. La mediación pedagógica orientó la participación activa, el intercambio entre pares y la revisión de errores durante el desarrollo de cada tarea, de modo que el aprendizaje no dependió de la recepción de contenidos aislados, sino que dependió de la relación entre acción, reflexión y verificación. En esa lógica, la propuesta promovió trabajo colaborativo, comunicación sobre los procedimientos y toma de decisiones frente al funcionamiento del sistema, con el fin de convertir el aula en un espacio donde el conocimiento se elaboró a partir de la resolución de problemas y de la interpretación de resultados observables, en coherencia con una perspectiva constructivista del aprendizaje (Vygotsky, 1995).

2.3 Marco conceptual

2.3.1 Educación STEM

La educación STEM integra ciencia, tecnología, ingeniería y matemáticas en experiencias de aprendizaje que conectan conceptos con problemas y productos. En este estudio, educación STEM se entiende como el contexto formativo dentro del cual se articularon la programación visual y el *hardware* interactivo para promover desempeños observables y percepciones del estudiantado frente a tareas tecnológicas. Esta definición se apoya en Bybee (2013), quien describe STEM como un enfoque integrador orientado a competencias aplicadas y a la resolución de desafíos. En esta investigación, ese enfoque se expresó en una intervención didáctica orientada por retos, donde la programación visual y el *hardware* interactivo articularon conceptos, decisiones y resultados en situaciones de aula.

2.3.2 Hardware interactivo

El *hardware* interactivo refiere a dispositivos físicos, sensores y placas que responden a instrucciones y permiten verificar resultados a través de señales observables. En este estudio, el *hardware* interactivo se entiende como el recurso que convierte decisiones de programación en respuestas del sistema, lo que facilita validar, corregir y ajustar la solución durante la tarea. Esta definición sigue a Guanotuña et al. (2024), quienes describen el uso educativo de dispositivos para materializar ideas en proyectos prácticos.

2.3.3 Pensamiento Computacional (PC)

El pensamiento computacional es una forma de abordar problemas mediante estrategias de organización, representación y control de procedimientos. En este estudio, pensamiento computacional se entiende como desempeño en tareas que exigen identificar lo necesario para resolver un reto y organizar acciones en una secuencia de ejecución, con énfasis en abstracción y descomposición. Esta orientación se relaciona con la formulación de Wing (2006), citado en Stolpe

y Hallström, 2023), que presenta el pensamiento computacional como un conjunto de procesos para plantear y resolver problemas con lógica.

2.3.3.1 Resolución de problemas. La resolución de problemas, para este estudio, se entiende como un proceso de análisis y decisión orientado a una meta, donde el estudiante identifica restricciones, propone una estrategia y verifica si la solución responde al reto planteado. En actividades STEM con programación y prototipos, esta habilidad se observa cuando el grupo formula un plan, prueba la solución y ajusta la secuencia al contrastar el resultado del dispositivo con lo esperado.

La tarea exige seleccionar información pertinente y sostener decisiones con evidencias del funcionamiento del sistema. Esta definición se alinea con el enfoque de habilidades que describe la OECD, que vincula el desempeño con la capacidad de enfrentar situaciones nuevas mediante procedimientos evaluables (OECD, 2019).

2.3.3.1.1 Descomposición. La descomposición se define como la capacidad de dividir un problema en partes manejables para organizar el trabajo y construir una solución paso a paso. En programación visual, se evidencia cuando el estudiante separa el reto en subobjetivos, asigna acciones a cada parte y establece el orden de ejecución que seguirá el programa. La descomposición facilita reconocer qué información se necesita en cada etapa y qué condiciones activan una respuesta del hardware.

En el aula, también se expresa cuando el grupo distribuye tareas y luego integra resultados en una sola secuencia funcional. Esta perspectiva se articula con la idea de pensamiento computacional como herramienta para resolver problemas mediante procedimientos estructurados (Wing, 2006).

2.3.3.1.2 Abstracción. La abstracción se entiende como la habilidad para identificar lo esencial del problema y representar la solución con elementos mínimos, sin perder la lógica que gobierna el comportamiento del sistema. En tareas con bloques, la abstracción se observa cuando el estudiante elige variables, condiciones y eventos que explican el funcionamiento del prototipo, evitando detalles que no inciden en el resultado. Este proceso permite construir reglas generales, reutilizar patrones de programación y describir una solución con claridad operativa. En evaluaciones de desempeño, la abstracción se refleja en ítems que exigen interpretar qué estructura resuelve un reto y por qué una alternativa falla. Esta lectura coincide con enfoques de medición que buscan interpretar habilidades de pensamiento computacional a partir de tareas con características definidas (Guggemos et al., 2023).

2.3.4 Pensamiento Sistémico (PS)

El pensamiento sistémico se orienta a comprender sistemas como conjuntos de componentes que interactúan y producen comportamientos observables. En este estudio, pensamiento sistémico se manifiesta en el desempeño de tareas que exigen reconocer relaciones entre partes, anticipar efectos de cambios y describir cómo una acción afecta el funcionamiento del conjunto. Esta definición se operacionaliza en los instrumentos mediante modelado y retroalimentación, dado que ambas capacidades permiten representar y ajustar el comportamiento del sistema durante la intervención.

2.3.4.1 Modelado. El modelado se define como la capacidad de representar un sistema y sus relaciones para anticipar su comportamiento y explicar cómo una decisión modifica el resultado. En esta investigación, el modelado se evidencia cuando el estudiante describe, con diagramas simples o explicaciones funcionales, cómo interactúan entradas, procesos y salidas del prototipo, y luego ajusta la programación para producir un comportamiento esperado. El modelado no se limita a dibujar componentes, también implica justificar relaciones causales entre acciones del programa y respuestas del *hardware*. Esta habilidad se articula con concepciones de pensamiento sistémico que distinguen entre percepciones declaradas y desempeño en escenarios donde se evalúa la comprensión de interacciones del sistema (Davis et al., 2023).

2.3.4.1.1 Trabajo en equipo. El trabajo en equipo, en el contexto del pensamiento sistémico, se entiende como la coordinación de acciones y acuerdos para construir una solución compartida, donde cada integrante aporta información y negocia decisiones sobre el sistema. En actividades con programación visual y *hardware*, esta competencia se observa cuando el grupo distribuye roles, integra aportes y mantiene un objetivo común mientras contrasta resultados y redefine el plan. El trabajo colaborativo también implica sostener criterios para aceptar cambios, documentar ajustes y evitar que la solución dependa de una sola persona. La OECD describe la colaboración como parte del desarrollo de habilidades que combinan procesos cognitivos y sociales, lo que permite ubicar esta dimensión dentro de un desempeño observable en tareas escolares (OECD, 2019).

2.3.4.1.2 Uso de recursos. El uso de recursos se define como la capacidad de seleccionar y aprovechar materiales, tiempo y herramientas disponibles para avanzar hacia la solución del reto, manteniendo consistencia con los requisitos del sistema. En la intervención didáctica, esta habilidad se evidencia cuando los estudiantes eligen sensores, conectan componentes y consultan guías o ejemplos para resolver un fallo, sin perder el control del proceso de prueba. El uso de recursos también incluye priorizar acciones cuando hay limitaciones de tiempo o de disponibilidad de materiales, y tomar decisiones que reduzcan errores repetidos. Este enfoque coincide con una mirada de desarrollo de habilidades que considera la gestión de recursos como parte de la competencia para actuar en entornos tecnológicos y resolver tareas con criterios de desempeño (OECD, 2019).

2.3.4.2 Retroalimentación. La retroalimentación se entiende como el proceso de revisar resultados, identificar discrepancias y ajustar la solución con base en evidencias del funcionamiento del sistema. En programación y *hardware*, esta competencia se observa cuando el estudiante interpreta errores, prueba hipótesis sobre la causa y modifica bloques, conexiones o parámetros hasta lograr el comportamiento esperado. La retroalimentación exige reconocer relaciones entre acciones y efectos, por lo que se vincula con el pensamiento sistémico al orientar ajustes que consideran el sistema como un conjunto de componentes interdependientes. Este proceso se representa con claridad cuando se evalúa desempeño mediante escenarios que obligan a decidir qué cambio realizar y por qué, en lugar de limitarse a declarar una percepción (Davis et al., 2023).

2.3.4.2.1 Comunicación. La comunicación se define como la capacidad de expresar procedimientos, justificar decisiones y compartir hallazgos durante la construcción y depuración de una solución. En el aula, esta habilidad se evidencia cuando el grupo explica qué variable controló un cambio, qué condición activó una respuesta del dispositivo y qué criterio usó para decidir el siguiente ajuste. La comunicación también implica escuchar propuestas, comparar interpretaciones del error y llegar a acuerdos sobre el modelo que guía la solución. Esta dimensión se relaciona con competencias sociales que acompañan el desempeño en tareas colaborativas, donde el logro depende de coordinar significados y acciones entre integrantes del equipo (OECD, 2019).

2.3.5 Motivación

La motivación, para este estudio, se entiende como un conjunto de disposiciones que orientan la participación del estudiante en las actividades STEM, expresadas en su interés por la tarea, su creencia de capacidad para realizarla y el valor que atribuye a lo que aprende. En experiencias con programación y prototipos, la motivación se observa cuando el estudiante sostiene el esfuerzo, busca alternativas ante el error y mantiene intención de terminar el reto. Su medición requiere ítems organizados por dimensiones que permitan diferenciar actitudes de participación, sin mezclar conceptos con el desempeño cognitivo del test. Este enfoque es coherente con trabajos de validación de escalas de motivación STEM en población escolar, que reportan estructuras factoriales para interpretar resultados por dimensiones (Ribarić et al., 2025).

2.3.5.1 Actitudes. Las actitudes se entienden como disposiciones relativamente estables que orientan la forma en que el estudiante valora y afronta las actividades STEM durante la intervención. En este estudio, la medición de actitudes se organiza en interés, autoeficacia y valor percibido, ya que estas dimensiones permiten describir la relación del estudiante con la tarea sin confundirla con el rendimiento en pruebas. Las actitudes se evidencian cuando el estudiante manifiesta gusto por el reto, confianza para intentar soluciones y sentido de utilidad de lo aprendido. Esta organización por dimensiones permite interpretar resultados con criterios comparables entre estudiantes y entre momentos de evaluación, como se propone en la validación de escalas de motivación STEM en población escolar (Ribarić et al., 2025).

2.3.5.1.1 Interés. El interés se define como la inclinación del estudiante a involucrarse con la actividad por curiosidad, agrado o atención sostenida hacia el reto planteado. En la intervención, el interés se observa cuando el estudiante busca comprender cómo funciona el prototipo, propone ideas de solución y mantiene la atención durante la programación y la prueba del *hardware*. Esta dimensión no se reduce a una opinión general sobre la clase, ya que se expresa en conductas como insistir en probar alternativas y formular preguntas sobre el funcionamiento del sistema. En este sentido, Ribarić et al. (2025) ubican el interés como un componente medible de la motivación STEM que permite comparar cambios en la relación del estudiante con la tarea.

2.3.5.1.2 Autoeficacia. La autoeficacia se entiende como la creencia del estudiante sobre su capacidad para ejecutar acciones necesarias para resolver el reto propuesto, aun cuando aparecen errores o ajustes por realizar. En actividades con programación visual, esta dimensión se manifiesta cuando el estudiante asume tareas, intenta depurar un bloque y mantiene el esfuerzo pese a fallos del prototipo. La autoeficacia se expresa en decisiones concretas como volver a probar, pedir retroalimentación específica y argumentar por qué un cambio puede funcionar. La escala motivacional permite captar estas creencias sin confundirlas con el resultado final del desempeño, lo que coincide con enfoques de medición por dimensiones en estudios psicométricos recientes (Ribarić et al., 2025).

2.3.5.1.3 Valor percibido. El valor percibido se define como el juicio del estudiante sobre la utilidad y el sentido de lo que aprende para su vida escolar y para situaciones fuera del aula. En este estudio, el valor percibido se refleja cuando el estudiante conecta el reto con usos posibles de la tecnología, reconoce aprendizajes aplicables y expresa razones para considerar valiosa la actividad. Esta dimensión se diferencia del interés, porque no depende únicamente del gusto por la tarea, también depende de la interpretación de su utilidad. En escalas motivacionales STEM, el valor percibido suele funcionar como un predictor de permanencia y esfuerzo sostenido, por lo que conviene medirlo como componente diferenciado (Ribarić et al., 2025).

2.3.5.2 Participación. La participación se entiende como la implicación observable del estudiante en las actividades de aula, expresada en acciones que sostienen el avance del trabajo individual y grupal. En este estudio, la participación se observa cuando el estudiante se integra a la discusión, asume un rol durante el reto, registra avances y contribuye en la toma de decisiones para ajustar el prototipo. Esta dimensión permite valorar el involucramiento sin reducirlo a una presencia física, ya que depende de conductas que apoyan el trabajo y el aprendizaje. La medición por componentes facilita describir cómo la motivación se traduce en acciones durante la intervención, en línea con modelos que separan actitudes de comportamientos de involucramiento (Ribarić et al., 2025).

2.3.5.1.1 Disposición. La disposición se define como la intención del estudiante de participar y persistir en la tarea, incluso cuando el reto exige ensayo, ajuste y coordinación con el grupo. En la intervención, esta dimensión se evidencia cuando el estudiante acepta participar, propone intentos de solución, retoma el trabajo tras un error y mantiene voluntad de completar el reto. La disposición también se expresa cuando el estudiante asume acuerdos del equipo y responde a la retroalimentación con acciones concretas, en lugar de detener el proceso. Esta medida permite interpretar la continuidad del esfuerzo como un componente distinto del interés y de la autoeficacia, lo que favorece una lectura más precisa de la motivación hacia actividades STEM (Ribarić et al., 2025).

CAPÍTULO III. METODOLOGÍA

3.1 Paradigma y enfoque que enmarcan el estudio

El estudio se ubicó en un paradigma pragmático con orientación constructivista, dado que privilegió decisiones metodológicas guiadas por su utilidad para responder la pregunta de investigación en el aula. En este trabajo, el pragmatismo se entiende como la selección de métodos según su capacidad para producir evidencia interpretable en un contexto escolar, con límites de control propios de la práctica educativa (Foster, 2024). El enfoque fue mixto, en tanto que combinó medición de variaciones antes y después, con comprensión cualitativa de procesos observables durante la intervención. Por ello, el análisis articuló resultados de pruebas y escalas (aspecto cuantitativo), con registros cualitativos de experiencia y dinámica de clase, de modo que la interpretación no dependiera de una sola fuente.

3.2 Diseño y métodos

El estudio adoptó un diseño cuasiexperimental pretest–posttest con enfoque mixto, apropiado para valorar cambios tras una intervención en contexto natural sin asignación aleatoria (Denny et al., 2023). El componente cuantitativo se orientó a comparar mediciones de entrada y salida vinculadas con pensamiento computacional y pensamiento sistémico, además de una prueba sobre motivación hacia STEM con ítems medidos en escala de Likert. Por otra parte, el componente cualitativo documentó procesos y significados en aula mediante observación y entrevista, con el fin de explicar condiciones, dinámicas y prácticas asociadas a los cambios reportados.

La organización metodológica mantuvo coherencia entre objetivos, fuentes y momentos de recolección. Dicha coherencia se sostuvo porque cada objetivo se vinculó con evidencias y momentos de aplicación definidos desde el inicio del estudio. El pretest y el posttest permitieron

describir variaciones en pensamiento computacional y pensamiento sistémico antes y después de la intervención, en correspondencia con el objetivo de analizar el nivel de desarrollo de estas habilidades. La encuesta de motivación aportó información sobre interés, autoeficacia, valor percibido y disposición frente a la experiencia, en relación con el objetivo orientado a valorar los alcances de la estrategia pedagógica. A su vez, la observación y la entrevista permitieron comprender cómo se desarrollaron los procesos de aula, qué dificultades surgieron y de qué manera el estudiantado resolvió los retos planteados. De esta manera, el componente cuantitativo ofreció medidas comparables de cambio y el componente cualitativo aportó elementos para interpretar esos resultados en su contexto de implementación, lo que mantuvo correspondencia entre la pregunta de investigación, los objetivos y la lógica del diseño mixto adoptado (Creswell, 2014).

3.2.1. Instrumentos

El estudio empleó cinco instrumentos articulados por propósito y momento de aplicación, con versiones completas en anexos, con el fin de obtener evidencia comparable de desempeño y evidencia descriptiva del proceso. En primer lugar, el pretest (ver Anexo A) correspondió a un cuestionario de 15 preguntas de selección múltiple diseñado para establecer la línea base en pensamiento computacional y pensamiento sistémico; su aplicación permitió obtener un punto de partida por dimensiones. En segundo lugar, el posttest (ver Anexo B) mantuvo 15 ítems cerrados comparables e incorporó cinco preguntas abiertas de desempeño, con el propósito de identificar variaciones posteriores a la intervención y recuperar explicaciones del procedimiento. En tercer lugar, la encuesta (ver Anexo C) de motivación hacia STEM utilizó escala tipo Likert y preguntas abiertas para describir actitudes y percepciones frente a la experiencia y obtener información sobre interés, autoeficacia y valor atribuido a las actividades. En cuarto lugar, la entrevista semiestructurada (ver Anexo D) se diseñó para profundizar en dificultades, aprendizajes declarados, preferencias de herramienta y sentido atribuido al trabajo con programación visual y

hardware interactivo. Finalmente, en quinto lugar, la guía de observación (ver Anexo E) organizó el registro de interacción en aula y permitió documentar trabajo en equipo, comunicación técnica, participación, uso de recursos y estrategias de resolución de problemas durante las sesiones. En conjunto, estos instrumentos permitieron valorar la intervención didáctica desde distintas fuentes, dado que registraron cambios en el desempeño, percepciones sobre la experiencia y dinámicas observables durante la resolución de retos con programación visual y *hardware* interactivo.

3.2.2 Validación de instrumentos

La validación de los cinco instrumentos descansa en dos pilares. Por un lado, la pertinencia teórica de los constructos que cada herramienta mide, anclada en la literatura especializada que dio origen a las versiones seminales. Por otro lado, la evidencia psicométrica que las fuentes originales reportan para cada escala o prueba, es decir, coeficientes de consistencia interna, cargas factoriales y correlaciones criterio. El equipo investigador tradujo los ítems al español cuando correspondía, ajustó el vocabulario al habla del corregimiento de Los Milagros y revisó la claridad semántica con base en lecturas comparativas entre las versiones publicadas. Las cuatro subsecciones siguientes ordenan la validación instrumento por instrumento. El pretest y el posttest se tratan de forma conjunta porque comparten origen, dimensiones y marco teórico.

3.2.2.1 Validación del cuestionario pretest y posttest de competencias STEM

El cuestionario pretest y su versión posttest adaptan ítems del Computational Thinking Test diseñado por Román-González et al. (2017) y del Beginners Computational Thinking Test propuesto por Zapata-Cáceres et al. (2020), dos pruebas consolidadas en la literatura sobre evaluación del pensamiento computacional escolar. Las tareas del concurso Bebras, descritas por Dagienė y Futschek (2008), aportaron ejemplos adicionales para la sección de reconocimiento de patrones. La sección A examina descomposición y abstracción, dimensiones que Wing (2006)

delimita como núcleo del pensamiento computacional y que Grover y Pea (2013) retoman en su revisión sobre K-12. La sección B, orientada al modelado y a la retroalimentación, articula la definición de pensamiento sistémico formulada por Arnold y Wade (2015) con los principios de estructura y comportamiento que postula Meadows (2008). La sección C, presente únicamente en el postest, incorpora cinco preguntas abiertas de desempeño siguiendo la tríada de conceptos, prácticas y perspectivas computacionales que plantean Brennan y Resnick (2012). Aristizábal Zapata et al. (2024) confirman la pertinencia de este tipo de tareas gráficas para primeros grados de primaria en el contexto hispano.

La literatura previa documenta indicadores psicométricos aceptables. Román-González et al. (2017) reportan un α de Cronbach entre 0.79 y 0.80 para el CTt, con estructura factorial coherente y correlaciones criterio con pruebas de aptitud general. Zapata-Cáceres et al. (2020) ponderan un α de 0.71 para el BCTt dirigido a escolares de primaria, con cargas factoriales que sustentan la validez de contenido de la prueba. El-Hamamsy et al. (2022) contrastan ambos instrumentos y delimitan rangos de dificultad diferenciados por grado escolar. Ahora bien, en el presente estudio el tamaño muestral reducido y la baja variabilidad de las respuestas impidieron replicar la estructura factorial reportada en la literatura original. Por ello, los resultados se interpretan de forma descriptiva, con porcentajes de acierto por dimensión y comparaciones pretest-postest, y una prueba inferencial no paramétrica de Wilcoxon-Pratt, mientras que los análisis factoriales confirmatorios quedan por fuera del alcance del estudio.

3.5.2 Validación de la escala Likert de motivación hacia STEM

La escala reúne 24 ítems distribuidos en cinco subescalas. La primera corresponde a interés con placer, la segunda a autoeficacia percibida, la tercera a utilidad y facilidad percibidas, la cuarta a trabajo en grupo con colaboración, y la quinta a pensamiento computacional. El diseño se apoyó

en el Student Attitudes toward STEM Survey desarrollado por Unfried et al. (2015), en el STEM Career Interest Survey de Kier et al. (2014) y en el Intrinsic Motivation Inventory adaptado por McAuley et al. (1989). Los ítems referidos a pensamiento computacional retoman la Computational Thinking Scale que Korkmaz et al. (2017) validaron inicialmente con estudiantes universitarios y que trabajos posteriores han ajustado a cohortes más jóvenes. La formulación lingüística privilegió enunciados breves en primera persona, con anclajes de cinco puntos, y el equipo investigador evitó expresiones idiomáticas ajenas al habla rural del Cauca.

Cada subescala reposa sobre una tradición teórica diferenciada. El bloque de interés con placer articula la teoría de la autodeterminación propuesta por Deci y Ryan (2000) con el modelo de cuatro fases del interés que formulan Hidi y Renninger (2006). Para la autoeficacia, Bandura (1997) expone el papel regulador de las creencias de control personal ante tareas escolares desafiantes. La utilidad y la facilidad percibidas remiten a la teoría expectativa-valor sistematizada por Eccles y Wigfield (2002). El componente colaborativo se sustenta en la teoría de interdependencia social examinada por Johnson y Johnson (2009), mientras que la subescala computacional recoge los constructos de Brennan y Resnick (2012). En cuanto a la consistencia interna, las subescalas originales reportan α de Cronbach entre 0.70 y 0.90, según documentan Unfried et al. (2015) y Korkmaz et al. (2017). La traducción siguió un procedimiento de doble revisión bilingüe por parte del equipo investigador, con atención especial a la equivalencia semántica y a la pertinencia del vocabulario al contexto rural colombiano. De manera similar a como ocurrió con los cuestionarios de desempeño, la baja variabilidad y el tamaño muestral reducido aconsejaron limitar el tratamiento psicométrico a estadísticos descriptivos.

3.5.3 Validación de la entrevista semiestructurada a estudiantes

La entrevista reúne diez preguntas organizadas en cuatro categorías de análisis. La primera categoría corresponde a pensamiento computacional, la segunda a pensamiento sistémico, la tercera a motivación y la cuarta a percepción de la intervención. El equipo investigador diseñó el guión siguiendo las pautas de Kvale (2011), quien delimita las fases de tematización, diseño, entrevista, transcripción, análisis, verificación y reporte como procedimientos esenciales del trabajo cualitativo. Flick (2015) aporta criterios sobre apertura conversacional, saturación temática y alcance de las preguntas en cohortes escolares de educación básica.

Las categorías conceptuales retoman las dimensiones del pensamiento computacional formuladas por Wing (2006), la tríada de conceptos con prácticas y perspectivas de Brennan y Resnick (2012) y los determinantes motivacionales postulados por Deci y Ryan (2000). La validez del instrumento se fundamenta en esa anclaje teórico y en la consistencia entre las categorías de análisis y las preguntas del guión. Patton (2015) recomienda que las entrevistas aplicadas a niños y adolescentes en contextos rurales se apoyen en vocabulario cercano a su experiencia cotidiana, principio que orientó el ajuste lingüístico del instrumento. La versión final conservó las diez preguntas originales con un orden que avanza desde lo concreto, como la descripción de lo que hicieron en clase, hacia lo reflexivo, como la justificación de sus decisiones de programación.

3.5.4 Validación de la guía de observación de clase

La guía organiza la observación cualitativa estructurada en torno a tres ejes. El primer eje recoge el pensamiento computacional, visible en la resolución de problemas y en la depuración de proyectos. El segundo eje captura el pensamiento sistémico, manifestado en el trabajo en equipo, el uso de recursos y la comunicación entre pares. El tercer eje documenta la motivación, inferida a partir de actitudes observables y niveles de participación durante las sesiones. Angrosino (2012)

describe la observación estructurada como herramienta idónea para registrar conductas contextuales mediante categorías predefinidas. Hernández-Sampieri et al. (2014) ofrecen criterios para construir rejillas que combinen descripción densa y valoración graduada. Zhou et al. (2023) aportan evidencia reciente sobre protocolos de observación STEM integrados en aulas K-12, cuyas categorías inspiraron la delimitación operacional de los indicadores del instrumento.

La validez del instrumento se sustenta en la correspondencia entre las categorías observacionales y los constructos teóricos que el estudio aborda, siguiendo el principio de validez de contenido descrito por Rodríguez Medina et al. (2021). Cada indicador se formuló con descriptores operacionales que permiten distinguir presencia, ausencia o nivel de manifestación de la conducta. El equipo investigador realizó una sesión de pilotaje previa a la intervención, que permitió afinar el vocabulario observacional, estimar los tiempos de registro y depurar la redundancia entre descriptores. La versión definitiva rigió el registro a lo largo de las sesiones de la secuencia didáctica y se complementó con un campo abierto para notas de campo que recogieran matices no previstos por la rejilla (ver Anexos A, B, C, D y E).

3.3 Población, muestra y escenario escolar

La población de interés correspondió a estudiantes de grado sexto de la Institución Educativa Agropecuaria José Dolores Daza, en quienes se implementó la integración de programación visual y hardware interactivo. La muestra se definió como un grupo intacto seleccionado por conveniencia y pertinencia curricular, coherente con un diseño cuasiexperimental sin asignación aleatoria, con criterios de inclusión vinculados con matrícula vigente, asistencia regular y consentimiento o asentimiento, y criterios de exclusión asociados con inasistencia sostenida o datos incompletos. El trabajo de campo se desarrolló en el entorno escolar habitual, con espacios que facilitaron colaboración y manipulación segura de dispositivos.

3.4 Estructura de la intervención didáctica

Este apartado delimita el trayecto formativo aplicado con los estudiantes de sexto grado de la Institución Educativa Agropecuaria José Dolores Daza, del corregimiento de Los Milagros. La intervención articuló las nueve sesiones con una carga total de once horas de trabajo en el aula, repartidas entre programación visual y montaje físico. El diseño antepone la acción del estudiante a la exposición magistral y ordena el avance desde la pantalla hacia el circuito construido, el producto integrador fue un semáforo funcional controlado por microcontrolador.

3.4.1 Descripción general de la intervención

La secuencia reúne nueve sesiones con una duración agregada de once horas, organizadas en dos fases consecutivas que avanzan de lo virtual a lo tangible. La primera fase recurre a *Scratch* como entorno de programación por bloques; la segunda articula *Flowcode* con el montaje en una *board* gobernada por el microcontrolador PIC16F73, el enfoque metodológico descansa en el aprendizaje activo y práctico, con talleres guiados y producción situada de artefactos. Para Bybee (2013), la integración STEM se concreta cuando los estudiantes enfrentan problemas que exigen conjugar ciencia, matemática e ingeniería en un mismo objeto; de ahí que el semáforo operara como eje articulador con un artefacto reconocible del entorno rural que condensa lógica secuencial, electrónica y razonamiento temporal. Esta decisión se fundamenta en estudios como el de Gavari et al. (2024), en el que se afirma que los proyectos STEM anclados al territorio incrementan la implicación del alumnado en zonas rurales.

3.4.2 Fase 1. Programación visual con Scratch

La primera fase abre con el reconocimiento de la interfaz con el escenario, paleta de bloques y objetos programables, los estudiantes manipulan bloques de movimiento, control y eventos mediante actividades guiadas, lo que introduce la noción de secuencia y de instrucción

condicional. Resnick (2017) sostiene que los entornos por bloques reducen la carga sintáctica y permiten que el aprendiz concentre su atención en la lógica del programa. El producto final consiste en un semáforo digital con cambios secuenciales rojo, amarillo y verde acompañado por un personaje que simula el tránsito y reacciona a las señales. Fagerlund et al. (2021) evidencia que este tipo de proyectos en *Scratch* favorece la comprensión de iteraciones y condicionales en estudiantes de primaria. Al respecto, Brennan y Resnick (2012) identifican en esas construcciones tres dimensiones del pensamiento computacional como los conceptos, prácticas y perspectivas. La actividad encaja con esa tríada al exigir anticipación, depuración y rediseño.

3.4.3 Fase 2. Programación con Flowcode y montaje en hardware

La segunda fase introduce *Flowcode* como entorno basado en diagramas de flujo. En esta fase, los estudiantes reconocen el panel de componentes, el área de simulación y la barra de herramientas, y resuelven ejercicios progresivos al encender un LED, temporizar encendidos y apagados y coordinar varios LED simultáneos. Wing (2006) argumenta que la construcción de soluciones computacionales supone descomposición, abstracción y pensamiento algorítmico: tres operaciones presentes en cada ejercicio. Tras la simulación, el grupo diseña el semáforo y selecciona el microcontrolador PIC16F73, asigna los pines de salida y estructura el bucle de ejecución. El montaje físico traslada el programa a la *board*; los LED rojo, amarillo y verde se conectan con resistencias limitadoras, y el código se carga desde *Flowcode* para verificar el comportamiento. Para Grover y Pea (2013), cerrar el ciclo con una ejecución real aporta retroalimentación que la simulación aislada no entrega. Jiang y Li (2021) reportan, en la misma dirección, que la transición del bloque al hardware mejora la persistencia y el razonamiento algorítmico en niños de educación básica.

3.4.4 Materiales, recursos y componentes electrónicos

Los recursos empleados comprenden computadores, *videobeam*, *protoboard* y la *board* de trabajo que aloja el microcontrolador. El *software* incluyó *Scratch* para la fase inicial, *Flowcode* para la programación del PIC y *Proteus* para la simulación previa del circuito. La Tabla 1 organiza los componentes electrónicos definidos en el diseño realizado en *Proteus*, con su identificador, cantidad y función en el circuito.

Tabla 1

Componentes electrónicos del semáforo diseñado en Proteus

Identificador	Componente	Cantidad	Función en el circuito
C1–C2	Condensador cerámico de 22 pF	2	Estabilización del cristal oscilador
R1–R3	Resistencia de 1 kΩ	3	Limitación de corriente para los LED
U1	Regulador de voltaje 7805	1	Entrega de 5 V regulados al microcontrolador
U2	Microcontrolador PIC16F73	1	Unidad de control del semáforo
D1–D3	LED rojo, LED amarillo y LED verde	3	Señalización secuencial del tráfico
BAT1	Batería de 9 V	1	Fuente primaria de alimentación
X1	Cristal oscilador	1	Referencia de frecuencia para el PIC

Nota. Elaboración propia con base en el esquema implementado en Proteus durante la fase de diseño del circuito.

3.4.5 Sustento teórico de las decisiones de diseño

El diseño descansa en el construccionismo de Papert (1980), que concibe el aprendizaje como construcción pública de artefactos con los que el estudiante piensa al tiempo que fabrica. Harel y Papert (1991) amplían el argumento al sostener que el saber se afianza cuando el aprendiz produce objetos compartibles con otros. El semáforo funcional cumple esa condición es observable, depurable y defendible ante pares. La progresión por fases recoge la tradición del método de proyectos, que Kilpatrick (1918) formuló como actividad intencional con propósito, y que Thomas (2000) consolidó como dispositivo didáctico centrado en tareas auténticas. Un meta-

análisis de Zhang y Ma (2023) confirma efectos positivos del aprendizaje basado en proyectos sobre el rendimiento y las habilidades del siglo XXI. Y la programación por bloques actúa como andamiaje cognitivo previo al código textual, hallazgo respaldado por Molina-Ayuso et al. (2024) en estudiantes de sexto de primaria. Brennan y Resnick (2012) aportan la rejilla conceptual para leer el avance de los aprendices. Para la integración disciplinar, Bybee (2013) provee el marco STEM, complementado por National Academy of Engineering y National Research Council (2014). La trayectoria de Proteus a la board materializa el tránsito de la simulación a la experiencia física, recorrido que Grover y Pea (2013) recomiendan para consolidar el pensamiento computacional en educación básica.

(ver Anexo F – Evidencias de la implementación y Anexo G – Cronograma de intervención didáctica)

3.5 Fases de la investigación

La investigación se desarrolló en cuatro fases articuladas. La fase de diagnóstico y línea base aplicó el pretest como punto de partida y registró condiciones iniciales del grupo, con el fin de identificar el nivel de entrada en pensamiento computacional y pensamiento sistémico. Esta primera fase permitió reconocer necesidades del aula y ajustar la planificación de la intervención de acuerdo con el contexto institucional, la disponibilidad de recursos y las características del estudiantado.

La segunda fase correspondió al diseño y ajuste de la intervención didáctica, la secuencia se planeó con criterios de logro definidos, consignas estandarizadas y momentos de acompañamiento pedagógico orientados a favorecer comprensión, participación y ajuste progresivo de las soluciones. La tercera fase correspondió a la implementación de la intervención en sesiones regulares dentro del horario escolar. Esta fase convirtió el aula en un espacio de diseño, prueba,

corrección y reflexión, donde la intervención didáctica promovió comunicación técnica, coordinación entre pares y resolución situada de problemas.

La cuarta fase correspondió al cierre y análisis de la información recolectada. En este momento se aplicó el posttest, se administró la encuesta de motivación y se recolectó información derivada de entrevistas, observaciones y respuestas abiertas. En el componente cuantitativo, el estudio calculó estadísticos descriptivos y contrastó cambios entre pretest y posttest mediante pruebas pareadas acordes con las características de los datos. En el componente cualitativo, la información se organizó mediante codificación abierta y axial para identificar patrones vinculados con la ejecución de los retos, la interacción entre estudiantes y la respuesta frente al error. La integración final relacionó evidencias numéricas y narrativas para sostener una interpretación más completa sobre los alcances de la intervención didáctica en el aprendizaje STEM.

3.6 Metodología de interpretación de resultados

Los resultados del pretest y posttest de habilidades STEM, de la prueba de motivación y de los registros cualitativos requirieron técnicas cuantitativas y cualitativas de análisis. Este proceso involucró procesamiento y exploración descriptiva en Excel y aplicación de técnicas inferenciales en el lenguaje de código abierto R (v. 4.5.0). En Excel se realizó la consolidación de resultados a diferentes niveles de resumen y las gráficas presentadas en este trabajo.

En R, los resultados pretest y posttest se compararon por cada habilidad STEM indagada, descomposición, abstracción, modelado y retroalimentación, con una prueba de signos de Wilcoxon-Pratt de datos pareados bajo el supuesto de que las preguntas para cada habilidad en el pretest y posttest tenían una dificultad comparable. Esta prueba se hizo con el paquete **coin**, versión 1.4.3. La prueba de Wilcoxon-Pratt es una prueba no paramétrica con pocos supuestos, como independencia e identidad de los pares, y evalúa si el cambio es simétrico o no respecto a cero, razón por la cual su alcance de aplicabilidad incluye la situación de este trabajo (Conover, 1999).

También se evaluó la posibilidad de realizar análisis factorial confirmatorio (Finch y French, 2015), tanto con los ítems de la prueba de motivación como con las pruebas de habilidades STEM, mediante el paquete **psych**, versión 2.5.3. Respecto a esto último, la muestra fue tan pequeña y la variabilidad de respuesta tan baja que fue imposible obtener confirmación estadística de la validez de la agrupación en factores de los ítems; debido a esto, este aspecto no se discutirá demasiado en los resultados.

El tratamiento cualitativo siguió un procedimiento en tres fases apoyado en el método de análisis temático que formulan Braun y Clarke (2006). La primera fase consistió en la familiarización con el corpus a través de lecturas reiteradas de las transcripciones de entrevista y de los registros de la guía de observación. Durante la segunda fase se realizó una codificación manual de orientación deductivo-inductiva: los códigos deductivos derivaron de las tres categorías que orientaban el estudio junto con el pensamiento computacional, el pensamiento sistémico y la motivación; y los códigos inductivos emergieron del contacto directo con el material. La tercera fase se agrupó en temas con sentido interpretativo, tal como recomienda el trabajo de Saldaña (2021) sobre ciclos de codificación. Para asegurar la validez de las interpretaciones, el equipo aplicó triangulación metodológica entre los datos cuantitativos del pretest-posttest y la escala Likert, las transcripciones de la entrevista semiestructurada y los registros de observación, siguiendo el criterio que postula Flick (2015). Las divergencias entre fuentes se discutieron hasta alcanzar consensos razonados, y los extractos textuales que ilustran cada tema

Esta ruta fue pertinente para el caso porque la investigación no buscó únicamente establecer variaciones entre el pretest y el posttest. También se buscó comprender cómo el grupo resolvió los retos, cómo reaccionó ante el error y qué sentido atribuyó al trabajo con programación visual y hardware interactivo. En este sentido, Creswell (2014) sostiene que, en los estudios con enfoque

mixto, el análisis cualitativo amplía la interpretación de los resultados numéricos al situarlos en su contexto de ocurrencia. En esa línea, la triangulación entre entrevista, observación y respuestas abiertas permitió interpretar los cambios cuantitativos a la luz de las dinámicas de aula, de la mediación docente y de las experiencias construidas por el estudiantado durante la intervención.

3.7. Criterios de calidad y consideraciones éticas

El componente ético contempló consentimiento informado y asentimiento, anonimización de datos y resguardo de archivos con acceso restringido (ver Anexo H). El trabajo con *hardware* incluyó supervisión docente y protocolos de seguridad durante la manipulación, con organización de turnos para favorecer participación equilibrada.

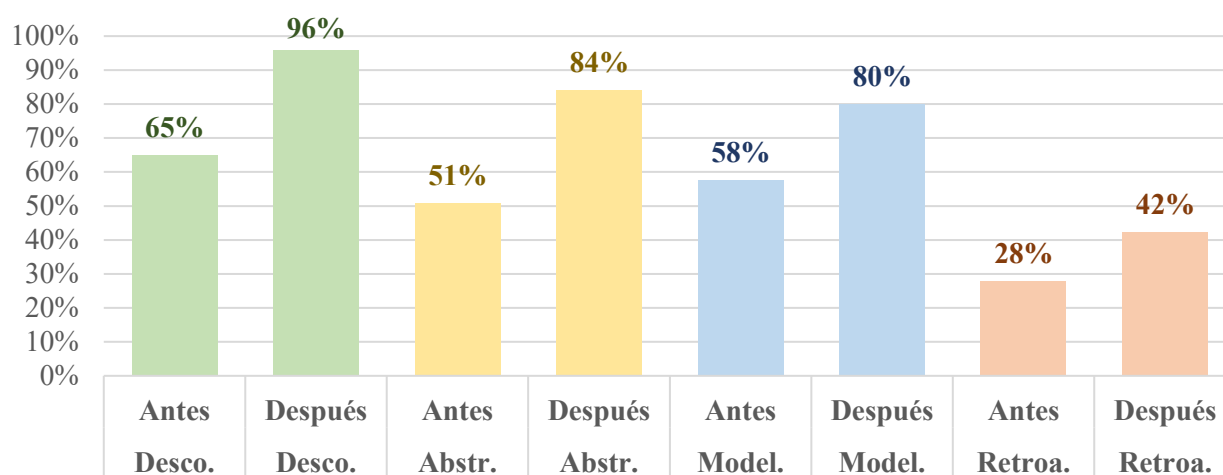
CAPÍTULO IV. ANÁLISIS Y DISCUSIÓN DE RESULTADOS

4.1. Resultados cuantitativos

4.1.1. Resultados pretest–postest de habilidades STEM

Figura 1

Porcentaje de acierto en las preguntas de prueba de habilidades STEM antes y después de la intervención didáctica.



Nota. Elaboración propia a partir de los resultados del pretest y postest aplicados a 30 estudiantes de grado sexto. Los porcentajes representan la proporción de respuestas correctas obtenidas en las habilidades de descomposición, abstracción, modelado y retroalimentación antes y después de la intervención didáctica.

En la figura 1 se resumen los porcentajes de acierto por habilidad tanto en el pretest como en el postest. Este porcentaje resulta de dividir la suma de las respuestas correctas de todos los estudiantes respecto al total de respuestas correctas posibles. Como la muestra fue de 30 estudiantes y las primeras tres habilidades (descomposición, abstracción y modelado) tenían cuatro preguntas

en cada test, estas tres habilidades tenían cada una 120 respuestas correctas posible en cada test. En cambio, la habilidad de retroalimentación tenía tres preguntas por test; por ende, tenía 90 respuestas correctas posibles por test. Se aprecia que en todas las habilidades hubo una mejora en el porcentaje de acierto después de la intervención didáctica, en particular en la habilidad de descomposición.

El mayor rendimiento de la descomposición, incluso desde el pretest, sugiere que esta habilidad partía de una base más próxima a las formas escolares de resolver tareas por pasos. En este sentido, Wing (2006) la define como la posibilidad de dividir un problema en partes manejables, y esa operación parece haber sido más accesible para el grupo durante la intervención. Esto puede explicarse porque los retos con programación visual exigían identificar subobjetivos, decidir una secuencia y verificar cada paso antes de continuar. En esa línea, Stolpe y Hallström (2023) sostienen que la programación por bloques facilita la comprensión de conceptos al centrar la atención en la lógica del procedimiento y en su orden. Por otro lado, Kanobel et al. (2019) vinculan el trabajo colaborativo con mejores condiciones para comprender tareas STEM. Por ello, la descomposición parece haber funcionado como un punto de apoyo inicial para el aprendizaje con programación y *hardware*, mientras las otras habilidades exigieron relaciones menos visibles y decisiones de mayor integración. Esa diferencia abre la discusión sobre las demandas que luego aparecieron en modelado y retroalimentación, donde el estudiantado tuvo que coordinar simultáneamente relaciones, ajustes y efectos del sistema.

Como las preguntas eran de selección múltiple de cuatro opciones con única respuesta correcta, el porcentaje de acierto medio por puro azar es del 25%, lo que significa que la gran mayoría de porcentajes de acierto están bastante por encima del umbral que se explica por puro azar, incluso en el pretest. Sin embargo, hay una excepción: la habilidad de retroalimentación

presentó un porcentaje de acierto en el pretest comparable con el umbral de puro azar, e incluso su porcentaje de acierto en el posttest no supera el 50%.

El menor dominio de la retroalimentación sugiere que esta habilidad exigió una operación cognitiva distinta a seguir una secuencia o reconocer partes del problema. En esa línea, Davis et al. (2023) plantean que el pensamiento sistémico se vuelve más exigente cuando la tarea obliga a decidir qué cambio hacer y por qué, a partir de la interacción entre componentes. En la intervención, esa demanda apareció cuando el estudiantado tuvo que interpretar fallos del programa, ordenar ciclos, conectar circuitos y verificar la respuesta del hardware. En ese punto, el trabajo ya no dependía solo de ejecutar instrucciones, dependía de explicar discrepancias y ajustar la solución. Bajo esta lógica, el hardware *interactivo* ofreció señales observables para revisar el funcionamiento, pero esa visibilidad no garantizó por sí sola una comprensión estable de las relaciones entre entrada, proceso y salida (Guanotuña et al., 2024).

A ello se suma que la secuencia didáctica introdujo primero tareas de programación más directas y luego el trabajo con *Flowcode* y la baquela. Por esa razón, la retroalimentación quedó asociada a momentos de mayor complejidad técnica y menor margen de práctica. Por eso, el avance observado puede leerse como una aproximación inicial a la lógica de prueba y corrección, aunque todavía con dificultades para justificar causalmente los cambios realizados. Esta tensión permite entender que la intervención abrió una ruta de aprendizaje en pensamiento sistémico, pero también mostró que la retroalimentación requiere más tiempo de ejercitación y mediaciones más explícitas que otras habilidades.

Con el propósito de complementar la lectura descriptiva de los porcentajes de acierto, la tabla 2 presenta los resultados de la prueba de Wilcoxon-Pratt aplicada a las cuatro habilidades STEM evaluadas. Este contraste permitió examinar si las diferencias observadas entre el pretest y el

postest correspondieron a variaciones estadísticamente significativas y no solo a cambios aparentes en la distribución de respuestas. De esta manera, la tabla 2 aporta un sustento inferencial al análisis cuantitativo del estudio y permite valorar con mayor precisión el alcance de la mejora reportada tras la intervención didáctica en descomposición, abstracción, modelado y retroalimentación.

Tabla 2

Resultados de la prueba de Wilcoxon-Pratt sobre la mejora en los resultados de la prueba de habilidades STEM.

Habilidad	Estadístico Z	Valor p
Descomposición	-4,447	$4,77 \times 10^{-7}$
Abstracción	-4,035	$7,51 \times 10^{-6}$
Modelado	-2,528	$5,83 \times 10^{-3}$
Retroalimentación	-2,502	$6,85 \times 10^{-3}$

Nota. Elaboración propia con base en los resultados de la prueba de Wilcoxon-Pratt aplicada a las habilidades STEM evaluadas en el pretest y el posttest.

Aunque los resultados de porcentaje de acierto conjuntos ya sugieren una fuerte evidencia de que la intervención didáctica produjo una mejora en el nivel de comprensión en las cuatro habilidades STEM evaluadas, es necesario establecer una medida de la significación estadística de estas conclusiones. Para tal efecto se hizo una prueba de hipótesis tipo signos de Wilcoxon-Pratt. Dentro de los supuestos de la prueba se asume que la dificultad de las preguntas en cada habilidad en el pretest y posttest era comparable, al menos en promedio de las tres o cuatro preguntas de cada habilidad. También se asumió que cada estudiante correspondía a un elemento muestral idéntico e independiente: una suposición bastante razonable puesto que los test fueron individuales y la población que se tiene en mente son estudiantes con características análogas a los que participaron en el estudio.

La prueba de hipótesis fue pareada y solo de signos, lo que significa que consistió en averiguar por cada estudiante y cada habilidad si el número de preguntas correctas después del test aumentó, disminuyó, o quedó igual (manejo de empates). La prueba fue de signos porque no detecta la cantidad de preguntas correctas en las que se mejoró o no (pueden ser 1, 2, 3 o 4), solo detecta si mejoró, empató o disminuyó. La hipótesis nula consistió en asumir que el número de preguntas tenía mayor probabilidad de ser peor después o de permanecer igual (prueba de una cola). La tabla 2 muestra los resultados de la prueba y se concluye que, para cada una de las cuatro habilidades STEM, se rechaza la hipótesis nula y se muestra la probabilidad de equivocarse en esa conclusión (valor p). Los valores p fueron muy bajos, lo que implica una alta significación en la conclusión de que hubo una mejora en el desempeño de las pruebas. Esta significación sugiere que la conclusión es robusta incluso bajo la presencia de ruido o de errores de medición sistemáticos no deseados (como por ejemplo que las preguntas en el postest fueran algo más fáciles). En otras palabras, lo más probable es que la intervención didáctica sí haya ayudado a mejorar notoriamente la comprensión de los estudiantes en las habilidades STEM evaluadas.

Dado que los resultados se agruparon en cuatro grupos de preguntas correspondientes a cuatro constructos teóricos distintos, habría sido deseable confirmar estadística esta estructura teórica mediante análisis factorial. Sin embargo, esto no fue posible por las limitaciones en tamaño muestral y variabilidad de los resultados. Esto no significa que se haya encontrado evidencia estadística contraria a la validez de estos constructos; por el contrario, lo más probable es que se trate de un error tipo 2, de potencia estadística, o falso negativo (Finch y French, 2025).

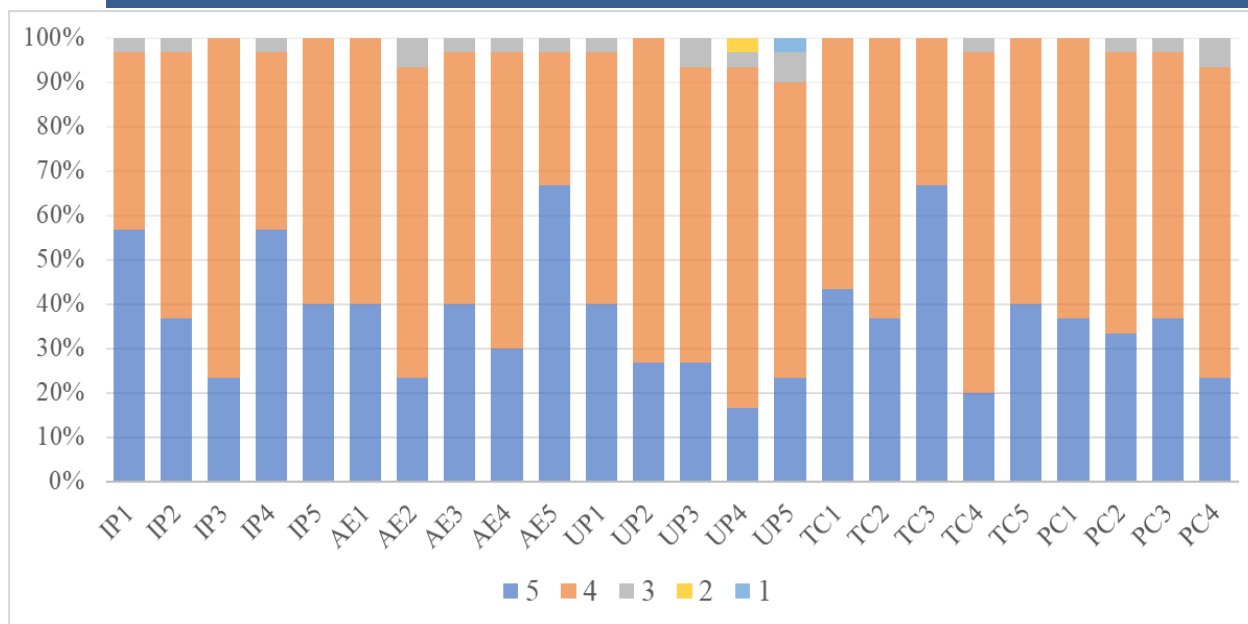
La imposibilidad de confirmar estadísticamente la estructura de los constructos exige leer los puntajes como indicadores de cambio en el desempeño y no como prueba concluyente de una organización interna estable de los instrumentos. En ese sentido, Ribarić et al. (2025) recuerdan

que la interpretación por dimensiones en escalas STEM requiere bases psicométricas suficientes para sostener la agrupación de ítems, condición que aquí quedó restringida por el tamaño de la muestra. A la vez, este límite no anula el valor analítico del estudio, porque Guggemos et al. (2023) sostienen que las pruebas de desempeño permiten identificar manifestaciones del pensamiento computacional aun cuando la validación estructural de los instrumentos deba asumirse con reserva. Bajo esa misma lógica, Davis et al. (2023) señalan que las tareas de desempeño, la observación y otras fuentes de información capturan facetas distintas de habilidades complejas, de modo que la triangulación resulta pertinente cuando una sola medida no basta para explicar el proceso. Por ello, la cautela estadística no debilita por completo los hallazgos, pero sí desplaza su fuerza interpretativa hacia la convergencia entre pruebas, entrevista, observación y respuestas abiertas, donde el aprendizaje aparece ligado a la manera en que el estudiantado enfrentó, corrigió y reajustó los retos propuestos.

4.1.2. Resultados prueba de motivación

Figura 2

Respuestas de los estudiantes a los ítems de la prueba de motivación en escala de Likert.



Nota. Elaboración propia a partir de las respuestas de 30 estudiantes de grado sexto en la prueba de motivación aplicada al finalizar la intervención didáctica. La figura muestra la distribución porcentual de las respuestas en escala de Likert para cada ítem del instrumento.

En la figura 2 se presenta un resumen de los resultados de la prueba de motivación donde se aprecia que la gran mayoría de los estudiantes respondieron estar de acuerdo con los ítems en un nivel de Likert 4 o 5. Esto es un indicio de alta motivación, pues el mayor nivel de Likert implica directamente una autopercepción de sentirse motivado hacia el aprendizaje de las competencias STEM centrales de este estudio en algún aspecto concreto. Los siguientes ítems destacan por tener más del 50% de respuestas en 5: AE5 (poder corregir errores), TC3 (justicia en la distribución del trabajo en equipo), IP4 (interés en seguir explorando) e IP1 (placer en participar en actividades como esta).

El peso que alcanzaron el placer por participar, el interés por seguir explorando, la posibilidad de corregir errores y la percepción de justicia en el trabajo compartido sugiere que la motivación del grupo se sostuvo en una experiencia donde aprender dependió de hacer, revisar y cooperar. En esa dirección, Ribarić et al. (2025) plantean que la motivación hacia STEM se expresa cuando el

estudiante encuentra sentido en la tarea y mantiene disposición para continuar, incluso frente a dificultades. Junto a ello, Kanobel et al. (2019) señalan que las dinámicas colaborativas favorecen la implicación cuando cada integrante percibe que su aporte cuenta dentro del proceso. Este hallazgo permite pensar que la intervención funcionó mejor cuando articuló reto, corrección y trabajo compartido, aunque esa disposición pudo ser menos estable en momentos donde la tarea exigía mayores niveles de comprensión técnica. Desde ahí, la motivación observada no remite solo al gusto por la actividad, también remite a una forma de participación sostenida que ayuda a entender los matices del desempeño posterior.

En cambio, los ítems UP4 (facilidad de aprender *flowcode*) y UP5 (la interfaz de *flowcode* es comprensible) tienen al menos un estudiante con respuesta menor que 3 y menos del 25% de los estudiantes que responden en 5. La menor valoración de la facilidad de *flowcode* sugiere que la experiencia resultó motivante porque combinó interés con un grado de exigencia que obligó a comprender, probar y corregir, y no porque redujera la tarea a una interacción inmediata. En esa línea, Ribarić et al. (2025) permiten leer este patrón como una motivación que se sostiene aun cuando la actividad demanda esfuerzo, siempre que el estudiante perciba utilidad y mantenga disposición para continuar. Por su parte, Vygotsky (1978) plantea que el aprendizaje cobra fuerza cuando la tarea exige avanzar con mediación y apoyo, de modo que cierta dificultad puede favorecer la construcción de conocimiento.

Este matiz ayuda a entender que la intervención funcionó cuando el reto técnico activó curiosidad y persistencia, aunque ese equilibrio podía debilitarse si la complejidad de la herramienta superaba las ayudas disponibles. Desde ahí, la dificultad percibida en *flowcode* no contradice la motivación observada, sino que aporta una lectura más precisa sobre las condiciones

pedagógicas que hicieron posible sostenerla. Esto indica que algunos estudiantes no percibieron la actividad como fácil y la gran mayoría no la percibieron como demasiado fácil.

Esta percepción de dificultad controlada puede leerse como una condición formativa del aprendizaje, pues Vygotsky (1978) vincula el avance cognitivo con tareas que exigen mediación y esfuerzo, mientras Ribarić et al. (2025) permiten entender que la motivación puede sostenerse cuando el reto mantiene interés sin convertirse en bloqueo. Hay que aclarar que las respuestas 3 o inferiores se concentraron fundamentalmente en apenas dos estudiantes cuyo promedio de respuesta fue menor que 4, lo cual es natural e indica una pequeña proporción de estudiantes que se sintió más frustrada y, posiblemente debido a esa frustración, menos motivada.

4.2 Resultados cualitativos

4.2.1. *Resultados de entrevistas y observación*

La información cualitativa permitió examinar cómo el estudiantado vivió la intervención didáctica y cómo se manifestaron, en la práctica de aula, los procesos asociados con pensamiento computacional, pensamiento sistémico y motivación. A partir de las entrevistas y de los registros de observación, el análisis organizó patrones de acción, dificultades recurrentes, formas de interacción y sentidos atribuidos a la experiencia de trabajo con programación visual y *hardware* interactivo. Esta lectura buscó complementar los resultados cuantitativos del capítulo, porque permitió interpretar si hubo cambios, y también cómo se construyeron durante la resolución de los retos, la comunicación entre pares, la respuesta frente al error y la mediación desarrollada en el aula.

4.2.1.1. Pensamiento Computacional (PC). Las respuestas de la entrevista sobre resolución de problemas describen el procedimiento como una secuencia de pasos que guía la acción. En la subcategoría descomposición, varios estudiantes mencionaron pensar paso a paso, ordenar lo primero y lo siguiente, y ubicar el error antes de cambiar el programa. Esa forma de resolver incluye persistencia ante fallos, con referencias a no rendirse (resiliencia), corregir errores y volver a intentar hasta lograr que el sistema responda. La comprensión declarada de programas y dispositivos refuerza este patrón, porque el estudiantado alude al orden de instrucciones y al rol de cada parte del proyecto. El registro de observación coincide con esta ruta al mostrar planificación previa, prueba de opciones y ajustes tras errores, lo que prepara la lectura de la abstracción en el discurso.

En la subcategoría abstracción, la entrevista muestra que el estudiantado recuerda las actividades por lo que el sistema hacía, como mover personajes, prender luces o responder a instrucciones. Varias respuestas justifican la preferencia por *scratch* o por *flowcode* al hablar de un manejo más sencillo o de un circuito que parecía real, sin detallar rasgos decorativos. Cuando el instrumento pregunta por aprendizajes nuevos, aparecen referencias a programar con bloques, usar ciclos y repeticiones, dar instrucciones claras y controlar dispositivos desde el computador. Algunos estudiantes señalan la corrección de errores y el ensayo como parte del aprendizaje, junto con la idea de crear proyectos sin escribir código difícil. La observación en aula refuerza esta síntesis al registrar uso de bloques, secuencias y repeticiones, y relación entre la instrucción programada y la respuesta del hardware, puente hacia el análisis.

El hallazgo indica que el pensamiento computacional se expresó, sobre todo, como organización de pasos y depuración localizada durante la ejecución de proyectos. Este patrón sugiere que el grupo transformó consignas en procedimientos verificables, lo cual favorece avances

cuando la tarea exige decisiones observables y producto final. En este sentido, Bybee (2013) describe STEM como trabajo orientado a competencias aplicadas, por lo que la secuenciación y la revisión del error ganan centralidad en este tipo de experiencias. La preferencia por pensar paso a paso coincide con la lógica de la programación visual, donde el orden de bloques hace explícita la dependencia entre acciones (Stolpe y Hallström, 2023). La ausencia de explicaciones detalladas en algunos relatos indica que la abstracción se consolida cuando la tarea ofrece pistas claras, y pierde precisión cuando exige justificar la selección de variables. La OECD (2019) plantea que esta diferencia implica ajustar actividades para que el estudiante explicita criterios de pertinencia y contraste alternativas antes de programar la solución definitiva. Esta lectura prepara el análisis del pensamiento sistémico, porque los mismos procedimientos deben conectarse con relaciones entre componentes y con decisiones de control del prototipo.

Wing (2006) delimita el pensamiento computacional como un proceso cognitivo que articula descomposición, abstracción, reconocimiento de patrones y diseño algorítmico al servicio de la resolución de problemas. En el presente estudio se evidenció esa arquitectura cognitiva a través de dos indicadores. Primero, las ganancias porcentuales entre pretest y posttest superaron los veinte puntos en los ítems de identificación de subproblemas, con valores que pasaron del 42 % al 68 % de acierto, y alcanzaron mejoras similares en los reactivos de abstracción, donde los estudiantes debían describir el comportamiento del semáforo sin detallar cada línea de código. Segundo, la guía de observación registró, desde la cuarta sesión, prácticas espontáneas de reutilización de bloques y ensayo iterativo en Scratch. La conexión con la teoría resulta directa: los estudiantes comenzaron a operar con las unidades cognitivas que Wing describe, aunque de modo incipiente y con andamiaje docente. Brennan y Resnick (2012) distinguen entre conceptos, prácticas y perspectivas computacionales; el estudio confirmó las dos primeras y dejó al descubierto la

debilidad de la tercera, pues las verbalizaciones reflexivas aparecieron de forma apenas tenue durante las entrevistas

4.2.1.2. Pensamiento Sistémico (PS). Las respuestas sobre trabajo con compañeros describen el modelado como una organización social del sistema de tarea, donde el grupo comparte ideas, reparte acciones y acuerda qué hacer primero. En la subcategoría trabajo en equipo, el estudiantado menciona apoyo mutuo y explicación entre pares, con referencias a turnos para usar el computador y a búsqueda conjunta de soluciones. En uso de recursos, las sugerencias de mejora apuntan a más tiempo, un computador por estudiante y menos fallas técnicas, junto con pedidos de equipos y espacios adecuados. El registro de observación complementa esta mirada al documentar roles diferenciados, como programar, observar, sugerir y verificar errores, y al mostrar manejo cuidadoso de computadores y dispositivos compartidos. Con esta organización de relaciones y recursos, el análisis de retroalimentación permite leer cómo el grupo ajustó decisiones según la respuesta del prototipo y según la comunicación en aula.

En retroalimentación, la entrevista muestra un tránsito desde confusión o nerviosismo inicial hacia satisfacción cuando el programa funciona, y varias respuestas vinculan ese cambio con probar y corregir. Las dificultades reportadas abarcan abrir o manejar *flowcode*, ordenar bloques en ciclos, conectar circuitos y recordar secuencias, lo que ubica puntos de ajuste para futuras sesiones. La motivación declarada descansa en ver resultados del proyecto, crear juegos, trabajar con el computador y repetir intentos, con menciones puntuales al acompañamiento docente y al trabajo con pares. En comunicación, el registro de observación describe intercambio oral sobre procedimientos, escucha a compañeros y respeto por la palabra, y ese soporte acompaña la toma de decisiones durante la programación. Este ciclo de prueba, conversación y ajuste deja un insumo

para discutir cómo el pensamiento sistémico se construyó al articular la respuesta del *hardware* con acuerdos del grupo.

El hallazgo muestra que el pensamiento sistémico se manifestó en roles coordinados, uso de recursos compartidos y ajustes del programa según la respuesta del *hardware*. Este resultado se relaciona con lo mencionado por los autores Guanotuña et al. (2024) quienes explican que el *hardware* interactivo convierte decisiones de programación en señales observables, lo que facilita revisar conexiones y corregir secuencias en tiempo real. Bajo esta condición, la cooperación descrita por el grupo tiene relación con actividades tipo reto, donde el resultado depende de acuerdos y pruebas compartidas (Feijoo y Rodríguez, 2022). Las solicitudes de más tiempo y de equipos por estudiante indican adaptación a restricciones de aula, lo que limita la exploración de escenarios alternativos. Por su parte, Marín et al. (2023) advierten que la implementación STEM en Colombia exige estrategias situadas ante limitaciones de recursos, y esa advertencia ayuda a interpretar estas demandas como condición pedagógica. Este análisis conduce a revisar la motivación, porque la persistencia y el interés se sostienen cuando el grupo percibe que puede avanzar pese a errores y limitaciones del entorno.

Arnold y Wade (2015) postulan que el pensamiento sistémico exige reconocer relaciones de causalidad, límites del sistema y circuitos de retroalimentación. En el presente estudio se evidenció un avance sólo parcial en esa dirección. Por un lado, los estudiantes modelaron el semáforo como sistema con entradas y salidas, y seis de los diez entrevistados produjeron expresiones equivalentes a “si cambio el tiempo, cambia el ciclo” o “si quito la resistencia el LED se quema”, frases que delatan una primera intuición sobre relaciones causales. Por otro lado, al solicitar justificaciones del porqué de un fallo o ajustes a partir del error, las respuestas derivaron hacia descripciones del objeto, sin explicar el comportamiento del programa. Las ganancias pretest-postest en los ítems de

retroalimentación no superaron los diez puntos porcentuales. Grover y Pea (2013) advierten justamente sobre esta brecha: la verbalización argumentada suele rezagarse respecto al desempeño procedimental. El hallazgo matiza la propuesta teórica al mostrar que, en sexto grado, el pensamiento sistémico emerge antes como gesto operativo que como explicación articulada.

4.2.1.3. Motivación. En el plano de actitudes, la entrevista recoge interés cuando el estudiantado describe las clases como divertidas, diferentes a las habituales y asociadas con orgullo al ver funcionar el programa. La autoeficacia aparece en expresiones que pasan de la confusión inicial a la idea de lograrlo, apoyadas en prácticas como corregir errores, pensar paso a paso y pedir explicaciones. El valor percibido emerge cuando algunos estudiantes conectan la experiencia con metas futuras, como estudiar ingeniería o programación, o con comprender mejor la tecnología y otros temas escolares. El registro de observación acompaña esta lectura al describir interés y entusiasmo, y al consignar frustración ante fallos que no detuvo el trabajo, porque el grupo insistió en corregir. Con este perfil de actitudes, la participación permite examinar cómo la disposición pasó directamente a acciones sostenidas durante la intervención.

La observación de aula describe participación mediante involucramiento, iniciativa para ensayar cambios y permanencia en la tarea cuando aparecieron dificultades, con equipos que mantuvieron el trabajo hasta cerrar el reto. En disposición, la entrevista muestra deseo de seguir con actividades similares en otras clases y de repetir las que generaron interés, junto con solicitudes de más tiempo para terminar. El estudiantado atribuye su motivación a ver resultados del proyecto, crear juegos y trabajar con compañeros, y reconoce barreras como acceso a *flowcode*, manejo del computador y orden de instrucciones. En conjunto, la disposición toma forma en volver a intentar tras un error, buscar apoyo en el grupo y sostener el trabajo pese a limitaciones de tiempo o de

equipos. Estos rasgos permiten interpretar la motivación como una condición que acompaña los procesos de pensamiento descritos en las categorías previas.

El hallazgo sugiere que la motivación se expresó como interés por la tarea, confianza que crece con la práctica y disposición a insistir cuando el programa falla. En este sentido, Ribarić et al. (2025) proponen interpretar la motivación STEM por dimensiones, y los relatos del grupo permiten distinguir interés, autoeficacia y valor percibido en su experiencia. El paso de nerviosismo a satisfacción cuando el proyecto funciona sugiere que la retroalimentación del sistema actuó como señal de avance y organizó la persistencia durante la clase. Aspecto relacionado con lo expuesto por Vygotsky (1978) quien entiende el aprendizaje como actividad mediada por interacción social, por lo que la ayuda entre pares explica que varios estudiantes mantuvieran el esfuerzo ante el error. Este mecanismo se alinea con estrategias de aula basadas en participación, donde la práctica compartida sostiene la atención y el compromiso durante tareas tecnológicas (Kanobel et al., 2019). La implicación es diseñar apoyos iniciales y tiempos de trabajo que reduzcan barreras de acceso, para que la motivación se traduzca en participación sostenida en todo el grupo. Con esta interpretación, el capítulo puede cerrar con ajustes pedagógicos que articulen motivación, cooperación y uso de recursos, y que preparen la discusión de alcances y límites de la estrategia.

Deci y Ryan (2000) postulan que la motivación se articula en torno a tres necesidades psicológicas básicas, con la autonomía, la competencia y la relación. En el presente estudio se evidenció un patrón coherente con esa tríada, aunque con matices. La escala Likert aplicada al final de la intervención registró medias de 4,2 sobre 5 en la subescala de interés con placer, 4,1 en trabajo en grupo y 4,0 en la percepción del pensamiento computacional; las subescalas de autoeficacia y utilidad percibida se mantuvieron favorables, con medias cercanas a 3,7. Las entrevistas aportaron la textura cualitativa al reporte cuantitativo: los estudiantes describieron el momento del encendido

de los LEDs como un “ya funciona” cargado de orgullo colectivo, expresión que recogieron siete de los diez participantes. Resnick (2017) enfatiza que los entornos de programación creativa sostienen el compromiso cuando el producto final es tangible, tesis que este estudio confirma. Pero las puntuaciones algo menores en autoeficacia sugieren que la sensación de competencia aún estaba en construcción al cierre de la intervención, lo que abre una pregunta sobre la duración necesaria para consolidar ese eje motivacional

CAPÍTULO V. CONCLUSIONES Y RECOMENDACIONES

5.1 Conclusiones

El presente capítulo articula los hallazgos obtenidos durante la intervención pedagógica con estudiantes de sexto grado de la Institución Educativa Agropecuaria José Dolores Daza, ubicada en el corregimiento de Los Milagros. Las cinco secciones siguientes recogen, en orden, la síntesis integrada de resultados, las conclusiones por objetivo específico, las implicaciones para el diseño curricular y la investigación futura, las limitaciones metodológicas identificadas y un bloque de recomendaciones dirigidas a la institución, al profesorado y a la comunidad investigadora.

Se evidenció desarrollo de arquitectura cognitiva cuando los estudiantes fraccionaron el funcionamiento del semáforo en estados discretos rojo, amarillo, verde y formularon bloques de secuencia en *Scratch* antes de trasladarlos a *Flowcode*. Las mejoras porcentuales del postest en ítems de descomposición y abstracción confirman ese desplazamiento, con mejoras superiores a veinte puntos respecto del pretest en los reactivos de identificación de subproblemas. Los estudiantes modelaron el semáforo como sistema con entradas y salidas según el pulsador, el temporizador y el LED, y las entrevistas semiestructuradas recogieron expresiones como “si cambio el tiempo, cambia el ciclo” que delatan una primera intuición sistémica. Pero al solicitar justificaciones del porqué de un fallo o ajustes a partir del error, las respuestas derivaron hacia descripciones del objeto y no hacia explicaciones del comportamiento del programa. La observación de aula ratificó el patrón al documentar cinco episodios en que los estudiantes corrigieron el código por tanteo sin articular razones.

La escala Likert aplicada recogió puntuaciones mayoritariamente positivas en interés con placer, trabajo en grupo y percepción del pensamiento computacional, con medias cercanas o superiores a cuatro sobre cinco; las subescalas de autoeficacia y utilidad percibida se mantuvieron favorables aunque con valores algo menores, el semáforo funcional operó como artefacto integrador y las entrevistas describieron el momento del encendido de los LED como un “ya funciona” cargado de orgullo colectivo, coincidente con los hallazgos de Erol y Çırak (2022) sobre *Scratch* y resolución de problemas.

El primer objetivo buscaba diagnosticar el nivel inicial de pensamiento computacional y sistémico. Los resultados del pretest ubicaron al grupo en un rango bajo-medio, con medias que oscilaron entre el 35 % y el 48 % de acierto según la subdimensión. Descomposición y reconocimiento de patrones obtuvieron los valores más altos dentro de ese rango inicial, mientras que modelado, retroalimentación y justificación de decisiones registraron los más bajos. La distribución de puntajes reveló también una baja variabilidad, circunstancia que sugirió homogeneidad en la exposición previa prácticamente nula a entornos de programación visual y a *hardware* programable. Ese punto de partida orientó el diseño posterior de las sesiones, que privilegió entradas simples y progresivas antes de introducir la placa con el microcontrolador PIC16F73.

El segundo objetivo perseguía implementar una estrategia pedagógica orientada al pensamiento computacional y sistémico con integración de la motivación. Las nueve sesiones, con una carga total de once horas, resultaron viables dentro del calendario escolar rural y compatibles con la infraestructura disponible. La transición *Scratch* → *Flowcode* → *board* fue comprendida por los estudiantes como tres representaciones del mismo problema: primero un juego con *sprites*, luego un diagrama de flujo y finalmente un circuito físico. La observación de aula registró un

aumento progresivo del trabajo colaborativo, con reasignación espontánea de roles entre el programador con el verificador y el cableador, y episodios de enseñanza entre pares. El temor inicial ante el *hardware*, documentado en las primeras dos sesiones, cedió hacia la sesión cinco, cuando la mayoría de duplas encendió el primer LED sin asistencia del docente.

El tercer objetivo proponía evaluar el desarrollo alcanzado mediante postest y escala de motivación. Las ganancias porcentuales fueron heterogéneas. Descomposición, abstracción y modelado básico mostraron mejoras promedio entre dieciocho y veinticinco puntos porcentuales respecto del pretest, con ítems en que se rozó el 80 % de acierto. Retroalimentación y justificación de decisiones avanzaron de modo más tímido, con ganancias inferiores a diez puntos. La escala Likert ratificó una motivación mayoritariamente positiva, con pico en las subescalas de interés con placer y trabajo en grupo, y con valores ligeramente más discretos en autoeficacia y utilidad percibida. Los datos confirman esa lectura la experiencia resultó motivadora y eficaz para competencias básicas del pensamiento computacional, pero dejó al descubierto un techo en las habilidades argumentativas y metacognitivas.

5.3 Implicaciones para el diseño curricular y trabajos futuros

La retroalimentación y justificación de decisiones no se desarrollaron plenamente en sexto grado, pese a una intervención bien recibida y metodológicamente cuidada. Los estudiantes de sexto grado, con edades entre once y doce años, se ubican justo en el umbral de esa transición, lo que explica buena parte de las dificultades observadas para verbalizar la causa de un error o reformular el diseño a partir de evidencia.

De ahí se desprende una ruta para la investigación posterior y para el diseño curricular. Replicar una investigación con diseño similar en grados noveno, décimo u once permitiría medir con mayor nitidez el aprendizaje de retroalimentación y justificación de decisiones, justo cuando

el sustrato cognitivo soporta operaciones formales estabilizadas. Una propuesta pedagógica consecuente con los hallazgos distribuiría las competencias así en sexto grado trabajaría descomposición, abstracción y modelado básico con *Scratch*; en séptimo y octavo profundizaría el modelado con *Flowcode* y las primeras prácticas con hardware; en noveno a once incorporaría ciclos deliberados de depuración, justificación escrita de decisiones de diseño y evaluación por pares.

5.4 Limitaciones del estudio

La muestra fue reducida y homogénea, circunstancia que impidió replicar la estructura factorial de los instrumentos reportada por la literatura previa y restringió los análisis estadísticos disponibles. El diseño de un solo grupo, sin grupo control, descarta por construcción la atribución causal estricta entre la intervención y las ganancias observadas. La duración acotada, con once horas distribuidas en nueve sesiones, representa una exposición corta para competencias cognitivas que requieren práctica sostenida.

El contexto rural específico del corregimiento de Los Milagros introduce una condición adicional. La conectividad intermitente, la ausencia de un aula informática plenamente equipada y la novedad absoluta del *hardware* programable configuran un escenario particular cuyos resultados no pueden extrapolarse sin matizaciones a entornos urbanos con trayectorias previas de alfabetización digital. A ello se suman dos posibles sesgos inherentes al diseño: la deseabilidad social en el autoinforme Likert, donde los estudiantes pueden inclinarse a respuestas favorables por apreciar al docente-investigador, y el efecto del investigador-docente, que simultáneamente diseña, enseña y evalúa.

5.5 Recomendaciones

A la institución educativa conviene anteponer tres líneas de acción. La primera pasa por fortalecer la infraestructura tecnológica mínima con equipos, conectividad estable y *kits* de hardware programable, bajo una planeación plurianual que garantice sostenibilidad y no dependa de proyectos aislados. La segunda consiste en articular el currículo de tecnología e informática con matemáticas y ciencias naturales mediante proyectos interdisciplinarios que aprovechen el semáforo, la estación meteorológica o el sistema de riego como artefactos integradores. La tercera demanda sostener la formación docente continua en programación visual, electrónica básica y didáctica del pensamiento computacional.

Al profesorado de sexto grado y de secundaria se sugieren metodologías activas ancladas en el aprendizaje por proyectos, con secuencias que progresen desde *Scratch* hacia *Flowcode* y luego hacia el *hardware*, de modo que cada transición refuerce el modelo mental construido en la etapa previa. También conviene incorporar rutinas explícitas de depuración donde el estudiante verbalice hipótesis sobre el fallo, pruebe, revise y explique, para trabajar de forma deliberada la retroalimentación.

A futuros investigadores se recomienda ampliar la muestra para permitir análisis factoriales y comparaciones por subgrupos, adoptar diseños cuasiexperimentales con grupo control equivalente y extender la intervención a al menos veinticinco horas distribuidas en un semestre. Sería valioso replicar el estudio en noveno, décimo y once, donde las operaciones formales se encuentran más consolidadas y donde sería factible evaluar con instrumentos específicos la justificación de decisiones y la retroalimentación iterativa.

REFERENCIAS BIBLIOGRÁFICAS

- Angrosino, M. (2012). Etnografía y observación participante en investigación cualitativa (T. del Amo y C. Blanco, Trans.). Ediciones Morata. (Obra original publicada en 2007)
- Aravena Díaz, M. D., Cárcamo Mansilla, N., Díaz Levicoy, D., Vergara Gómez, A., y Rodríguez Alberto, M. (2024). Desarrollo de habilidades STEM mediante el uso de prototipos tecnológicos en educación secundaria: Una revisión sistemática. *Interciencia*, 49(8), 481–489.
- Aristizábal Zapata, J. H., Gutiérrez Posada, J. E., y Diago, P. D. (2024). Design and validation of a Computational Thinking Test for children in the first grades of elementary education. *Multimodal Technologies and Interaction*, 8(5), Artículo 39.
<https://doi.org/10.3390/mti8050039>
- Arnold, R. D., & Wade, J. P. (2015). A definition of systems thinking: A systems approach. *Procedia Computer Science*, 44, 669–678. <https://doi.org/10.1016/j.procs.2015.03.050>
- Bandura, A. (1997). Self-efficacy: The exercise of control. W. H. Freeman.
- Bocconi, S., Chiocciariello, A., Dettori, G., Ferrari, A., Engelhardt, K., Kampylis, P., & Punie, Y. (2016). Developing computational thinking in compulsory education: Implications for policy and practice (EUR 28295 EN; JRC104188). Publications Office of the European Union. <https://doi.org/10.2791/792158>
- Braun, V., & Clarke, V. (2006). Using thematic analysis in psychology. *Qualitative Research in Psychology*, 3(2), 77–101. <https://doi.org/10.1191/1478088706qp063oa>

- Brennan, K., & Resnick, M. (2012, abril). *New frameworks for studying and assessing the development of computational thinking* [Ponencia]. Annual Meeting of the American Educational Research Association (AERA 2012), Vancouver, BC, Canadá.
- Bybee, R. W. (2013). The case for STEM education: Challenges and opportunities. National Science Teachers Association Press.
- Çetin, M., y Demircan, H. Ö. (2020). Empowering technology and engineering for STEM education through programming robots: A systematic literature review. *Early Child Development and Care*, 190(9), 1323–1335.
- Conover, J. W. (1999). *Practical nonparametric statistics* (3th ed.). John Wiley y sons.
- Creswell, J. W. (2014). *Research design: Qualitative, quantitative, and mixed methods approaches* (4th ed.). SAGE Publications.
- Dagienė, V., y Futschek, G. (2008). Bebras international contest on informatics and computer literacy: Criteria for good tasks. En R. T. Mittermeir y M. M. Sysło (Eds.), *Informatics education – Supporting computational thinking* (Lecture Notes in Computer Science, Vol. 5090, pp. 19–30). Springer. https://doi.org/10.1007/978-3-540-69924-8_2
- Davis, K. A., Grote, D., Mahmoudi, H., Perry, L., Ghaffar zadegan, N., Grohs, J., Hosseinichimeh, N., Knight, D. B., y Triantis, K. (2023). Comparing self-report assessments and scenario-based assessments of systems thinking competence. *Journal of Science Education and Technology*, 32, 793–813.
- Deci, E. L., & Ryan, R. M. (2000). The “what” and “why” of goal pursuits: Human needs and the self-determination of behavior. *Psychological Inquiry*, 11(4), 227–268.
https://doi.org/10.1207/S15327965PLI1104_01

- Denny, M., Denieffe, S., y O'Sullivan, K. (2023). Non-equivalent control group pretest–posttest design in social and behavioral research. En A. L. Nichols y J. Edlund (Eds.), *The Cambridge handbook of research methods and statistics for the social and behavioral sciences* (Vol. 1, pp. 314–332). Cambridge University Press.
- Eccles, J. S., y Wigfield, A. (2002). Motivational beliefs, values, and goals. *Annual Review of Psychology*, 53, 109–132. <https://doi.org/10.1146/annurev.psych.53.100901.135153>
- El-Hamamsy, L., Zapata-Cáceres, M., Marcelino, P., Bruno, B., Dehler Zufferey, J., Martín-Barroso, E., y Román-González, M. (2022). Comparing the psychometric properties of two primary school Computational Thinking (CT) assessments for grades 3 and 4: The Beginners' CT test (BCTt) and the competent CT test (cCTt). *Frontiers in Psychology*, 13, Artículo 1082659. <https://doi.org/10.3389/fpsyg.2022.1082659>
- Esquer Zárate, M. D. P., y Fernández Morales, K. (2021). La práctica docente en áreas STEM: Mapeo sistemático de la literatura. *Revista Educación*, 45(1), 576–591.
- Fagerlund, J., Häkkinen, P., Vesisenaho, M., & Viiri, J. (2021). Computational thinking in programming with Scratch in primary schools: A systematic review. *Computer Applications in Engineering Education*, 29(1), 12–28. <https://doi.org/10.1002/cae.22255>
- Feijoo A., A., y Rodríguez G., C. H. (2022). Hardware-software platform for the development of STEM skills. *IEEE Revista Iberoamericana de Tecnologías del Aprendizaje*, 17(2), 170–177.
- Finch, W. H. Jr. y French, B. F. (2015). *Latent variable modeling with R*. Routledge.
- Flick, U. (2015). *El diseño de investigación cualitativa* (2ª ed.). Ediciones Morata.

- Foster, C. (2024). Methodological pragmatism in educational research: From qualitative-quantitative to exploratory-confirmatory distinctions. *International Journal of Research y Method in Education*, 47(1), 4–19.
- Gavari-Starkie, E., Espinosa-Gutiérrez, P.-T., Lucini-Baquero, C., & Pastrana-Huguet, J. (2024). Importance of STEM and STEAM education for improvement of the land in the rural environment: Examples in Latin America. *Land*, 13(3), 274.
<https://doi.org/10.3390/land13030274>
- Grover, S., & Pea, R. (2013). Computational thinking in K–12: A review of the state of the field. *Educational Researcher*, 42(1), 38–43. <https://doi.org/10.3102/0013189X12463051>
- Guanotuña Balladares, G. E., Pujos Basantes, A. A., Oñate Pazmiño, M. F., Ponce Jiménez, M. A., Carrillo Llumitaxi, E. P., Delgado Yar, N. P., Váscenez Maza, E. C., y Calvopiña Trujillo, M. C. (2024). Adaptación de la metodología STEM-STEAM en la educación pospandemia: Un enfoque integral para la recuperación académica. *Revista InveCom*, 4(2), e040259.
- Guggemos, J., Seufert, S., y Román-González, M. (2023). Computational thinking assessment: Towards more vivid interpretations. *Technology, Knowledge and Learning*, 28(2), 539–568.
- Harel, I., & Papert, S. (Eds.). (1991). *Constructionism: Research reports and essays, 1985–1990*, by the Epistemology & Learning Research Group, the Media Laboratory, Massachusetts Institute of Technology. Ablex Publishing Corporation.
- Hernández Sampieri, R., Fernández Collado, C., y Baptista Lucio, P. (2014). *Metodología de la investigación* (6.^a ed.). McGraw-Hill Education.

- Hidi, S., y Renninger, K. A. (2006). The four-phase model of interest development. *Educational Psychologist*, 41(2), 111–127. https://doi.org/10.1207/s15326985ep4102_4
- Jiang, B., & Li, Z. (2021). Effect of Scratch on computational thinking skills of Chinese primary school students. *Journal of Computers in Education*, 8(4), 505–525.
<https://doi.org/10.1007/s40692-021-00190-z>
- Johnson, D. W., y Johnson, R. T. (2009). An educational psychology success story: Social interdependence theory and cooperative learning. *Educational Researcher*, 38(5), 365–379. <https://doi.org/10.3102/0013189X09339057>
- Kanobel, M. C., y Arce, A. S. (2019). Aula invertida en cursos de carreras STEM: Motivación y desempeño académico de los estudiantes. En N. Moreno Cáceres (Comp.), *Educación STEM/STEAM: Apuestas hacia la formación, impacto y proyección de seres críticos*. Fondo Editorial Universitario Servando Garcés.
- Kier, M. W., Blanchard, M. R., Osborne, J. W., y Albert, J. L. (2014). The development of the STEM Career Interest Survey (STEM-CIS). *Research in Science Education*, 44(3), 461–481. <https://doi.org/10.1007/s11165-013-9389-3>
- Kilpatrick, W. H. (1918). The project method. *Teachers College Record*, 19(4), 319–335.
- Koray, Ö. C., y Bilgin, A. (2023). The effect of block coding (Scratch) activities integrated into the 5E learning model in science teaching on students' computational thinking skills and programming self-efficacy. *Science Insights Education Frontiers*, 15(2), 2201–2217.
- Korkmaz, Ö., Çakir, R., y Özden, M. Y. (2017). A validity and reliability study of the Computational Thinking Scales (CTS). *Computers in Human Behavior*, 72, 558–569.
<https://doi.org/10.1016/j.chb.2017.01.005>

Kvale, S. (2011). Las entrevistas en investigación cualitativa (T. del Amo y C. Blanco, Trads.).

Ediciones Morata. (Obra original publicada en 2007)

Lavi, R., & Bertel, L. B. (2024). The system architecture–function–outcome framework for fostering and assessing systems thinking in first-year STEM education and its potential applications in case-based learning. *Education Sciences*, 14(7), Artículo 720.

<https://doi.org/10.3390/educsci14070720>

Marín-Ríos, A., Cano-Villa, J., y Mazo-Castañeda, A. (2023). Apropiación de la educación STEM/STEAM en Colombia: Una revisión a la producción de trabajos de grado. *Revista Científica*, 47, 55–70.

McAuley, E., Duncan, T., y Tammen, V. V. (1989). Psychometric properties of the Intrinsic Motivation Inventory in a competitive sport setting: A confirmatory factor analysis.

Research Quarterly for Exercise and Sport, 60(1), 48–58.

<https://doi.org/10.1080/02701367.1989.10607413>

Meadows, D. H. (2008). Thinking in systems: A primer (D. Wright, Ed.). Chelsea Green Publishing.

Molina-Ayuso, Á., Adamuz-Povedano, N., Bracho-López, R., & Torralbo-Rodríguez, M. (2024).

Computational thinking with Scratch: A tool to work on geometry in the fifth grade of primary education. *Sustainability*, 16(1), 110. <https://doi.org/10.3390/su16010110>

Mujib, M., y Mardiyah, M. (2025). Evaluación de actitudes hacia la ciencia, tecnología, ingeniería y matemáticas (STEM) para fomentar la creatividad en la educación secundaria. *Pixel-Bit. Revista de Medios y Educación*, 72, 39–69.

- National Academy of Engineering, & National Research Council. (2014). STEM integration in K-12 education: Status, prospects, and an agenda for research. The National Academies Press. <https://doi.org/10.17226/18612>
- OECD. (2019). OECD Future of Education and Skills 2030: OECD Learning Compass 2030. A series of concept notes. OECD Publishing.
https://www.oecd.org/content/dam/oecd/en/about/projects/edu/education-2040/1-1-learning-compass/OECD_Learning_Compass_2030_Concept_Note_Series.pdf
- Papert, S. (1980). Mindstorms: Children, computers, and powerful ideas. Basic Books.
- Patton, M. Q. (2015). Qualitative research and evaluation methods: Integrating theory and practice (4^a ed.). SAGE Publications.
- Resnick, M. (2017). *Lifelong kindergarten: Cultivating creativity through projects, passion, peers, and play*. The MIT Press. <https://doi.org/10.7551/mitpress/11017.001.0001>
- Ribarić, N., Novoselić, D., Ajduković, D., y Kovač-Andrić, E. (2025). Construct validation of the STEM motivation scale for children: A multi-timepoint study. *Education Sciences*, 15(9), 1241.
- Rodríguez Medina, M. A., Poblano-Ojinaga, E. R., Alvarado Tarango, L., González Torres, A., y Rodríguez Borbón, M. I. (2021). Validación por juicio de expertos de un instrumento de evaluación para evidencias de aprendizaje conceptual. *RIDE Revista Iberoamericana para la Investigación y el Desarrollo Educativo*, 11(22), Artículo e240.
<https://doi.org/10.23913/ride.v11i22.960>
- Román-González, M., Pérez-González, J.-C., y Jiménez-Fernández, C. (2017). Which cognitive abilities underlie computational thinking? Criterion validity of the Computational

Thinking Test. Computers in Human Behavior, 72, 678–691.

<https://doi.org/10.1016/j.chb.2016.08.047>

Saldaña, J. (2021). The coding manual for qualitative researchers (4^a ed.). SAGE Publications

Senge, P. M. (2006). The fifth discipline: The art and practice of the learning organization (Rev. and updated ed.). Doubleday/Currency.

Stave, K., & Hopper, M. (2007, julio 29–agosto 2). What constitutes systems thinking? A proposed taxonomy [Ponencia]. 25th International Conference of the System Dynamics Society, Boston, MA, Estados Unidos.

<https://proceedings.systemdynamics.org/2007/proceed/papers/STAVE210.pdf>

Stolpe, K., y Hallström, J. (2023). Visual programming as a tool for developing knowledge in STEM subjects. In BRILL eBooks (pp. 130–169).

Thomas, J. W. (2000). A review of research on project-based learning. The Autodesk Foundation.

UNESCO. (2023). Global education monitoring report 2023: Technology in education A tool on whose terms? UNESCO Publishing. <https://doi.org/10.54676/UZQV8501>

Unfried, A., Faber, M., Stanhope, D. S., y Wiebe, E. (2015). The development and validation of a measure of student attitudes toward science, technology, engineering, and math (S-STEM). Journal of Psychoeducational Assessment, 33(7), 622–639.

<https://doi.org/10.1177/0734282915571160>

Videla Reyes, R., Rossel Salas, S., Bugueño Egaña, H. E., y Urrutia Urrutia, C. (2021). Diseño e implementación de entorno educativo STEM en estudiantes de tercer año básico: Abordaje enactivo y ecológico de la experiencia de aprendizaje. Revista de Estudios y Experiencias en Educación, 20(44), 408–427.

Vygotsky, L. S. (1978). *Mind in society: The development of higher psychological processes*.

Harvard University Press.

Wing, J. M. (2006). Computational thinking. *Communications of the ACM*, 49(3), 33–35.

<https://doi.org/10.1145/1118178.1118215>

York, S., Lavi, R., Dori, Y. J., & Orgill, M. (2019). Applications of systems thinking in STEM education. *Journal of Chemical Education*, 96(12), 2742–2751.

<https://doi.org/10.1021/acs.jchemed.9b00261>

Yu, Q., Yu, K., Li, B., y Liang, Z. (2025). Effects of block-based visual programming on K-12 students' learning outcomes: A systematic review and meta-analysis. *Journal of Educational Computing Research*, 63(1), 31–63.

Zapata-Cáceres, M., Martín-Barroso, E., y Román-González, M. (2020). Computational Thinking Test for Beginners: Design and content validation. En 2020 IEEE Global Engineering Education Conference (EDUCON) (pp. 1905–1914). IEEE.

<https://doi.org/10.1109/EDUCON45650.2020.9125368>

Zhang, L., & Ma, Y. (2023). A study of the impact of project-based learning on student learning effects: A meta-analysis study. *Frontiers in Psychology*, 14[Article 1202728].

<https://doi.org/10.3389/fpsyg.2023.1202728>

Zhou, S., Merzdorf, H. E., Douglas, K. A., y Moore, T. J. (2023). Development and validation of an integrated STEM teacher classroom observation protocol. *Journal of Pre-College Engineering Education Research (J-PEER)*, 13(1), Artículo 2.

<https://doi.org/10.7771/2157-9288.1357>

ANEXOS

ANEXO A.

CUESTIONARIO PRETEST – DIAGNÓSTICO DE COMPETENCIAS STEM

Instrucciones

Lee atentamente cada una de las preguntas y selecciona la opción que consideres correcta, marcando con una “X” la letra correspondiente (a, b, c o d).

Recuerda:

- Solo una respuesta es correcta por pregunta.
- Responde con calma, sin ayuda externa.
- No dejes preguntas en blanco.
- El tiempo estimado para completar el cuestionario es de 35 minutos.

Sección A. Pensamiento Computacional (PC)

Descomposición

1. Cuando enfrentas un problema muy grande, la mejor forma de resolverlo es:
 - a) Intentar resolverlo completo sin dividirlo.
 - b) Dividirlo en partes más pequeñas y manejables.
 - c) Buscar la respuesta directamente en internet.
 - d) Esperar a que otro lo solucione.
2. Si debes programar un semáforo con tres luces (roja, amarilla y verde), lo primero que debes hacer es:
 - a) Encender todas las luces a la vez.
 - b) Analizar las funciones de cada luz y su orden.
 - c) Colocar los bloques sin pensar en el orden.
 - d) Apagar el sistema cada vez que cambie una luz.
3. Para organizar una receta en un programa, el paso correcto sería:
 - a) Escribir todos los ingredientes sin pasos.
 - b) Dividir el proceso en pasos ordenados y simples.
 - c) Copiar la receta sin revisarla.
 - d) Poner los pasos al azar.
4. Si una tarea tiene varios subprocesos (medir, comparar y mostrar), al programarla lo correcto sería:
 - a) Juntar todo en un solo bloque.
 - b) Separar cada subproceso en partes distintas.
 - c) Eliminar pasos para ahorrar tiempo.
 - d) Repetir los pasos sin orden lógico.

Abstracción

5. En un problema sobre el clima, centrarse solo en temperatura, humedad y viento significa:
 - a) Usar todos los datos disponibles sin selección.

- b) Ignorar lo que no influye en la solución.
 - c) Copiar datos sin analizarlos.
 - d) Evitar pensar en los elementos principales.
6. Si quieres diseñar un juego donde un personaje salte obstáculos, ¿qué información necesitas primero?
- a) Los colores del fondo.
 - b) Las reglas y condiciones del salto.
 - c) El tamaño de la pantalla.
 - d) El nombre del juego.
7. Cuando abstraes un problema, estás:
- a) Eliminando detalles innecesarios para concentrarte en lo importante.
 - b) Repitiendo todos los pasos sin analizarlos.
 - c) Memorizando todo sin entender.
 - d) Agregando más información de la necesaria.
8. Si debes programar un robot que recoja basura, lo primero es:
- a) Dibujar el robot sin definir su función.
 - b) Analizar las tareas clave que debe realizar.
 - c) Empezar a escribir sin planificar.
 - d) Colocar bloques al azar.

Sección B. Pensamiento Sistémico (PS)

Modelado

9. Si representas un ecosistema con flechas que muestran cómo los animales dependen unos de otros, estás haciendo:
- a) Un dibujo artístico.
 - b) Un modelo que representa las relaciones del sistema.
 - c) Una lista de especies.
 - d) Un resumen de nombres.
10. Cuando un estudiante usa un programa para simular el crecimiento de plantas según la cantidad de luz y agua, está:
- a) Usando un modelo que representa el sistema natural.
 - b) Repitiendo datos sin sentido.
 - c) Analizando un solo elemento aislado.
 - d) Inventando los resultados.
11. Si al aumentar una variable (como la lluvia) el modelo muestra más vegetación, esto indica que:
- a) No hay relación entre los datos.
 - b) El modelo refleja cómo los componentes se influyen entre sí.
 - c) La simulación está equivocada.
 - d) Las variables son independientes.
12. Un modelo computacional sirve principalmente para:
- a) Adivinar respuestas sin analizar.
 - b) Representar y probar comportamientos de un sistema.
 - c) Mostrar solo los resultados finales.
 - d) Repetir valores sin explicarlos.

Retroalimentación

13. Cuando un sensor de temperatura apaga un ventilador al llegar a un valor alto, ocurre:

- a) Un error en el sistema.
- b) Una retroalimentación que regula el comportamiento.
- c) Una acción sin relación.
- d) Un fallo del sensor.

14. Si al encender una lámpara con sensor de luz esta se apaga al recibir más claridad, el sistema muestra:

- a) Un proceso de retroalimentación negativa.
- b) Una acción automática sin control.
- c) Un error de conexión.
- d) Una reacción sin lógica.

15. Cuando un grupo revisa los resultados de una simulación para mejorarlos, están aplicando:

- a) Retroalimentación del sistema.
- b) Repetición sin cambios.
- c) Copia de otros modelos.
- d) Ensayo sin reflexión.

Frase de cierre

Gracias por tu participación.

Tus respuestas no serán calificadas con nota, sino utilizadas para conocer mejor tus aprendizajes y diseñar actividades que te ayuden a desarrollar tus habilidades en pensamiento computacional y sistémico. ¡Tu esfuerzo y sinceridad son muy importantes para el éxito de este proyecto!

ANEXO B.

CUESTIONARIO POSTEST – EVALUACIÓN DE COMPETENCIAS STEM

Instrucciones

Lee atentamente cada pregunta y marca con una “X” la opción que consideres correcta (a, b, c o d).

En la última sección encontrarás preguntas de desempeño práctico, donde deberás escribir o seleccionar la respuesta que mejor refleje tu razonamiento.

- Cada ítem vale 1 punto (total 30 puntos).
- El tiempo estimado es de 45 minutos.
- No hay respuestas buenas o malas; lo importante es que respondas con sinceridad y razonamiento propio.

Sección A. Pensamiento Computacional (PC)

Descomposición

1. Si un programa no funciona correctamente, lo primero que debes hacer para encontrar el problema es:
 - a) Cambiar todos los bloques sin revisar.
 - b) Revisar cada parte del programa por separado.
 - c) Reiniciar el computador.
 - d) Eliminar el archivo.
2. Para diseñar una aplicación que controle un semáforo, el paso inicial es:
 - a) Unir todos los colores en un solo bloque.
 - b) Definir la secuencia de encendido de cada luz.
 - c) Hacer que todas las luces cambien al azar.
 - d) Usar cualquier bloque sin orden.
3. Cuando divides un problema en partes más pequeñas, estás aplicando:
 - a) Pensamiento memorístico.
 - b) Descomposición.
 - c) Abstracción.
 - d) Repetición.
4. Para construir un robot que riegue plantas, ¿cuál es la primera acción lógica?
 - a) Crear el código sin analizar el proceso.
 - b) Identificar los pasos: medir humedad, decidir, regar.
 - c) Pegar sensores sin planificar.
 - d) Encenderlo y probar sin programación.

Abstracción

5. Si en un proyecto decides usar solo los datos necesarios para resolver el problema, estás aplicando:
 - a) Abstracción.
 - b) Ensayo y error.
 - c) Repetición.
 - d) Prueba aleatoria.
6. Para diseñar un juego, concentrarte en las reglas principales y no en los colores es un ejemplo de:

- a) Generalización.
- b) Abstracción.
- c) Confusión.
- d) Copia.

7. Si el estudiante agrupa elementos similares y elimina los irrelevantes, demuestra:

- a) Falta de análisis.
- b) Abstracción.
- c) Descomposición.
- d) Repetición.

8. En programación, seleccionar solo las variables que afectan el resultado final refleja:

- a) Análisis de datos.
- b) Abstracción.
- c) Ensayo sin control.
- d) Copia de código.

Sección B. Pensamiento Sistémico (PS)

Modelado

9. Si un estudiante usa una simulación para mostrar cómo la lluvia afecta el crecimiento de las plantas, está aplicando:

- a) Cálculo mental.
- b) Modelado de un sistema.
- c) Repetición.
- d) Copia de datos.

10. Representar las relaciones entre componentes del sistema con flechas es una forma de:

- a) Decorar el diagrama.
- b) Modelar visualmente un sistema.
- c) Escribir sin análisis.
- d) Elaborar listas.

11. Un modelo que muestra cómo el aumento de luz mejora la fotosíntesis indica que el estudiante:

- a) Comprende relaciones causales.
- b) Copia información textual.
- c) Memoriza resultados.
- d) No analiza datos.

12. Cuando se crea un modelo para probar “qué pasaría si” cambian los valores de entrada, se está aplicando:

- a) Retroalimentación.
- b) Modelado predictivo.
- c) Descomposición.
- d) Observación pasiva.

Retroalimentación

13. Si al simular un circuito el LED se enciende cuando el sensor detecta poca luz, esto representa:

- a) Retroalimentación del sistema.
- b) Error lógico.
- c) Causalidad sin sentido.

d) Ciclo incorrecto.

14. Cuando el estudiante revisa los resultados del experimento para ajustar su programación, está aplicando:

a) Retroalimentación para mejora.

b) Repetición de errores.

c) Prueba sin reflexión.

d) Ensayo aleatorio.

15. Si un sensor detiene el motor al alcanzar un límite, este comportamiento muestra:

a) Un bucle sin fin.

b) Un proceso de control con retroalimentación.

c) Una falla eléctrica.

d) Un error lógico.

Sección C. Ítems de desempeño (aplicación práctica)

16. Explica con tus palabras cómo dividirías el proceso para crear un robot que mida la temperatura y encienda un ventilador cuando supere cierto valor.

Respuesta: medir, comparar y encender.

17. Dibuja o describe brevemente cómo modelarías un sistema que mida la cantidad de agua en una planta y active una bomba cuando esté seca.

Respuesta: el sensor seco y la bomba se activa

18. Si un modelo muestra resultados incorrectos, ¿qué pasos seguirías para corregirlos?

Respuesta: Probaría con nuevos datos

19. Describe un ejemplo real de retroalimentación en la naturaleza o la tecnología (por ejemplo, un sistema de control automático).

Respuesta: purificador de aire.

20. ¿Qué aprendiste sobre cómo los sistemas tecnológicos y naturales están conectados y responden a los cambios?

Respuesta: siguen reglas para funcionar.

ANEXO C.

FORMATO GUIA ENCUESTA DE MOTIVACIÓN HACIA STEM

Identificación del instrumento:

- Tipo: Escala tipo Likert con preguntas abiertas
- Momento de aplicación: Posterior a la intervención
- Población: Estudiantes de grado sexto
- Duración aproximada: 20 minutos

Escala de respuesta (Likert)

1 = Totalmente en desacuerdo

2 = En desacuerdo

3 = Ni de acuerdo ni en desacuerdo

4 = De acuerdo

5 = Totalmente de acuerdo

Dimensión	Ítem por pregunta	Escala Likert
Interés	1. Me sentí interesado/a durante las actividades de programación y uso de tecnología.	1-2-3-4-5
	2. Las actividades realizadas me parecieron divertidas.	1-2-3-4-5
	3. Disfruté participar en las clases con programación y tecnología.	1-2-3-4-5
Autoeficacia	4. Creo que soy capaz de resolver problemas usando programación.	1-2-3-4-5
	5. Me sentí seguro/a al realizar las actividades propuestas.	1-2-3-4-5
Valor percibido	6. Las actividades me ayudaron a aprender de una forma diferente.	1-2-3-4-5
	7. Considero que lo aprendido será útil para otras materias.	1-2-3-4-5

Disposición	8. Me gustaría seguir aprendiendo con este tipo de actividades.	1-2-3-4-5
	9. Me interesa aprender más sobre tecnología y programación.	1-2-3-4-5
Motivación	10. Me sentí motivado/a para participar activamente en las clases.	1-2-3-4-5

Nota. Elaboración propia
Observaciones:

ANEXO D.

FORMATO GUÍA ENTREVISTA SEMIESTRUCTURADA A ESTUDIANTES

Identificación del instrumento:

- Tipo: Entrevista semiestructurada
- Momento de aplicación: Al finalizar la intervención
- Población: Submuestra de estudiantes
- Duración aproximada: 15–20 minutos

Preguntas:

Sección A. Pensamiento Computacional (PC)

Descomposición

1. ¿Crees que estas actividades te ayudaron a pensar mejor para resolver problemas?
Explicalo.

Abstracción

2. ¿Qué cambiarías o mejorarías de las actividades realizadas?

Sección B. Pensamiento Sistémico (PS)

Modelado

3. ¿Sientes que ahora entiendes mejor cómo funcionan los programas o los dispositivos que usaste?

Retroalimentación

4. ¿De qué manera trabajaste con tus compañeros durante las actividades?

Motivación

Interés

5. ¿Cómo te sentiste al participar en las actividades de programación y uso de hardware durante las clases?
6. ¿Qué actividad o ejercicio te gustó más? ¿Por qué?
7. ¿Te sentiste motivado/a para participar en las clases? ¿Qué fue lo que más te motivó?

Autoeficacia

8. ¿Hubo alguna actividad que te pareciera difícil? ¿Cuál y por qué?

Valor percibido

9. ¿Crees que aprendiste algo nuevo que no habías visto antes? ¿Qué fue?

Disposición

10. ¿Te gustaría seguir aprendiendo con este tipo de actividades en otras clases? ¿Por qué?

Observaciones:

ANEXO E.
FORMATO GUÍA DE OBSERVACIÓN DE CLASE

Identificación del instrumento:

- Tipo: Observación cualitativa estructurada
- Momento de aplicación: Durante la intervención pedagógica
- Observador: Investigador

Categoría	Aspectos a observar	Registro descriptivo
PENSAMIENTO COMPUTACIONAL		
Resolución de problemas	Estrategias utilizadas, intentos, ajustes y toma de decisiones	
PENSAMIENTO SISTÉMICO		
Trabajo en equipo	Colaboración entre estudiantes, distribución de roles, apoyo mutuo	
Uso de recursos	Manejo de materiales, programación visual, hardware interactivo	
Comunicación	Expresión de ideas, explicación de procedimientos, escucha activa	
MOTIVACIÓN		
Actitudes	Interés, entusiasmo, frustración, perseverancia	

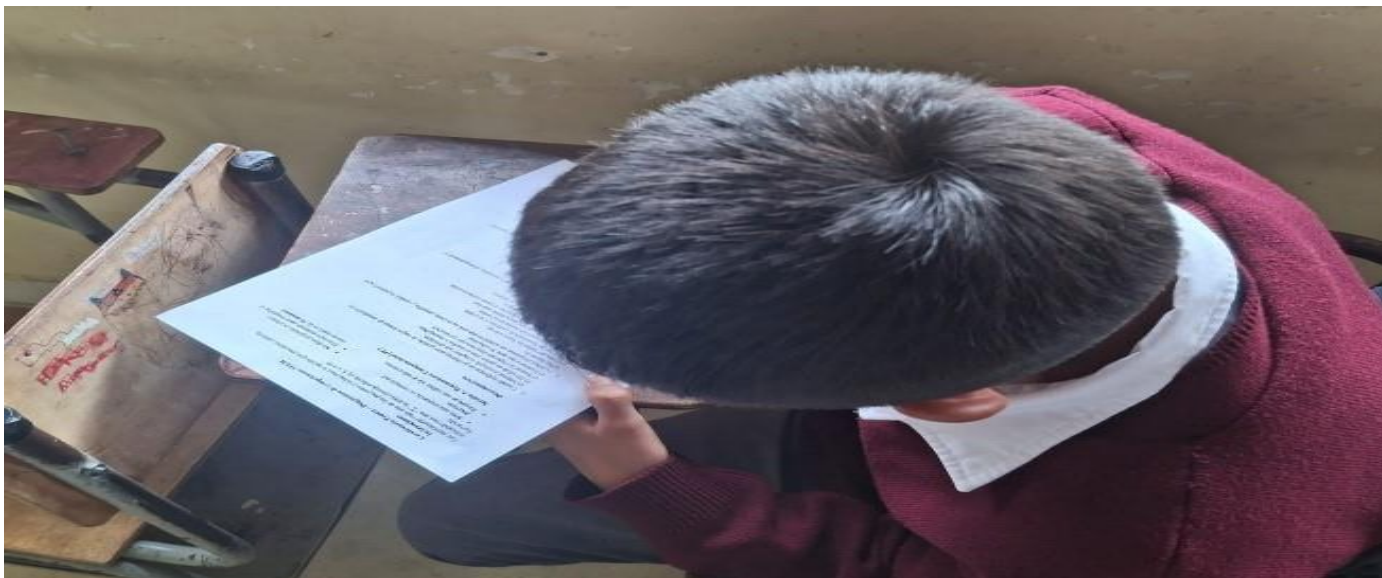
Participación

Nivel de involucramiento, iniciativa,
permanencia en la tarea

Nota. Elaboración propia

Observaciones:

ANEXO F. EVIDENCIAS DE LA IMPLEMENTACIÓN







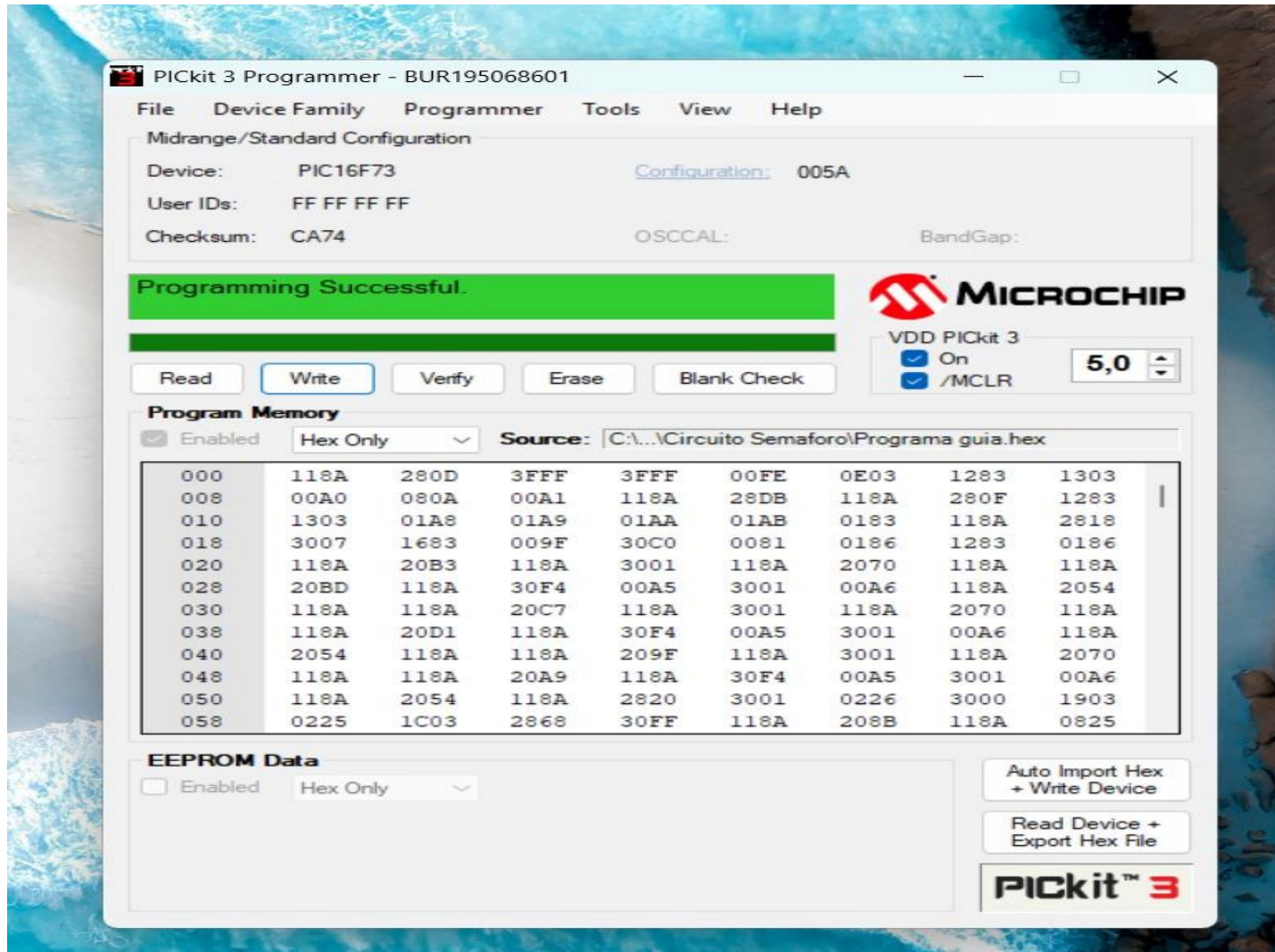
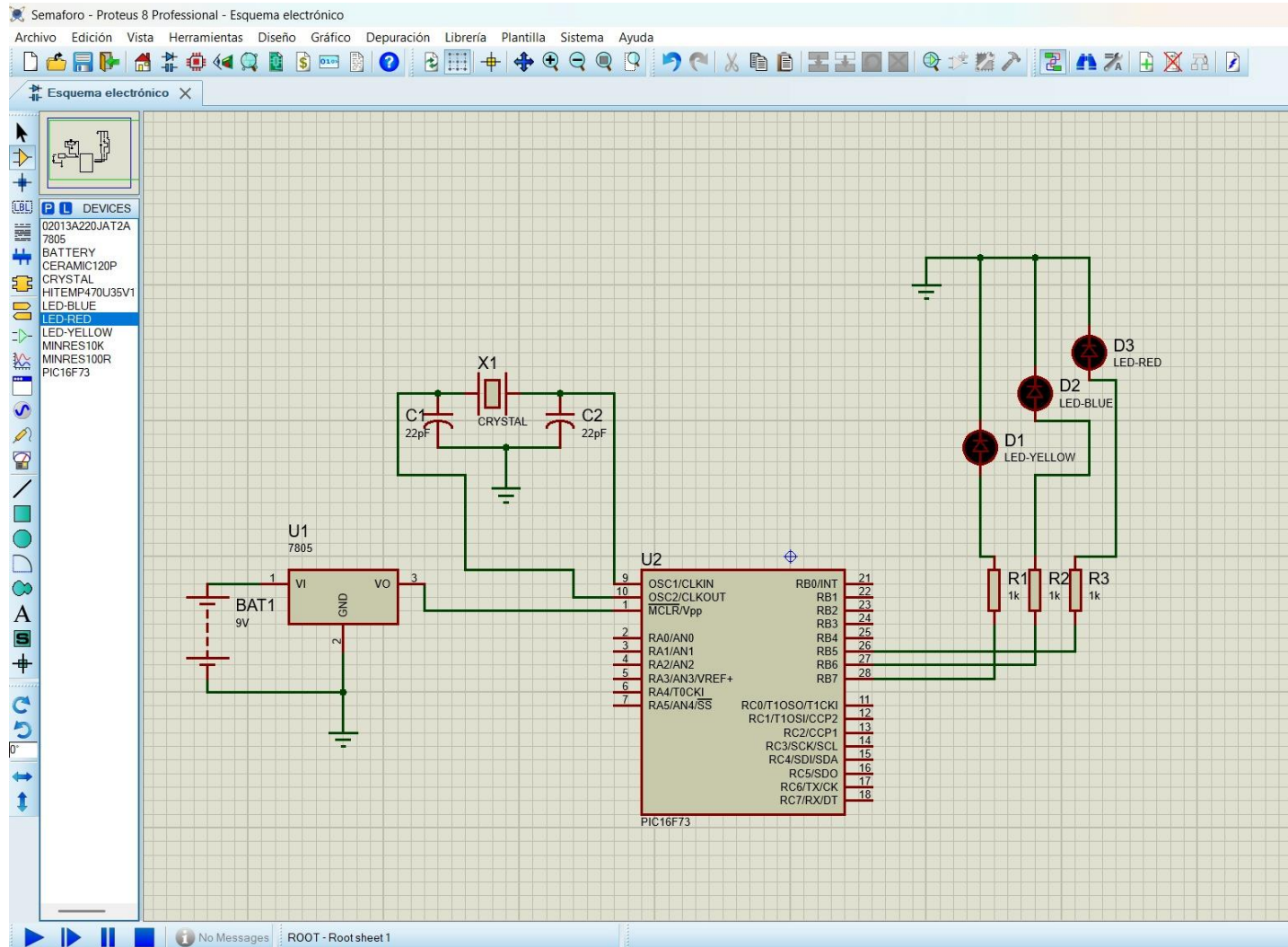


DIAGRAMA ESQUEMATICO AUTORIA PROPIA SIMULADO EN PROTEUS



PROGRAMA GUIA FLOWCODE AUTORIA PROPIA

Flowcode - Flowcode1 *

Archivo Editar Ver Herramientas de Comandos Herramientas de Componentes Macros de usuario Depurar Realizar Ventana Ayuda

Habilitar ICD Opciones de fantasmas Ghost

Vamos Pausar Parar Entrar Saltar Salir Simular código C Ejecutar

Bascular Punto de Interrupción Anular Todo Punto de Interrupción

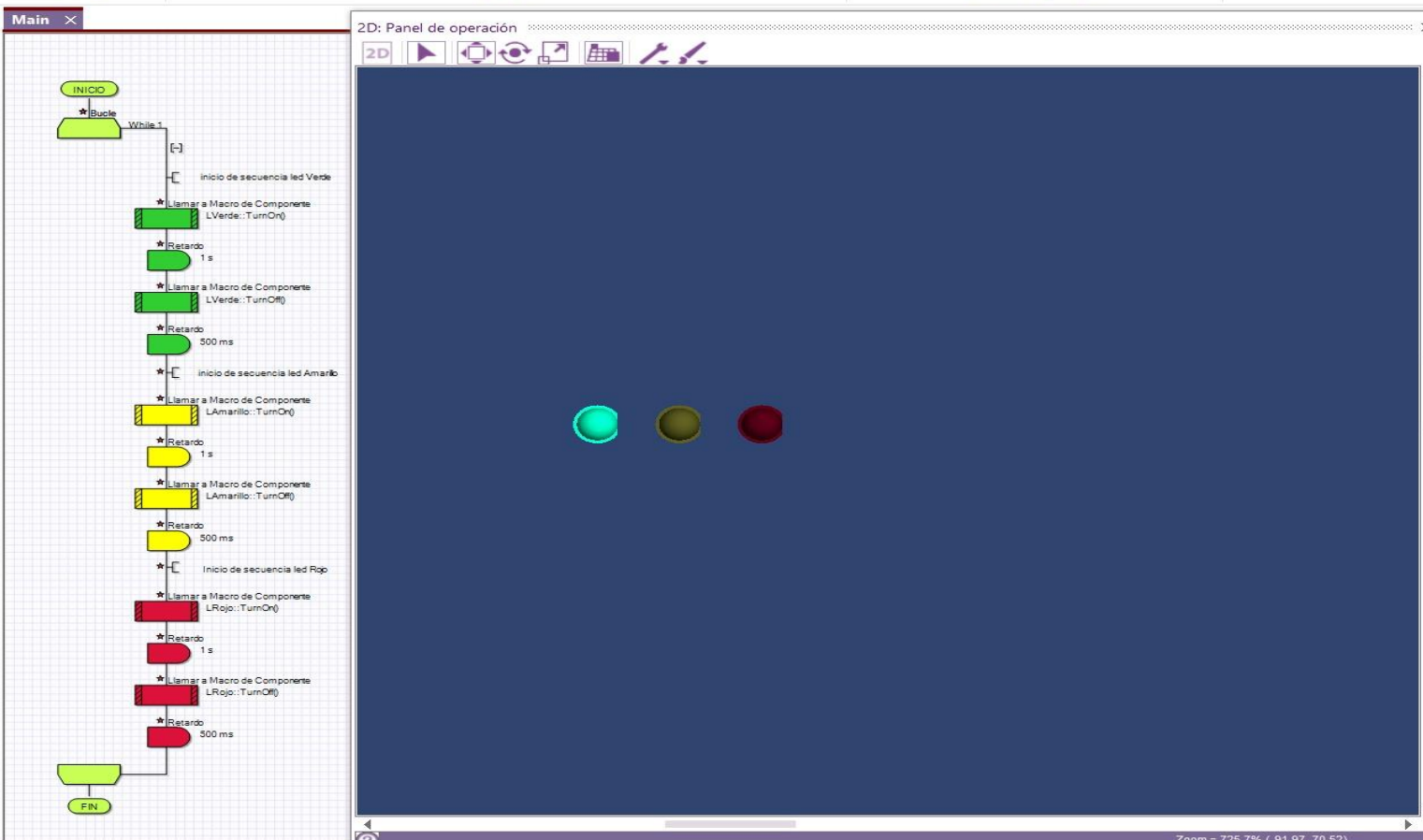
Mostrar Restablecer perfil perfil

☒ Mostrar iconos sin hit
☐ Mostrar valor en los iconos

Perfiles de código

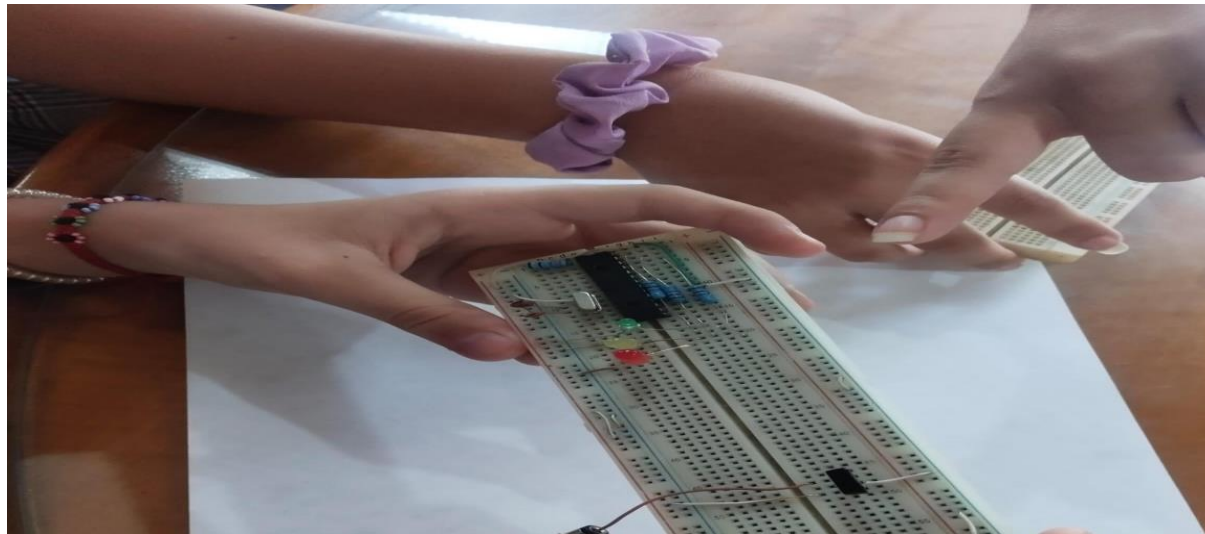
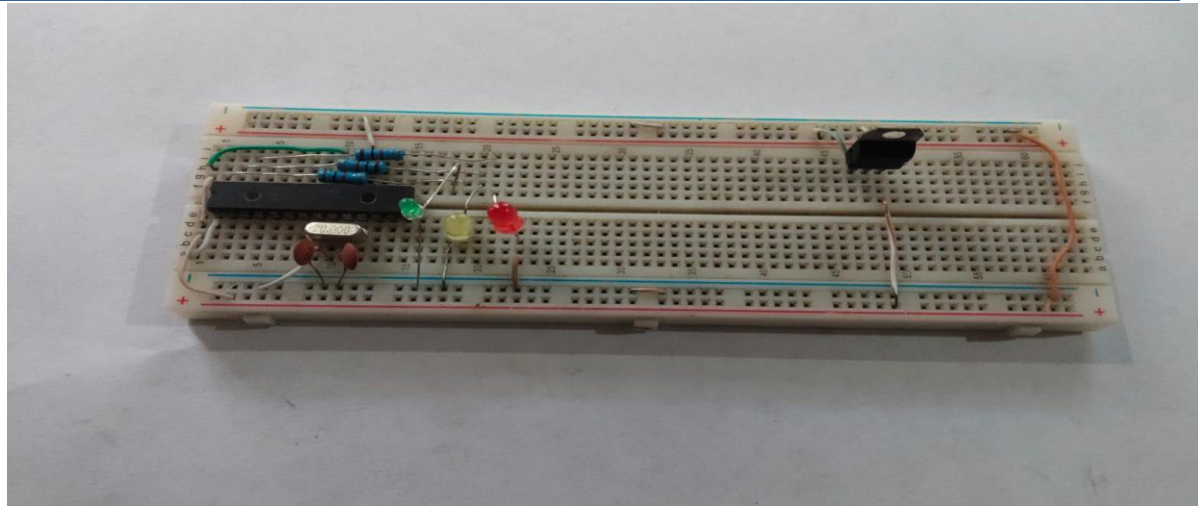
Main

2D: Panel de operación



Para Ayuda, oprima F1

Zoom = 725.7% (-91.97, 70.52)



Clase teórica. Reconocimiento del entorno flowcode: Es una interfaz gráfica diseñada para programar y simular sistemas electrónicos de manera visual. Sus áreas principales: • Área de diagrama de flujo: donde se construye el programa usando símbolos que representan instrucciones. • Panel de componentes: permite agregar elementos electrónicos como LEDs, botones, sensores o pantallas. • Área de simulación: Muestra el circuito funcionando en tiempo real para observar cómo responde el programa. • Barra de herramientas: Contiene opciones para crear, guardar, ejecutar o detener la simulación. Este entorno facilita el aprendizaje porque permite programar, visualizar y probar circuitos de forma sencilla y gráfica.	1 hora					x									
Los estudiantes acceden al software flowcode desde el computador teniendo en cuenta los pasos orientados y reconocen sus funciones.	1 hora						x								
Simulación del hardware básico. Encender un led Encender y apagar el led determinando tiempos Estudiantes experimentan el encendido de varios leds	1 hora							x							
Creación del semáforo en flowcode teniendo en cuenta: primero se abre el programa y se crea un nuevo proyecto seleccionando el microcontrolador. Luego se agregan tres LED que representarán las luces roja, amarilla y verde, y se conectan a los pines del microcontrolador. Después, en el diagrama de flujo se programan instrucciones para encender y apagar cada luz con un tiempo de espera, siguiendo el orden del semáforo (rojo, verde y amarillo). Finalmente, se coloca el proceso en un bucle para que el ciclo se repita continuamente y se ejecuta la simulación para observar su funcionamiento.	2 horas								x						
Clase teórica: conocimiento y explicación de los componentes del hardware (board o tarjeta electrónica) board o tarjeta electrónica es una plataforma donde se conectan componentes para realizar proyectos electrónicos. Al trabajar con	1 hora									x					

[illegible]

[illegible]

ANEXO H. FORMATO DE AUTORIZACIÓN INSTITUCIONAL PARA EL DESARROLLO DE LA INVESTIGACIÓN

La Institución Educativa Agropecuaria José Dolores Daza, representada por el(la) rector(a) Felipe Ernesto Perafán Martínez, en ejercicio de sus funciones legales, autoriza a los investigadores Sixta Veronica Samboni Buesaquillo, identificado(a) con CC No. 1.002.807.869, de la Cruz Nariño y Fernando José Vizcaíno Linero, identificado(a) con CC No.84.459.169 de Santa Marta para el desarrollo del proyecto de investigación titulado: ***“Aprendizaje de STEM en la infancia a través de entornos de programación visual y dispositivos de hardware interactivo”***.

El presente proyecto se enmarca en la normativa vigente en Colombia, particularmente en la Resolución 8430 de 1993 del Ministerio de Salud y Protección Social, la Ley 1098 de 2006 (Código de Infancia y Adolescencia), la Ley 1581 de 2012 (protección de datos personales) y las orientaciones éticas del Ministerio de Ciencia, Tecnología e Innovación. El nivel de riesgo de la investigación es considerado MÍNIMO, en tanto se basa en encuestas, pruebas académicas y actividades pedagógicas propias del entorno educativo. La autorización comprende:

1. Permitir el acceso a estudiantes de grado sexto para la aplicación de pre-test, post-test y actividades pedagógicas relacionadas con el estudio.
2. Garantizar que la participación de los estudiantes estará precedida por la firma de consentimientos informados de padres/madres o acudientes y asentimientos de los estudiantes.

3. Salvaguardar la confidencialidad, anonimato y uso responsable de la información recopilada, de acuerdo con la Ley 1581 de 2012.

4. Asegurar que los resultados de la investigación serán utilizados con fines académicos y científicos, sin perjuicio de los derechos de los estudiantes ni de la institución.

En constancia, se firma en Los Milagros Bolívar C a los 28 días del mes de octubre de 2025.

Faupe Rorafón M.

Rector(a)
Institución Educativa

Fernando Vizcaino

Fernando José vizcaino Linero
Investigador(a) Principal

Verónica Samboni Buesaquillo
Sixta Verónica Samboni Buesaquillo

Investigador(a) Principal

ANEXO I. CONSENTIMIENTO INFORMADO DE REPRESENTANTE LEGAL

Señores

PADRES DE FAMILIA

Institución Educativa Agropecuaria José Dolores Daza

Cordial saludo

Por medio de la presente me permito solicitar su autorización y consentimiento para la participación de su hijo en el proyecto de investigación “Aprendizaje de STEM en la infancia a través de entornos de programación visual y dispositivos de hardware interactivo”, a cargo de los docentes Fernando José Vizcaino Linero y Sixta Verónica Samboni Buesquillo como requisito para obtener el título de Magister en Educación STEM de la universidad Politécnico Grancolombiano

Dicho proyecto cuenta con las siguientes características:

Objetivos:

1. Diagnosticar el nivel de pensamiento computacional y pensamiento sistémico de los estudiantes de sexto grado mediante la aplicación de un pretest validado.
2. Implementar una estrategia pedagógica orientada al desarrollo del pensamiento sistémico, integrando la motivación y la mejora de experiencias en el aula, a partir de competencias STEM
3. Evaluar el desarrollo del nivel del pensamiento computacional y pensamiento sistémico de los estudiantes mediante el uso de programación visual y hardware interactivo por medio de un postest.

Procedimiento: Previa autorización de la institución y consentimiento informado por parte de los padres y los estudiantes, debidamente firmado, se procederá a desarrollar una intervención pedagógica que incluye el desarrollo de cuestionarios antes y después de la propuesta metodológica para medir variables relacionadas con el pensamiento lógico-algorítmico, razonamiento espacial y un cuestionario tipo Likert para valorar la motivación e interés hacia la educación STEM. Igualmente, a un subgrupo de estudiantes se les aplicará una entrevista semiestructurada para profundizar en la categoría de análisis de motivación.

Agradeciendo su atención,

Cordialmente

Fernando José Vizcaino Linero y Sixta

Verónica Samboni Buesquillo

Maestranes

Universidad Politécnico Gran Colombiano

Teléfono: 3117644565

Correo electrónico: veronnicabuesaquillo@gmail.com

Se adjunta: Formato de consentimiento informado.

ANEXO J. CONSENTIMIENTO INFORMADO

Nosotros abajo firmantes, identificados como aparece al pie de nuestras firmas, representantes legales del estudiante: _____, del grado: _____, deseamos manifestar a través de este documento, que fuimos informados suficientemente y comprendemos la justificación, los objetivos, los procedimientos y las posibles molestias y beneficios implicados en la participación de nuestro hijo(a), en el proyecto de investigación sobre “Aprendizaje de STEM en la infancia a través de entornos de programación visual y dispositivos de hardware interactivo”.

Participación Voluntaria

La participación de nuestro hijo(a) en este estudio es completamente voluntaria, si él o ella se negara a participar o decidiera retirarse, esto no le generará ningún problema, ni tendrá consecuencias a nivel institucional, ni académico, ni social. Si lo desea, nuestro hijo(a) informaría los motivos de dicho retiro a la docente encargada.

Confidencialidad

La información suministrada por nuestro hijo(a) será confidencial. Los resultados podrán ser publicados o presentados en reuniones o eventos con fines académicos sin revelar su nombre o datos de identificación.

Así mismo, declaramos que fuimos informados suficientemente y comprendemos que tenemos derecho a recibir respuesta sobre cualquier inquietud que mi hijo(a) o nosotros tengamos sobre dicha investigación, antes, durante y después de su ejecución; nos permitimos informar que consentimos, de forma libre y espontánea, la participación de nuestro hijo(a) en el mismo.

En constancia de lo anterior, firmamos el presente documento, en el corregimiento de Los Milagros _____ el día _____, del mes _____ de _____,

Firma _____

Nombre _____

C. C. No. _____ de _____

Firma: _____

Nombre _____

C. C. No. _____ de _____