

# khbjhbkj

*por* Hector VELASQUEZ CUEVA

---

**Fecha de entrega:** 31-mar-2023 09:25a.m. (UTC-0500)

**Identificador de la entrega:** 2052076616

**Nombre del archivo:** erruto\_-\_Br.\_Buenaventura\_Quispe\_Apaza\_-\_22\_marzo\_-turnitin.docx (265.21K)

**Total de palabras:** 12630

**Total de caracteres:** 70697

**UNIVERSIDAD CATÓLICA DE TRUJILLO**

**BENEDICTO XVI**

**FACULTAD DE HUMANIDADES**

**PROGRAMA COMPLEMENTACIÓN PEDAGÓGICA**

**UNIVERSITARIA**



**DOMINIO DEL SOFTWARE EDUCATIVO SCRATCH Y EL  
DESARROLLO DEL PENSAMIENTO COMPUTACIONAL EN  
ESTUDIANTES DE UNA INSTITUCIÓN EDUCATIVA SECUNDARIA  
DE YUNGUYO 2022**

**Tesis para obtener el título profesional de  
LICENCIADO EN EDUCACIÓN SECUNDARIA CON MENCIÓN EN  
COMPUTACIÓN E INFORMÁTICA**

**AUTORES**

**Br. Jose El Justo Ortega Serruto**

**Br. Buenaventura Quispe Apaza**

**ASESOR**

**Mg. Hector Israel Velásquez Cueva**

**5**

**LÍNEA DE INVESTIGACIÓN**

**Educación y responsabilidad social**

**TRUJILLO - PERÚ**

**2022**

## I. INTRODUCCIÓN

Es fundamental mejorar significativamente los procesos de enseñanza-aprendizaje para hacer frente a los posibles problemas que puedan ocurrir, especialmente los relacionados con las habilidades informáticas, puesto que vivimos en un mundo en el que la tecnología está en continua expansión. En el contexto internacional según el estudio de Bocconi et al. (2016) sobre “la programación y el pensamiento computacional” se han incorporado a la educación primaria y secundaria en varias naciones europeas (Inglaterra, Francia, Finlandia, Italia, Portugal), ya sea como asignatura obligatoria o como actividad extracurricular. Además, Duarte et al. (2020) señalan que estas actividades son en parte posibles gracias a la amplia disponibilidad de ordenadores y acceso a Internet en las naciones europeas. Finlandia es uno de los países que ya no exige a los alumnos que reciban clases de escritura cursiva. El sistema educativo finlandés ha decidido que, a partir de 2016-17, las clases de mecanografía con el teclado QWERTY sustituirán a las de escritura a mano. Debido a la buena reputación de la que goza el país gracias a los resultados de sus exámenes PISA, es probable que esta decisión comience a ser imitada en el resto del mundo. En España, donde se señaló que también hay que trabajar para fomentar el pensamiento innovador de los profesores mediante procedimientos de formación y alfabetización audiovisual (Triviño y Vaquero, 2019). En México se considera la imaginación como un componente fundamental para las soluciones novedosas (Hernández et al. 2018).

En el contexto Latinoamericano, en cuanto al pensamiento computacional, según Vásquez et al. (2019) los gobiernos de América Latina están comenzando a tomar en serio el pensamiento computacional puesto que en Argentina se hizo obligatorio desde 2018. En Uruguay no es obligatorio, pero como es voluntario, es difícil medir su avance. Así mismo, Vásquez et al. (2019) mencionan que el proyecto “Programa tus ideas”, activo desde 2015, y la adopción por parte de la Red Escolar SIP de lecciones de codificación a través de sitios web como Code.org, Scratch y Khan Academy son solo dos ejemplos de las iniciativas relacionadas con el PC que se han desarrollado en Chile. En un estudio realizado en Ecuador por Sandoval et al. (2020) revelaron que el lenguaje de programación Scratch beneficia el pensamiento abstracto, la autorreflexión, la aritmética y las capacidades cognitivas de los niños.

El proceso de sensibilización y formación de los alumnos es uno de los componentes más significativos del sistema educativo de la EBR en los niveles primaria y secundaria. Los profesores entenderán que el uso de las nuevas tecnologías redundará en una mejora del nivel educativo en la medida en que se expongan a las posibilidades que estas herramientas pueden proporcionarles como recursos didácticos en su instrucción y así fomentar el desarrollo de habilidades y destrezas en los alumnos, puesto que hoy en día es fundamental conocer acerca de la programación y el uso eficiente de los avances de tecnología digital basados en el pensamiento computacional. Además, supone dar libertad a los estudiantes para que diseñen sus propios proyectos, lo que genera nuevas oportunidades educativas. (Gómez y Williamson, 2018).

En el contexto nacional actualmente nuestro Sistema Educativo viene siendo afectado por diversos cambios sociales y sobre todo por la coyuntura del estado de emergencia sanitaria que vivimos debido a la pandemia del COVID-19. Donde se descubrió muchas carencias y brechas respecto al acceso a la tecnología digital, en donde las competencias Digitales cobraron mucha fuerza y se demostró su gran importancia en la educación de modalidad virtual.

En el contexto local, específicamente la Institución Educativa Secundaria José Gálvez de la ciudad de Yunguyo no fue ajeno a las consecuencias ocasionadas por el aislamiento social, puesto que durante el periodo de emergencia optaron por la enseñanza de la modalidad virtual, basados bajo el enfoque de la estrategia de “Aprendo en casa”, cuya característica principal fue una enseñanza interactiva mediante el uso de la tecnología digital en base al internet y soporte tecnológico.

Esta situación logró evidenciar la actual problemática que presentan los estudiantes, puesto que se logró observar muchas dificultades y obstáculos durante el desarrollo de las actividades académicas a través de la modalidad virtual, siendo una principal deficiencia el no conocer el sistema de programación y el funcionamiento del sistema operativo de la computadora. Además de no contar con suficiente conocimiento acerca del uso de programas y acceso a las redes virtuales. Lo mencionado permitió establecer el propósito de conocer el dominio del software Scratch y el nivel de pensamiento computacional con la finalidad de establecer la relación que existe entre ambas variables.

Por lo tanto, se planteó como pregunta general de investigación ¿De qué manera el dominio del software educativo Scratch se relaciona con el nivel del pensamiento computacional de los estudiantes de la IES José Gálvez de Yunguyo, 2022? Y los problemas específicos: PE1: ¿De qué manera el dominio del software educativo Scratch se relaciona con el desarrollo de conceptos computacionales de los estudiantes? PE2: ¿De qué manera el dominio del software educativo Scratch se relaciona con el desarrollo de prácticas computacionales de los estudiantes? PE3: ¿De qué manera el dominio del software educativo Scratch se relaciona con el desarrollo de perspectivas computacionales de los estudiantes?

Con respecto a la justificación el Sistema educativo en la actualidad requiere del desarrollo de habilidades de orden superior, como los pensamientos crítico, creativo, reflexivo, computacional. Así mismo, se busca utilizar numerosas herramientas digitales, (Monjelat y San Martín, 2016). Siendo el Scratch una ellas para gestionar diversas tareas académicas. El estudio fue realizado con los estudiantes del nivel Secundaria de la Institución Educativa “José Gálvez”. Así mismo, se empleó las instalaciones de la Institución, gracias al apoyo y visto bueno de sus autoridades.

El pensamiento computacional se ha introducido en el plan de estudios de los programas de educación básica en países como Estonia, Francia y el Reino Unido. “Esto ha apoyado el desarrollo de habilidades transversales como el pensamiento analítico, la resolución de problemas y la creatividad, que son vitales para enfrentar los cambios del entorno actual”. Por lo tanto, se requiere de investigaciones que muestren los procesos adecuados para el desarrollo del pensamiento computacional y que involucren a los estudiantes de nuestra nación en este desarrollo. (Bocconi et al. 2016)

Cabe precisar que, es fundamental destacar que el éxito futuro de un profesional en su campo depende en gran medida de las habilidades que ha desarrollado desde sus primeros años escolares con relación a las competencias tecnológicas. Debido a los resultados que ha venido mostrando a favor del potencial mostrado por los estudiantes, así como por la aplicación de estas herramientas en su vida cotidiana, el pensamiento computacional ha cobrado recientemente una importancia significativa. Sin embargo, en nuestro país, “aún se encuentra marginado porque no es visto como relevante para la mejora del rendimiento académico y porque carece de recursos para ser implementado en el ámbito de la educación básica regular”. (Zapata, 2015)

En este ámbito existe un vacío reconocido en el desarrollo de técnicas y herramientas de evaluación del pensamiento computacional, dado que no hay muchas investigaciones. Por lo tanto, es pertinente y necesario realizar estudios con la intención de aportar material pertinente. En este sentido, Soria y Rivero (2019) afirman que hay que hacer esfuerzos en la sociedad para incorporar el pensamiento computacional como una habilidad necesaria en la formación de los alumnos en el currículo educativo. Así mismo, Code learn (2021) respalda que el pensamiento computacional tiene una serie de ventajas educativas, empezando por el uso de abstracciones que refuerzan y desarrollan las capacidades intelectuales y son transferibles a todos los demás campos. Por lo tanto, estos hallazgos serán de gran ayuda para comprender la realidad educativa y también ayudará a mejorar las deficiencias identificadas del desarrollo de aprendizaje del pensamiento computacional con el fin de sugerir actividades pertinentes a la comunidad educativa y científica.

En este sentido se estableció como objetivo general, determinar la relación que existe entre el dominio del Software Educativo Scratch y el nivel del pensamiento computacional de los estudiantes de la IES José Gálvez de Yunguyo y los objetivos específicos: OE1: Identificar la relación entre el dominio del Software Educativo Scratch y el desarrollo de conceptos computacionales de los estudiantes. OE2: Establecer la relación entre el dominio del Software Educativo Scratch y el desarrollo de prácticas computacionales de los estudiantes. OE3: Identificar la relación entre el dominio del Software Educativo Scratch y el desarrollo de perspectivas computacionales de los estudiantes. Así mismo, se formuló como hipótesis estadística general alterna (H1): Existe una relación directa entre el dominio del software educativo Scratch y el nivel del pensamiento computacional de los Estudiantes de la IES José Gálvez de Yunguyo. (H0): No existe una relación directa entre el dominio del software educativo Scratch y el nivel del pensamiento computacional de los Estudiantes de la IES José Gálvez de Yunguyo. Y las hipótesis específicas: HE1: Existe una relación directa entre el dominio del Software Educativo Scratch y el desarrollo de conceptos computacionales de los estudiantes. HE2: Existe una relación directa entre el dominio del Software Educativo Scratch y el desarrollo de prácticas computacionales de los estudiantes. HE3: Existe una relación directa entre el dominio del Software Educativo Scratch y el desarrollo de perspectivas computacionales de los estudiantes.

En cuanto a los estudios previos del contexto internacional, Barahona et al. (2022) en Ecuador, realizaron un estudio científico denominado “Fortalecimiento del pensamiento computacional en niños y adolescentes de la ciudad de Loja”. Con el objetivo de medir la mejora en el nivel de pensamiento computacional de la población. Para el efecto, emplearon la metodología del diseño experimental. En sus hallazgos lograron determinar una mejora del 15,96% con respecto a los conocimientos del pre test de los participantes, cuyo resultado les permitió arribar a la conclusión de que deberían incluirse contenidos curriculares relacionados con el pensamiento computacional independientemente de los elementos sociales, culturales y académicos que afecten al alumno.

Gamito et al. (2022) en su artículo de investigación denominado “The development of computational thinking in education: Assessment based on an experience with Scratch”. Tuvieron como objetivo “valorar la experimentación basada en Scratch llevada a cabo con el alumnado de Grado de Educación Primaria de la Universidad del País Vasco”. Así mismo, en el estudio emplearon la metodología del enfoque fenomenológico. Además, se observó la opinión de los alumnos sobre las posibles aplicaciones de Scratch para el crecimiento de las habilidades y actitudes del pensamiento computacional. Finalmente llegaron a la conclusión de que “la experiencia tuvo una valoración significativa y que Scratch puede ser útil para el desarrollo de procesos de pensamiento computacional en el aula de Educación Primaria” (p.71).

Adelmo et al. (2022) en su estudio “A Data-Driven Approach to Assess Computational Thinking Concepts Based on Learners Artifacts”. Explicaron que la integración del pensamiento computacional en la educación K-12 ha sido un tema ampliamente explorado en los últimos años. En particular, la evaluación eficaz del pensamiento computacional puede apoyar la comprensión de cómo los alumnos desarrollan conceptos y prácticas computacionales. Con el objetivo de ayudar a avanzar en la investigación sobre este tema, lograron proponer un enfoque de evaluación basado en datos utilizando el lenguaje de programación Scratch. Por lo tanto, a través del estudio tipo propositivo, lograron concluir que a través de la programación Scratch se puede mejorar y orientar el diseño de los cursos, así como también apoyar estudios sobre la evaluación del pensamiento computacional.

Kyza et al. (2022) realizaron un estudio denominado <sup>19</sup> *A cross-sectional study investigating primary school childrens coding practices and computational thinking using Scratch*. Cuyo objetivo fue “examinar las prácticas de codificación y el pensamiento computacional de los niños de la escuela primaria utilizando Scratch”. Así mismo, emplearon una metodología del estudio transversal, cuyos niños fueron organizados en dos cohortes (Cohorte 1: 6-9 años; Cohorte 2: 10-12 años). Durante la intervención, a los niños se les presentó Scratch y se les pidió que diseñaran colaborativamente una historia digital sobre acciones de gestión de residuos ambientales, adoptando así una perspectiva disciplinaria del pensamiento computacional. Los análisis de datos examinaron los artefactos finales de los niños, en términos de prácticas de codificación y el nivel de pensamiento computacional demostrado por cada cohorte. Los resultados se presentan y contrastan a través de una perspectiva de desarrollo donde se determinaron que el apoyo instructivo es necesario para apoyar <sup>1</sup> el desarrollo de las prácticas de codificación y el pensamiento computacional de los niños de primaria.

Selamí y Gülhanım (2021) realizaron un estudio denominado <sup>13</sup> *Effect of Tinkercad on Students Computational Thinking Skills and Perceptions: A Case of Ankara Province*. <sup>23</sup> Con el objetivo de determinar el efecto del uso de Tinkercad <sup>4</sup> en la educación en programación de computadoras en las percepciones y habilidades de pensamiento computacional de los estudiantes. <sup>21</sup> En este contexto, 583 estudiantes de secundaria que estudian en la provincia de Ankara de Turquía en el nivel de quinto, sexto, séptimo y octavo grado en el año académico 2019-2020 constituyen la muestra de la investigación. Para ello, emplearon los métodos de investigación mixtos. Los datos de la investigación se recopilaron utilizando el Formulario de información personal, el Cuestionario de percepción del estudiante sobre el software Tinkercad y la Escala de pensamiento de procesamiento de información. Al observar las percepciones de los estudiantes sobre Tinkercad, se determinó que estaban muy motivados por el interés y la apreciación y encontraron que Tinkercad era generalmente útil y fácil de usar. Así mismo, lograron determinar que la percepción de los estudiantes sobre el pensamiento computacional fue moderada. Se encontró que había una relación positiva de bajo nivel entre la frecuencia de uso de Tinkercad de los estudiantes y su percepción de Tinkercad. Por lo tanto, lograron concluir que “existe una relación positiva de nivel moderado entre su percepción de Tinkercad y sus habilidades de pensamiento computacional” (p.35).



Jiang et al. (2021) en su estudio Understanding the relationship between computational thinking and computational engagement: a case study from Scratch. Tuvieron como objetivo “comprender la relación entre el pensamiento computacional y la participación computacional”. Para ello, emplearon una metodología de estudio de caso basado en la teoría del aprendizaje social con una base de datos a gran escala, exploraron el impacto del pensamiento computacional reflejado en los proyectos Scratch en la participación de los usuarios. Para lo cual examinaron a los usuarios en línea de Scratch perfil de pensamiento computacional a través del análisis de conglomerados en los proyectos que crearon. El análisis de agrupamiento en sus resultados reveló tres grupos de estudiantes, y los estudiantes avanzados no crearon más proyectos que otros, pero sus proyectos atraen más participación de sus compañeros. Su análisis estadístico encuentra una fuerza de correlación de baja a moderada entre el nivel de pensamiento computacional reflejado en los proyectos y su popularidad.

İlic (2021) en su artículo titulado The impact of Scratch-assisted instruction on computational thinking (CT) skills of pre-service teachers. Planteó como objetivo determinar el efecto de las expresiones y aplicaciones asistidas por Scratch en las habilidades de pensamiento computacional de los maestros en formación. Para ello, la investigación se diseñó con un diseño secuencial exploratorio, un método de investigación mixto. Treinta y tres futuros profesores participaron en el estudio. Con base en los datos obtenidos. Con sus resultados concluye que “existe una correlación positiva y significativa entre el Pensamiento Computacional y el rendimiento académico. Además, los maestros en formación afirmaron que las aplicaciones Scratch contribuyeron a la adquisición de habilidades de pensamiento computacional” (p.46).

Pérez et al. (2019) en su estudio denominado Mastery of computational thinking skills in the students of the Sucre Superior Technological Institute of Quito – Ecuador Hamilton. Cuyo objetivo fue “diagnosticar el nivel de desarrollo del pensamiento computacional en los estudiantes del Instituto Tecnológico Superior Sucre”. Para ello, emplearon la metodología del diseño correlacional. Los resultados más significativos revelan la existencia de una relación significativa “entre las capacidades de los alumnos en el uso de bucles y algoritmos direccionales para resolver problemas algorítmicos, así como sus debilidades en el uso de la toma de decisiones y de subrutinas” (p.58).

<sup>4</sup> Arranz y Pérez (2017) realizaron un estudio denominado “Evaluación del Pensamiento Computacional en Educación” con el objetivo de “valorar el Pensamiento Computacional de los alumnos de Educación Primaria que han trabajado con el software de programación Scratch”. Para ello, 28 estudiantes participaron en un estudio que utilizó una metodología descriptiva y un cuestionario para la recolección de datos cuantitativos. A través de sus resultados lograron demostrar que el uso de Scratch influye en el crecimiento de los conceptos, las prácticas y los puntos de vista del pensamiento computacional.

En el contexto nacional <sup>1</sup> Churata et al. (2021) en Arequipa, realizaron una investigación denominada “Influencia del Scratch en el desarrollo del pensamiento computacional en los estudiantes del quinto de primaria de la institución educativa 40028 Alto Selva Alegre-Arequipa-2019”. Con el objetivo de “determinar la influencia del Uso del Scratch en el desarrollo del Pensamiento Computacional en los estudiantes de Primaria”. Para ello, emplearon la metodología del enfoque cuantitativo, con diseño descriptivo correlacional. <sup>12</sup> A través de los resultados obtenidos en la prueba t (9.323), lograron concluir que Scratch “tiene un impacto positivo en el desarrollo del pensamiento computacional de los alumnos de quinto grado, demostrando que existe una clara relación entre las variables antes y después de aplicar las sesiones” (p.54).

Taco (2019) realizó un estudio denominado “Influencia del Programa Scratch en el Pensamiento Computacional en Estudiantes del Nivel Primario de la Institución Educativa de la Policía Nacional del Perú Alférez Mariano Santos Mateos, Tacna 2018”. Con el objetivo de “determinar la influencia del programa Scratch en el pensamiento computacional de 384 estudiantes del nivel primario en Tacna”. Para ello, empleó <sup>3</sup> una metodología de tipo básica, bajo el enfoque cuantitativo, de diseño correlacional-explicativo. A través de sus resultados logró concluir que la aplicación Scratch tuvo un gran impacto en el pensamiento computacional de los niños, lo que indica su valor como herramienta para potenciar las habilidades computacionales de los estudiantes. Además, lograron evidenciar que los estudiantes del quinto grado tuvieron un desempeño regular en contraste con los estudiantes de sexto grado quienes recibieron altas puntuaciones tanto en la evaluación de las prácticas, como en los puntos de vista computacionales.

Araujo (2021) en Lima, realizó una investigación denominada “Lenguaje de programación Scratch y pensamiento computacional en niños de 6 a 12 años, Institución Influence SAC”. Donde planteó como objetivo “determinar de qué manera el lenguaje de programación Scratch impacta en el pensamiento computacional en niños de 6 a 12 años”. A través de la metodología de un estudio de tipo experimental, bajo el enfoque cuantitativo, según sus hallazgos, logró concluir que “la enseñanza de clases de programación utilizando Scratch tuvo un buen efecto en los estudiantes y que el lenguaje de programación Scratch influye en el pensamiento computacional” (p.46).

<sup>1</sup> Chancolla y Pacori (2017) en Arequipa, en su investigación denominada “El uso del Software Scratch para mejorar el Pensamiento Computacional en los estudiantes del quinto grado de primaria de la Institución Educativa N° 40009 San Martín de Porres del Distrito de Paucarpata, Arequipa, 2016”. Plantearon como objetivo “desarrollar el pensamiento computacional a través del programa Scratch en los estudiantes de quinto grado de primaria de la Institución Educativa parroquial San Martín de Porres del distrito de Paucarpata, 2016”. Por consiguiente, emplearon la metodología experimental, bajo el enfoque cuantitativo, cuya muestra estuvo conformada por 43 alumnos. Para determinar la existencia de una mejora significativa entre estos dos puntos, lograron “diagnosticar y evaluar el desarrollo del pensamiento computacional”. Además, utilizaron sesiones de aprendizaje entre el pre y el postest, en las que se aplicó el programa "Scratch" como medio didáctico para lograr el objetivo. A través de sus resultados lograron llegar a la conclusión de que el uso del programa “Scratch” en el aula mejora el pensamiento computacional de los alumnos.

En el contexto local Condemayta (2022) en su investigación denominada “Aplicación del Software de Programación Scratch en el Desarrollo del Pensamiento Computacional de los Estudiantes del quinto grado de la Institución Educativa Primaria N° 70 116 Caritamaya, 2020”. Tuvo como objetivo “determinar la eficacia de la aplicación del software de programación Scratch en el desarrollo del pensamiento computacional en los estudiantes”. En cuanto a la metodología empleó un enfoque cuantitativo. En sus resultados demuestra que las medias de las puntuaciones de los alumnos en el pretest y el postest se distinguen por una diferencia significativa de 7,15. Por lo tanto, logró concluir que “el uso del software de programación Scratch para promover el pensamiento computacional de los niños es eficiente” (p.80).

Para conocer mejor el software educativo Scratch, primero conceptualizaremos el software educativo, según definición por autores: Al respecto, Gómez y Williamson (2018) refieren que el software educativo supone dar libertad a los estudiantes para que diseñen sus propios proyectos, lo que genera nuevas oportunidades educativas.

Según Pumario (2004) el principio didáctico de la interactividad, que permite al alumno “obtener los resultados finales de las tareas realizadas sobre las emanadas del programa (responder, señalar, elegir, errores y nuevas opciones de respuestas), ha sustituido al principio didáctico de la interactividad en el software educativo” (p.44). Desde otra perspectiva, Márquez (2016) refiere que se trata de programas informatizados cuya única finalidad es ser utilizados como herramienta de enseñanza. Gracias a ellos se consiguen importantes mejoras en la enseñanza y el aprendizaje. También se considera un producto con valor educativo.

Por lo tanto, la expresión “software educativo”, utilizada tanto por instructores y profesores especializados como por empresas <sup>37</sup> de desarrollo de software, tiene su origen en el campo de la educación y se basa en formas innovadoras de describir los resultados producidos a través de los programas empleados en el proceso de enseñanza-aprendizaje. Los programas informáticos deben haber sido creados con la intención de impartir nuevos conocimientos para que puedan considerarse educativos; esto significa que tanto su objetivo como sus características deben ser de naturaleza educativa.

En la actualidad, existe una gran variedad de paquetes de software que pretenden ser educativos y que deben reconocerse en función de sus características y de los objetivos educativos a los que pretenden servir, ya que presentan diversas características. Como resultado, se pueden encontrar las siguientes cuestiones: El software educativo debe crearse con un objetivo concreto; Debe tener componentes que dirijan el proceso de aprendizaje según la metodología; Debe diseñar entornos interactivos que permitan la interacción con el alumno; Una necesidad fundamental para el uso del alumno es la usabilidad; El alumno debe estar motivado para interesarse y comprometerse con este tipo de material educativo; Debe contener procedimientos de evaluación y retroalimentación que permitan seguir el avance de la aplicación y la consecución de los objetivos educativos deseados.

El software educativo se clasifica como: Tutor; El objetivo es desarrollar determinados contenidos de forma sistemática; Hipertextos e hipermedias, proporcionan un entorno de aprendizaje no lineal; Micro mundo, un entorno de aprendizaje cerrado, creado mediante la resolución de problemas, lo proporciona el micro mundo; Simulador, crea escenarios para el aprendizaje basados en circunstancias reales; Práctica y ejercitación, se proporcionan ejercicios para practicar y ayudar a aprender una habilidad.

El software como recurso didáctico, según Márquez (2016) son todos los recursos que los profesores emplean para apoyar y agilizar el proceso de enseñanza y aprendizaje de sus alumnos. Este componente está vinculado a los recursos que el profesor utiliza para ayudar o facilitar la enseñanza y el aprendizaje. Por otro lado, Begoña (2018) señala que “para que los niños puedan utilizar las herramientas y los programas informáticos, deben tener los conocimientos y las habilidades necesarias para reconocer, clasificar y comprender la información más útil, relacionarla con la nueva información y utilizarla en su beneficio” (p.22).

Piaget subraya la importancia de los entornos de aprendizaje para el éxito de la aplicación de los programas informáticos, que “se logra mediante el crecimiento cognitivo y las interacciones entre el sujeto y el objeto en las que el objeto llega al sujeto”. Dado que el aprendiz es quien debe ajustarse a las circunstancias novedosas y establecer la asimilación del objeto. En una línea similar, Piaget afirma que los niños deben interactuar con su entorno para adquirir habilidades intelectuales y que las etapas del desarrollo intelectual son correlativas a la edad (Begoña, 2018).

En cuanto a la importancia del software en la educación, las actividades de aprendizaje se incluyen en el software educativo, donde el discurso y el intercambio de información pueden conectarse para que el estudiante pueda trabajar a su propio ritmo mientras utiliza principalmente el entorno educativo para determinados objetivos. Según Arroyo (2006) el aspecto principal es “la interactividad que permite al alumno comunicarse e interactuar con el programa con la finalidad de desarrollar sus habilidades y conocimientos, permitiendo la utilización de diversos entornos y el aprendizaje de nuevas alternativas que permitan la adquisición de nuevos conocimientos” (p.44). De forma similar, el software de instrucción asocia el conocimiento divergente teniendo en cuenta el estilo de aprendizaje de cada alumno. Los elementos que intervienen en la comunicación deben funcionar correctamente para que se generen conocimientos.

Con respecto a las funciones de un Software educativo, el propósito del software educativo consiste en lograr impartir nuevos conocimientos, dependiendo de cómo se utilice, porque los estudiantes pueden utilizarlo para simplificar sus tareas escolares sin esforzarse por aprender. En ese sentido, Arroyo (2006) Función informativa, los programas ofrecen contenidos con información ordenada sobre la realidad a través de sus múltiples acciones. Estos recursos presentan y ordenan la realidad, como medio didáctico. Hay varios tipos de software que cumplen este propósito, como las bases de datos, los simuladores y los tutoriales; son programas que sirven más como herramientas educativas; Función instructiva, los numerosos programas educativos rigen el aprendizaje de los alumnos, ya sea directa o implícitamente, y apoyan actividades que contribuyen a la consecución de determinados objetivos educativos. También especifican el tipo de aprendizaje exhaustivo o uno secuencial; Función evaluadora, por su interactividad, “que les permite responder instantáneamente a los comentarios y actividades de los alumnos, estas herramientas son especialmente adecuadas para evaluar el trabajo realizado con ellas” (p.33). Función motivadora, como el software educativo suele incluir elementos que ayudan a captar la atención de los alumnos y a mantenerla cuando es necesario, centrándose en las partes más cruciales de la actividad, los alumnos se sienten atraídos por él; Función investigadora, los estudiantes pueden llevar a cabo investigaciones intrigantes utilizando “bases de datos, simuladores y programas informáticos de construcción buscando información específica, modificando los valores de las variables del sistema”, etc. También pueden ofrecer a educadores y alumnos recursos muy útiles para crear estudios que se realizan esencialmente fuera de línea. (Arroyo, 2006); Función expresiva, los ordenadores pueden procesar los símbolos que emplean las personas para representar y comunicar conocimientos, lo que los convierte en dispositivos con un amplio potencial expresivo. “Los alumnos se comunican con el ordenador y con sus compañeros a través de actividades programadas”. Función metalingüística, los alumnos pueden estudiar lenguajes informáticos utilizando lenguajes de programación como PYTHON y C, así como sistemas operativos como Windows, Unix y Linux; Función lúdica, con frecuencia, los alumnos asocian el uso de ordenadores en actividades de enseñanza con algo lúdico y festivo. (Arroyo, 2006)

Software educativo Scratch, es la mayor comunidad de programación infantil del mundo, “tiene una interfaz fácil de usar que fomenta el pensamiento computacional y la capacidad de resolver problemas”. Centrado en niños de entre 8 y 16 años, es accesible en más de 70 idiomas diferentes (Scratch, 2020). Al respecto, Pascual (2015) refirió como “un lenguaje de programación creado por un equipo de investigación del (MIT), cuya principal característica es un entorno de programación visual y multimedia que ofrece una interfaz gráfica de usuario para la creación de programas sencilla e intuitiva” (p.44). Al respecto, López (2020) refirió que según la herramienta Scratch, todos los que empiecen a programar por primera vez encontrarán que la programación es más placentera. (p.11). Además, precisa que esta herramienta fue creada para permitir la expresión creativa de ideas <sup>3</sup> con el fin de promover el crecimiento de las habilidades de pensamiento computacional. Según Santoyo (2016) <sup>4</sup> es un programa gratuito que puede descargarse del sitio web de Scratch (<http://scratch.mit.edu>). Está construido “para todos los sistemas operativos, su descarga e instalación son increíblemente sencillas, y su interfaz es increíblemente interactiva, lo que permite a cualquiera empezar a programar” (p.22).

<sup>1</sup> Características del software Scratch según Scratch (2020) plantea lo siguiente: **El lenguaje de programación debe ser lúdico:** “La interfaz de usuario del programa, creada para que sea lo más cómoda posible para el usuario, se basa en la programación mediante bloques de varios colores que pueden conectarse con conectores”. <sup>1</sup> **La experiencia debe ser significativa:** Pone énfasis en el trabajo individual y anima a cada usuario a trabajar en la creación de soluciones que sean individualmente significativas para él. “Los diseñadores de Scratch priorizaron la versatilidad (para que admita diversas aplicaciones, como cuentos, juegos, animaciones y simulaciones) y la personalización al crearlo (que los proyectos puedan personalizarse importando fotos, voces, gráficos)” (Scratch, 2020). <sup>1</sup> **Su uso debe propiciar la interacción social:** Con el fin de debatir, apoyar, criticar, colaborar y construir sobre el trabajo de los demás, se requiere una comunidad considerable para el desarrollo tanto de Scratch como de su sitio web. Al seleccionar “compartir”, los estudiantes publican sus proyectos en el sitio web de Scratch para compartirlos, lo que en última instancia crea un entorno de aprendizaje atractivo y gratificante para todos.

En cuanto a los entornos del software Scratch, Área de bloques: En Scratch, los bloques pueden arrastrarse al área de programación. Como resultado, se pueden dar órdenes a los objetos para que realicen diversas tareas. Área de programación: esta área está conformado por: Programas: Las instrucciones de los bloques deben arrastrarse al área de programa. Haga doble clic en un bloque para ponerlo en acción; Sonidos: Le permite ver la selección de sonidos que ha creado para los objetos. Puedes importar o grabar archivos de sonido nuevos de Scratch; Disfraces: Te da la opción de inspeccionar y modificar las máscaras de los objetos. Además, puedes esbozar nuevos disfraces, importar archivos de imagen y hacerles fotos para colocarlos en el escenario. Área de Escenario: Aquí puede ver cómo cobran vida cuentos, juegos y animaciones. Además, es donde las cosas se mueven e interactúan entre sí.

Para abordar las dimensiones del dominio del Software Scratch se toma en cuenta a Scratch (2020), el mismo que considera dos dimensiones con sus respectivos indicadores: Dimensión criterios estéticos: esta dimensión cuenta los siguientes indicadores: Experiencia Previa: Se refiere a “la colección de ideas que el usuario tenía antes de vincularse al nuevo software” Características: “El conjunto de características que diferencian el programa de los competidores crea expectativas y singularidad frente a los demás” (Scratch, 2020). Usabilidad: Un aspecto fundamental será la facilidad de uso del programa, que sea sencillo de entender y cuya introducción resulte atractiva al utilizarlo. Dimensión Criterios operacionales: esta dimensión tiene en cuenta los siguientes indicadores: Funcionalidad: “Está referido a la funcionalidad o manejo del software”. Comunidad: Scratch se utiliza en diversos lugares, como hogares, escuelas, bibliotecas, museos y centros comunitarios. La mayoría de sus usuarios son jóvenes de entre 9 y 16 años. (Scratch, 2020)

Con respecto al Pensamiento computacional, el referido término fue utilizado por primera vez en la comunidad de las ciencias de la computación por Wing (2006) quien lo define como “una forma de pensar que no se limita a los programadores y a los informáticos, sino como un conjunto de habilidades útiles para todas las personas”. Así mismo, “implica el uso de conceptos de la base de la informática para resolver problemas, diseñar sistemas y comprender el comportamiento humano” (p.56).



Desde otra perspectiva Raja (2014) señala que “el método computacional se construye sobre la base de entender el mundo como una serie de rompecabezas, que pueden descomponerse en piezas más pequeñas y resolverse poco a poco a través de la lógica y el razonamiento deductivo” (p.1). Esta es otra idea que se origina en el ámbito computacional. Denning (2017) refirió “el pensamiento computacional como el acto de articular los problemas de manera que permita presentar soluciones como pasos computacionales y algoritmos que puedan ser ejecutados eficazmente por un agente de procesamiento de información”. (párr.11)

Por otro lado, Vera et al. (2015) agregan que “para proponer la mejor solución, las personas pueden desarrollar sus habilidades de resolución de problemas mediante el uso del pensamiento computacional”. Este método se utiliza en diversos campos del conocimiento, aunque es más frecuente en algunos que en otros. Se basa en conceptos informáticos como la abstracción, los algoritmos y la programación, entre otros. (p.730). Desde su perspectiva, Román (2016) explica que “la capacidad de crear y resolver problemas utilizando nociones computacionales (secuencias, bucles, condicionales, funciones, variables), así como la adhesión a la lógica algorítmica incorporada en los lenguajes de programación informática”, se conoce como pensamiento computacional.

En tal sentido, Code learn (2021) refiere “el pensamiento computacional como un proceso mental que dirige a alguien a buscar las soluciones mejores, más rentables y abiertas” (p.44). No se limita al ámbito de la informática. Al respecto, los estudiantes de secundaria deberían trabajar su pensamiento computacional. Deben ser conscientes de que los procesos pueden racionalizarse y automatizarse mediante ordenadores para resolver problemas con mayor eficacia.

Para abordar las dimensiones del pensamiento computacional, se toma en cuenta a Brennan y Resnick (2012) quienes señalan que, Scratch “es un entorno que permite a los estudiantes crear juegos o simulaciones utilizando conceptos de programación de una manera más imaginativa y divertida” (p.24). Así mismo, se resalta el hecho de que no se haya desarrollado una definición estandarizada, los autores caracterizan dicho concepto a través de tres dimensiones.

Conceptos <sup>1</sup> computacionales, según Brennan y Resnick (2012) se refiere a las ideas que los programadores y diseñadores deben considerar. Los siguientes son sus indicadores: Secuencias: “Una idea fundamental en programación es que una tarea o actividad se define como una colección de acciones discretas o instrucciones que el ordenador puede llevar a cabo”. El orden de las instrucciones del ordenador, como en una receta, muestra la <sup>3</sup> acción que debe producirse. Ciclos: “Los ciclos son los tipos de mecanismos que ejecutan repetidamente la misma secuencia”. Eventos: Un evento que causa (desencadena) otro evento; se refiere a “cuando algo ocurre, entonces (desencadena) otro evento”. Los componentes básicos de los objetos programáticos interactivos son los eventos. Paralelismo: Se refiere al “conjunto de instrucciones que se ejecutan simultáneamente”. Así mismo, la mayor parte de los lenguajes de programación actuales admiten el paralelismo, es decir, una serie de acciones paralelas que se realizan como reacción a un evento. Condicionales: Es la capacidad de elegir acciones en función de determinadas condiciones o circunstancias. Esto confiere <sup>6</sup> flexibilidad a un programa al permitirle expresar diversos resultados en función de los parámetros. Operadores: Los programadores pueden utilizar expresiones lógicas, matemáticas y de cadena en su código gracias a los operadores. Scratch admite numerosos operadores. Datos: En un programa, implica almacenar, recuperar y actualizar valores. Por ejemplo, Scratch proporciona dos tipos diferentes de contenedores de datos: variables y listas. “Las variables se utilizan para crear marcadores en los videojuegos que actualizan automáticamente la puntuación del jugador, y las listas, se caracteriza por almacenar, recuperar y actualizar conjuntos de números” (Brennan y Resnick 2012).

Prácticas computacionales, en este sentido, las prácticas computacionales se concentran “en el proceso de aprendizaje más que en el contenido del aprendizaje, lo que el niño aprende al programar”. En otras palabras, se refieren a los procedimientos y técnicas que utilizan los alumnos para crear sus programas. Según Brennan y Resnick (2012) se refieren a las técnicas que los diseñadores aprenden mientras trabajan con ideas. Sus indicadores son los siguientes: Incremental e iterativa: El proceso de conceptualizar una idea, formular un plan para su diseño y luego aplicar ese plan en forma de código no es totalmente “limpio” y lineal cuando se diseña y programa un proyecto. A medida que se descubren soluciones en forma de código mediante métodos secuenciales incrementales paso a paso, el plan de <sup>6</sup> diseño se modifica en cambio mediante un proceso adaptativo e iterativo. Ensayo y depuración: Los programas rara

vez funcionan a la primera y exactamente como se pretendía. Es fundamental que los diseñadores-programadores tengan planes para prever y gestionar los problemas que surjan en su código. Reusar y remezclar: Crear un programa a partir de programas ya escritos. En el mundo de la programación, basarse en el trabajo de otros es una práctica común con una larga historia. Los jóvenes diseñadores-programadores pueden construir objetos digitales mucho más complicados con el apoyo de la comunidad en línea Scratch, aprendiendo a reutilizar y remezclar código existente, así como a identificar conceptos y bloques de construcción en los que basar nuevos programas. La reutilización y remezcla también fomentan la capacidad de “leer críticamente” el código de otras personas y plantean cuestiones importantes en torno a la propiedad y la autoría de los programas. Abstracter y Modularizar: Se define como el proceso de crear algo enorme añadiendo conjuntos de elementos más pequeños. Así mismo, “es bastante típico en Scratch construir proyectos muy complicados a partir de colecciones de conjuntos de código más simples” (Brennan y Resnick 2012).

<sup>1</sup> Perspectivas computacionales, según Brennan y Resnick (2012) se refiere a las opiniones que los diseñadores tienen sobre sí mismos y su entorno. Por ello, los autores incluyen un tercer componente en su modelo centrado en “los cambios de perspectiva que tienen los niños como resultado de utilizar Scratch durante su aprendizaje”. Sus indicadores son los siguientes: Expresar: Al respecto, “aunque los objetos digitales interactivos nos rodean, la mayoría de nuestras interacciones con ellos son como consumidores”. Así mismo, “un pensador computacional ve la tecnología como algo más que simples bienes de consumo; la informática se considera un medio que puede utilizarse para la autoexpresión”. Conectar: Dado que el aprendizaje y la creatividad son procesos fundamentalmente sociales, no es de extrañar que el uso de Scratch para construir medios informáticos se beneficie enormemente de la interacción social con los demás. Preguntar: Consiste en “buscar indicios de que los jóvenes no experimentan esta desconexión entre las tecnologías que les rodean y su capacidad para navegar por la realidad del mundo tecnológico utilizando la perspectiva computacional del cuestionamiento”. Así mismo, cuestionar lo que se da por sentado forma parte de hacer preguntas y en algunas circunstancias las propias preguntas se responden a propósito.

Con respecto a los de términos básicos empleados en el estudio se consideran los siguientes:

**Software educativo:** Márquez (2016) refiere que se trata de programas informatizados cuya única finalidad es ser utilizados como herramienta de enseñanza. Gracias a ellos se consiguen importantes mejoras en la enseñanza y el aprendizaje. También se considera un producto con valor educativo.

**Programa Scratch:** Según Cano y Delgado (2015) es un lenguaje de programación que simplifica el desarrollo de cuentos interactivos, juegos y animaciones, y permite compartir los trabajos con otras personas en Internet. Además, “un entorno de programación visual y multimedia construido sobre Squeak que puede utilizarse para aprender a programar, así como para crear y difundir secuencias animadas con o sin sonido” (p.66).

**Pensamiento Computacional:** Al respecto, Wing (2010) indicó que “el pensamiento computacional” es el acto de articular las cuestiones y sus respuestas de manera que estas últimas sean efectivamente representadas por un procesador de información (p.1).

**Pensamiento e Inteligencia:** Lara (2012) relaciona la idea de inteligencia con la rapidez con la que se procesa la información, la inventiva y la memoria; en la práctica, estos son factores que se utilizan para calibrar el nivel de inteligencia de una persona.

**Recursos didácticos:** Según Márquez (2016), son todos los recursos que los profesores emplean para apoyar y agilizar el proceso de enseñanza y aprendizaje de sus alumnos.

**Habilidades del pensamiento computacional:** Vera (2018) define las habilidades de pensamiento computacional como aquellas cualidades personales que permiten a un individuo abordar un problema utilizando el pensamiento computacional como guía.

**Alfabetización digital:** Según Lamoth et al. (2020) se refiere a las iniciativas educativas destinadas a enseñar “el uso crítico, eficaz y eficiente de las tecnologías digitales y su aplicación en determinadas situaciones, permitiendo a los alfabetizados resolver problemas de la vida cotidiana y mejorar su rendimiento en el uso de las TIC”. (p.196).

**Secuencia Didáctica:** De acuerdo a la definición se conoce como secuencia didáctica a una serie de actividades planificadas por el profesor que dirige el aprendizaje de un tema concreto por parte de los alumnos.

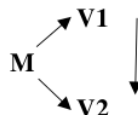
## 4 II. METODOLOGÍA

### 2.1. Enfoque, tipo y diseño de investigación

El presente estudio emplea el enfoque cuantitativo, porque se logró medir con precisión el comportamiento de cada una de las variables a través de un proceso determinado y estricto. Al respecto, según Hernández et al. (2014) “La investigación cuantitativa asume que el conocimiento debe ser objetivo, resultado de un proceso deductivo en el que se prueban hipótesis formuladas usando mediciones numéricas y análisis estadístico secuencial” (p.152).

Esta investigación es de tipo básica, No experimental, de corte transversal. Al respecto, Hernández et al. (2014) refiere “una investigación que se lleva a cabo sin influir intencionadamente en las variables. En otras palabras, se trata de un estudio en el que no cambiamos a propósito los factores para ver cómo afectan a los demás” (p.152).

Corresponde al diseño correlacional, debido a que se pretende establecer el grado de relación que existe entre las dos variables de estudio. Al respecto, Hernández-Sampieri y Mendoza (2018) refieren que “las investigaciones correlacionales pretenden aportar respuestas a la investigación con el fin de evaluar la relación o el nivel de asociación entre conceptos, categorías o variables en una determinada muestra o entorno” (p.93). Se grafica de la siguiente forma:



V1 = Software Educativo Scratch

V2 = Pensamiento Computacional

→ = Relación que existe entre las variables

Por otro lado, se empleó los siguientes métodos: Método deductivo: porque parte del reconocimiento de los problemas a escala mundial, nacional y local, lo que permitió recopilar y analizar los datos. Método transversal: puesto que los datos se recogieron y procesaron mediante programas en un único momento determinado. Método descriptivo: lo cual permitió el agrupamiento para un correcto análisis e interpretación de los hallazgos determinados.

### Cuadro de variables e indicadores

Variable	Dimensión	Indicador
<b>Variable 1</b> Software Educativo Scratch	1.1. Criterios estéticos	<ul style="list-style-type: none"> <li>- Experiencia Previa</li> <li>- Características</li> <li>- Usabilidad</li> </ul>
	1.2. Criterios operacionales	<ul style="list-style-type: none"> <li>- Funcionalidad</li> <li>- Comunidad</li> </ul>
<b>Variable 2</b> Pensamiento computacional	14 2.1. Conceptos computacionales	<ul style="list-style-type: none"> <li>- Secuencias</li> <li>- Ciclos</li> <li>- Eventos</li> <li>- Paralelismo</li> <li>- Condicionales</li> <li>- Operadores</li> <li>- Datos</li> </ul>
	2.2. Prácticas computacionales	<ul style="list-style-type: none"> <li>- Incremental e iterativa</li> <li>- Ensayo y depuración</li> <li>- Reusar y remezclar</li> <li>- Abstractar y Modularizar</li> </ul>
	2.3. Perspectivas computacionales	<ul style="list-style-type: none"> <li>- Expresar</li> <li>- Conectar</li> <li>- Preguntar</li> </ul>

### 2.2. Población, muestra y muestreo

**Población**, según Bernal (2010) la población es “el conjunto de componentes o individuos que comparten un conjunto de rasgos y de los que se quiere extraer un análisis o inferencia” (p.68). La población en estudio estuvo conformada por 893 alumnos desde primero a quinto grado de la IES José Gálvez de Yunguyo, caracterizado según sección y grado de la siguiente forma:

**Tabla 1**

*Conformación de la población de estudio*

Grado	A	B	C	D	E	F	G	H	Matriculados	Traslados	Regulares
Primero	22	21	23	20	22	19	22	19	168	3	165
Segundo	26	25	26	25	24	21	21	19	187	4	<b>183</b>
Tercero	28	27	26	25	26	27	24	21	204	14	190
Cuarto	25	27	25	26	25	19	16	16	179	8	171
Quinto	26	25	25	25	24	23	25	21	194	10	184
Total estudiantes									932	39	893
Total estudiantes regulares											893

*Nota.* Esta tabla demuestra la conformación de la población total de estudio.

**Muestra:** en el presente estudio se consideró a través del muestreo Probabilístico aleatorio simple a los 183 estudiantes del segundo grado de la institución en mención, según criterio de experiencia previa de conocimiento del Software Scratch por parte del estudiante. Aplicando la formula se obtiene lo siguiente:

$$n = \frac{N*Z^2*p*q}{e^2*(N-1)+Z^2*p*q}$$

**7**  
**Donde:**

n: Muestra

N: Población.

Z: Nivel de confianza (1.96)

e: Precisión o el error

p: Probabilidad de éxito (0.5)

q: Probabilidad de fracaso (0.5)

**Cálculo de la muestra**

N = 183

Z = 1,96

e = 0,05

p = 0.5

q = 0.5

**Reemplazando:**

$$n = \frac{183*1.96^2*0.5*0.5}{0.05^2*(183-1)+1.96^2*0.5*0.5}$$

$$n = 124$$

En cuanto a los criterios de inclusión, se consideraron lo siguiente: Estudiantes del segundo grado que tengan experiencia en el manejo del Scratch; Estudiantes que sean elegidos a través de la probabilidad; Estudiantes con actitud de colaborar y o participar. Por otro lado, los criterios de exclusión: Estudiantes ausentes o que faltaron a clases; Estudiantes que no cuenten con experiencia del software Scratch.

### **5** **2.3. Técnicas e instrumentos de recojo de datos**

La técnica de recolección de datos fue la **encuesta**, cuyo instrumento un **cuestionario**. al respecto, Charaja (2011) refiere que “son pasos o secuencias que hay que completar para reunir la información necesaria para probar la hipótesis principal o apoyar la postura que hemos adoptado” (p.43).

**Para la variable 1:** dominio del Software Scratch, el instrumento fue un cuestionario adaptado del autor (Taco, 2019). Así mismo, este instrumento fue validado por el autor a través de juicio de expertos y mediante la fiabilidad de Alpha de Cronbach de (0.826) presenta una confiabilidad alta. Por lo tanto, está diseñado para la evaluación de la “Dimensión Criterios estéticos con indicadores experiencia previa, características y usabilidad. Además, de la Dimensión Criterios operacionales conformado por indicadores comunidad y funcionalidad”. Conformando un total de 10 ítems con alternativas de elección múltiple (donde una sola respuesta es correcta).

**Para la variable 2:** pensamiento computacional, el instrumento fue adaptado de Arranz y Pérez (2017) denominado “Cuestionario para conocer la percepción del alumnado sobre la adquisición del Pensamiento Computacional”. Así mismo, está conformado por 14 ítems distribuidos en 3 dimensiones, con una escala de calificación tipo Likert cuya escala son de (1 – 5). Por lo tanto, este instrumento presenta una validez a través de juicio de experto y mediante la estadística de fiabilidad Alpha de Cronbach de (0.840) presenta una confiabilidad alta.

**Tabla 2**

*Intervalos para la categorización del dominio del programa Scratch*

Variable	Dimensión	Reactivos	Descripción	Escala Baremos
Software Educativo Scratch	Criterios estéticos	1-2-3-7	Respuesta correcta (2 puntos)	Nivel (0 – 8) bajo
	Criterios operacionales	5-6-8-9-10-4	Respuesta incorrecta (0 puntos)	(9 – 14) regular (15 – 20) alto

*Nota.* Intervalo para categorización del dominio del Scratch.

**Tabla 3**

*Intervalos para la categorización del nivel del pensamiento computacional*

Dimensión	Conceptos computacionales	Practicadas computacionales	Perspectivas computacionales	Pensamiento computacional
Nivel	Intervalo	Intervalo	Intervalo	Intervalo
Básico	(7 – 16)	(4 – 9)	(3 – 7)	(14 – 33)
Medio	(17 – 26)	(10 – 15)	(8 – 11)	(34 – 51)
Alto	(27 – 35)	(16 – 20)	(12 – 15)	(52 – 70)

*Nota.* Intervalo para categorización del nivel del pensamiento computacional.



## 2.4. Técnicas de procesamiento y análisis de la información

**La estadística inferencial:** Esta técnica permitió establecer los análisis inferenciales para determinar la relación existente entre las variables. Al respecto, Hernández-Sampieri y Mendoza (2018) recomiendan que, para los estudios correlacionales, “es necesario realizar la prueba de normalidad para determinar el estadígrafo de prueba de correlación a emplear” (p.156).

**La estadística descriptiva:** Esta técnica permitió agrupar los datos a través de tablas y figuras para una mejor presentación e interpretación.

**Prueba de normalidad:** Al respecto, “cuando la muestra es < a 50, se debe tomar los resultados de Shapiro-Wilk; mientras que, para una muestra de 50 a más, se debe utilizar el resultado de Kolmogorov-Smirnov”. Hernández et al. (2014)

**Tabla 4**

**2** Prueba de normalidad para las variables

Variable	Pruebas de normalidad					
	Kolmogorov-Smirnov <sup>a</sup>			Shapiro-Wilk		
	Estadístico	gl	Sig.	Estadístico	gl	Sig.
Dominio del Scratch	,363	124	,000	,676	124	,000
Nivel del pensamiento computacional	,456	124	,000	,572	124	,000

a. Corrección de significación de Lilliefors

Nota. Resultado del análisis de pruebas.

**1** Se observa en la Tabla 4, la prueba estadística de normalidad para determinar la integridad del ajuste de la distribución de las dos variables: Dado que utilizamos una muestra de 124 estudiantes. “Para determinar el nivel de correlación, asociación y significancia entre las variables, se emplea el coeficiente de correlación Rho de Spearman”. Esto se debe a que el valor Sig. es (0,000) y (0,000), ambos inferiores a (0,05), lo que indica que la muestra es no paramétrica.

**1** **Nivel de correlación:** Mondragón (2014), planteó los niveles o grados de correlación bilateral “caracterizado para estudiar muestras no paramétricas y muestras mixtas con la finalidad de conocer el grado de correlación, asociación y significancia” (p.46).

**Tabla 5***Nivel de correlación bilateral*

Valor	Correlación negativa
-1	Grande y perfecta
-0,9 a -0,99	Muy alta
-0,7 a -0,89	Alta
-0,4 a -0,69	Moderada
-0,2 a -0,39	Baja
-0,01 a -0,19	Muy baja
0	la
0,01 a 0,19	Correlación positiva muy baja
0,2 a 0,39	Baja
0,4 a 0,69	Moderada
0,7 a 0,89	Alta
0,9 a 0,99	Muy alta
1	Grande y perfecta

*Nota.* La tabla demuestra los intervalos para interpretar el nivel de correlación.

Con respecto a la contrastación de hipótesis, se planteó la hipótesis estadística: (H1) hipótesis alterna; (H0) hipótesis nula, con nivel de significancia de  $\alpha = 0.05 = 5\%$  a través del Estadígrafo de Rho de Spearman y P valor Sig. Bilateral, cuya regla para la decisión se basa en: Si P valor sig.  $< 0.05 =$  se rechaza la hipótesis nula y/o Si P valor sig.  $> 0.05 =$  se acepta la hipótesis nula. Para el procesamiento y análisis de datos se siguió la siguiente secuencia: Clasificación de datos, según dimensión y objetivo; Codificación de los datos para el análisis; Calificación y o ponderación según escala; Tabulación estadística; Interpretación, análisis de datos y presentación de resultados.

## 2.5. Ética investigativa

El presente estudio se llevó a cabo teniendo en cuenta las pautas éticas proporcionadas por los principios rectores de la investigación (Hernández y Mendoza 2018). Los lineamientos éticos para los estudios son la autenticidad, originalidad, confidencialidad y el permiso informado de las partes. También se aplican al presente estudio las normas legales vigentes establecidas por la Universidad Católica de Trujillo para los trabajos de investigación en el Programa de Complementación Pedagógica Universitaria. Además, tanto el proyecto como el informe fueron sometido a la verificación de índice de similitud y originalidad a través del Software Turniting. Así mismo, como respaldo a las ideas propias de autores, se les referencia mediante el gestor Mendeley en estilo APA 7 Edición.

### III. RESULTADOS

**Tabla 6**

*Identificación de las características generales de la población de estudio*

Género	Sección	Edad			Total
		12 años	13 años	14 años a más	
Mujer	Abraham Valdelomar	0	5	4	9
	Carlos Oquendo de Amat	0	0	5	5
	Ciro Alegría	1	4	4	9
	Dante Nava	0	4	5	9
	Gamaliel Churata	0	2	5	7
	Javier Heraud	0	4	5	9
	Mariano Melgar	0	3	3	6
	Ricardo Palma	0	1	9	10
	Total	1	23	40	64
Hombre	Abraham Valdelomar	2	2	2	6
	Carlos Oquendo de Amat	0	2	9	11
	Ciro Alegría	0	3	4	7
	Dante Nava	0	4	2	6
	Gamaliel Churata	0	1	7	8
	Javier Heraud	0	3	4	7
	Mariano Melgar	0	7	3	10
	Ricardo Palma	0	2	3	5
	Total	2	24	34	60

*Nota.* Tomado de los resultados del instrumento de investigación.

Se evidencia con respecto a las características generales de los participantes: con respecto al género, del total de los encuestados el 51.6% pertenecen al género femenino donde la mayor parte tiene 14 años a más. En cuanto a la sección de estudios, la mayor parte de los estudiantes pertenecen a la sección Ricardo Palma. Así mismo, el 48.4% pertenece al género masculino donde la mayor parte tienen 14 años a más. En cuanto a la sección de estudios, en su mayoría pertenecen a la sección Carlos Oquendo de Amat.

**Tabla 7**

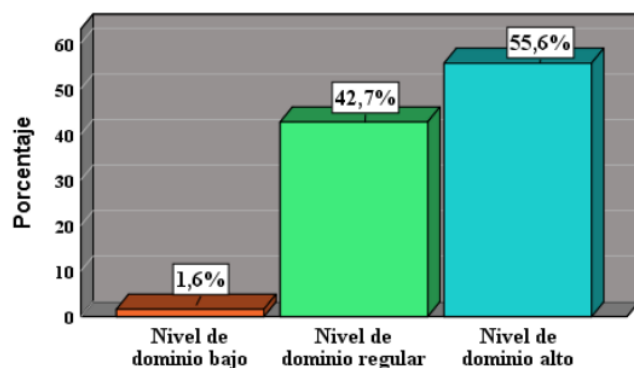
**3** *Nivel de dominio del Scratch de los estudiantes de la IES José Gálvez de Yunguyo*

Nivel de dominio	Frecuencia	Porcentaje
Nivel de dominio bajo	2	1,6%
Nivel de dominio regular	53	42,7%
Nivel de dominio alto	69	55,6%
Total	124	100%

*Nota.* Tomado de los resultados del instrumento de investigación.

**Figura 1**

*Frecuencia porcentual del dominio del Scratch de los estudiantes*



Se evidencia en la tabla 7 y figura 1 con respecto al dominio del Software académico Scratch. Del total de los estudiantes el 55.6% presenta un dominio alto; el 42.7% presenta un dominio regular y el 1.6% presenta un dominio bajo. Por lo tanto, se logró identificar que los estudiantes presentan un nivel de dominio alto con el 55.6%.

**Tabla 8**

*Indicadores de desempeño del dominio del Scratch*

Variable			Respuesta					
Dimensiones	Indicadores	Ítems	Incorrecto		Correcto		Total	
			N	%	N	%	N	%
Criterios estéticos	Experiencia P.	P1	14	11.3	110	88.7	124	100
	Características	P2	39	31.5	85	68.5	124	100
	Usabilidad	P3	35	28.2	89	71.8	124	100
	Usabilidad	P7	45	36.3	79	63.7	124	100
Criterios operacionales	Funcionalidad	P5	56	45.2	68	54.8	124	100
	Funcionalidad	P6	2	1.6	122	98.4	124	100
	Funcionalidad	P8	12	9.7	112	90.3	124	100
	Funcionalidad	P9	35	28.2	89	71.8	124	100
	Funcionalidad	P10	7	5.6	117	94.4	124	100
	Comunidad	P4	56	45.2	68	54.8	124	100
<b>Total</b>			<b>301</b>	<b>24.3</b>	<b>939</b>	<b>75.7</b>	<b>1240</b>	<b>100%</b>

*Nota.* Tomado de los resultados del instrumento de investigación

Se evidencia a través de los resultados estadísticos con relación a los indicadores evaluados para determinar el dominio del Software Scratch. En la dimensión criterios estéticos; los estudiantes mostraron un mejor desempeño en el indicador experiencia

previa, donde el 88.7% de los estudiantes respondieron de manera correcta. Esto indica que la mayor parte de los estudiantes tienen conocimiento y experiencia con el programa Scratch; en el indicador características, el 68.5% respondió de manera correcta, lo que significa que los estudiantes tienen conocimiento sobre lo que se puede crear con Scratch; en el indicador usabilidad, más del 50% respondieron de forma correcta, lo que significa que los estudiantes tienen conocimiento acerca de **lo que se requiere para crear en Scratch**. Sin embargo, un 36.3% respondieron de manera incorrecta en cuanto al indicador usabilidad, referido al conocimiento sobre **los botones que forman parte de la Barra de Herramientas**. Por otro lado, en la dimensión criterios operacionales, los estudiantes mostraron un mejor desempeño en el indicador funcionalidad, referido al manejo del sonido con Scratch, donde el 98.4% respondieron correctamente. Sin embargo, un 45.2% presentan deficiencias en el indicador comunidad, evidenciando que los estudiantes desconocen que a través del Scratch se puede crear historias interactivas.

**Tabla 9**

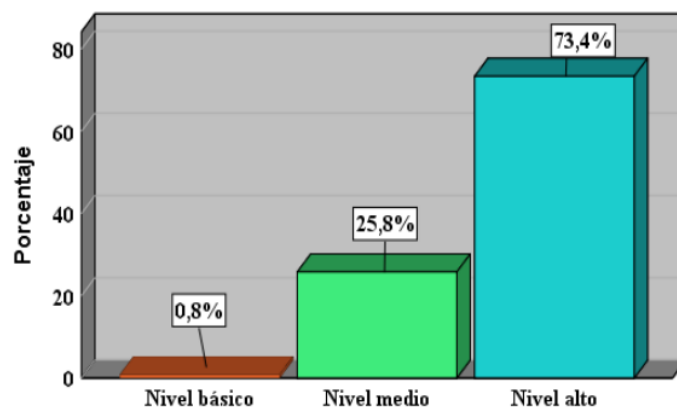
**3**  
*Nivel del pensamiento computacional de los estudiantes*

Nivel de pensamiento	Frecuencia	Porcentaje
Nivel básico	1	0,8%
Nivel medio	32	25,8%
Nivel alto	91	73,4%
Total	124	100%

*Nota.* Tomado de los resultados del instrumento de investigación.

**Figura 2**

**4**  
*Frecuencia porcentual del nivel de pensamiento computacional de los estudiantes*



Se evidencia en la tabla 9 y figura 2 respecto al nivel del pensamiento computacional desarrollado por los estudiantes; catalogados con experiencia previa del Software Scratch; donde el 73.4% manifiestan un nivel Alto; el 25.8% Medio y el 0.8% Básico. Predominando un desarrollo alto de pensamiento computacional con el 73.4%.

**Tabla 10**

*Indicadores de nivel de desarrollo del aprendizaje percibido según dimensiones*

Dimensión	Nivel de desarrollo percibido							
	Básico		Medio		Alto		Total	
	N	%	N	%	N	%	N	%
Conceptos C.	2	1,6	39	31,5	83	66,9	124	100
Prácticas C.	1	,8	61	49,2	62	50,0	124	100
Perspectivas C.	1	,8	34	27,4	89	71,8	124	100

*Nota.* Tomado de los resultados del instrumento de investigación.

Se evidencia a través de la tabla 10 el nivel del desarrollo del aprendizaje percibido que presenta después de la experiencia con el programa Scratch; En la dimensión conceptos computacionales; el 66.9% perciben un desarrollo alto; el 31.5% medio y el 1.6% básico. Así mismo, los estudiantes mostraron mejores aprendizajes en los indicadores: secuencia y ciclos, lo que significa que los estudiantes lograron entender el concepto de una secuencia de instrucciones. Así mismo, el concepto de los ciclos y su utilidad. Sin embargo, existe cierto desconocimiento en el indicador datos, referido a la opción almacenamiento y recuperación de información. En cuanto a la dimensión practicas computacionales; el 50.0% perciben un desarrollo de aprendizaje en nivel alto; el 49.2% medio y el 0.8% básico. Así mismo, los estudiantes mostraron mejores aprendizajes en los indicadores: (incremental e iterativa) y (ensayo y depuración). Lo que significa que los estudiantes lograron desarrollar el aprendizaje sobre búsqueda de solución poco a poco, probando opciones y adaptándolas en cada momento. Así mismo, lograron desarrollar el aprendizaje sobre búsqueda y detección de algún problema, realizando diversas pruebas. Sin embargo, existe cierta deficiencia en la práctica sobre reusar y remezclar algunas instrucciones. En cuanto a la dimensión perspectivas computacionales; el 71.8% perciben un desarrollo alto; el 27.4% medio y el 0.8% básico. Así mismo, mostraron mejores aprendizajes en el indicador perspectiva de expresión logrando desarrollar eficientemente la capacidad creativa, la autoexpresión, la producción y el diseño. Sin embargo, existe deficiencias en cuanto al cuestionamiento.

## Resultados para el objetivo general

**Tabla 11**

**6** Grado de *relación entre el dominio del Scratch y el pensamiento computacional*

**4**

Correlaciones - Rho de Spearman		
		Pensamiento computacional
Dominio del Scratch	Coeficiente de correlación	<b>,326**</b>
	Sig. (bilateral)	<b>,000</b>
	N	124

**2** *Nota. La correlación es significativa en el nivel 0,01 (bilateral).*

En la tabla 11 se evidencia el valor de Rho = (0,326), cuyo resultado indica que “existe una correlación – positiva baja entre la variable dominio del Scratch y la variable nivel del pensamiento computacional de los estudiantes”. Así mismo, estadísticamente **5** se identificó que el 55.6% de los estudiantes presentan un nivel de dominio del Scratch alto y en cuanto al pensamiento computacional el 73.4% manifiestan un nivel Alto, evidenciando una relación directa proporcional entre las variables.

### **8** **Contrastación de hipótesis**

#### **Hipótesis estadística**

- **(H1):** **3** Existe una relación directa entre el dominio del software educativo Scratch y el nivel del pensamiento computacional de los Estudiantes de la IES José Gálvez de Yunguyo.
- **(H0):** **9** No existe una relación directa entre el dominio del software educativo Scratch y el nivel del pensamiento computacional de los Estudiantes de la IES José Gálvez de Yunguyo.

#### **Decisión**

De acuerdo al P valor Sig. = (0,000) < (0,05), corresponde rechazar la H0 y aceptar la (H1), lo cual se interpreta que “Existe relación directa entre el dominio del software educativo Scratch y el nivel del pensamiento computacional de los Estudiantes de la IES José Gálvez de Yunguyo”. Es decir, “a medida que el estudiante mejore el dominio del Scratch, esto se reflejará en su desarrollo del nivel de pensamiento computacional”.

## Resultados para el objetivo 01

**Tabla 12**

*Grado de relación entre el dominio del Scratch y conceptos computacionales*

Correlaciones - Rho de Spearman		Conceptos Computacionales
Dominio del Scratch	Coefficiente de correlación	,210*
	Sig. (bilateral)	,020
	N	124

*Nota.* La correlación es significativa en el nivel 0,01 (bilateral).

En la tabla 12 se evidencia el valor de  $Rho = (0,210)$ , cuyo resultado indica que “existe una correlación – positiva baja entre el dominio del Scratch y el desarrollo de conceptos computacionales”. Así mismo, estadísticamente se identificó una relación directa proporcional donde predomina el dominio alto con el (55.6%) y en cuanto al desarrollo de conceptos computacionales predomina un desarrollo alto con el (66.9%). Evidenciando que la deficiencia existente en el dominio del Scratch repercute en el desarrollo de aprendizaje sobre conceptos computacionales.

### Contrastación de hipótesis

#### Hipótesis estadística

- **(H1):** Existe una relación directa entre el dominio del Software Educativo Scratch y el desarrollo de conceptos computacionales de los estudiantes.
- **(H0):** No existe una relación directa entre el dominio del Software Educativo Scratch y el desarrollo de conceptos computacionales de los estudiantes.

#### Decisión

De acuerdo al P valor  $Sig. = (0,020) < (0,05)$  corresponde rechazar la  $H_0$  y aceptar la  $(H_1)$ . Lo cual se interpreta que “existe una relación directa entre el dominio del Scratch y desarrollo de conceptos computacionales de los Estudiantes”. Es decir, “a medida que el estudiante mejore el dominio del Scratch, también mejorará su desarrollo de aprendizaje sobre conceptos computacionales.



## Resultados para el objetivo 02

Tabla 13

Grado de relación entre el dominio del Scratch y prácticas computacionales

Correlaciones - Rho de Spearman		
Dominio del Scratch	Coefficiente de correlación	,321**
	Sig. (bilateral)	,000
	N	124

Nota. La correlación es significativa en el nivel 0,01 (bilateral).

En la tabla 13 se evidencia el valor de Rho = (0,321) cuyo resultado indica que “existe una correlación – positiva baja entre la variable dominio del Scratch y el desarrollo de prácticas computacionales en los estudiantes”. Así mismo, los resultados estadísticos indican que los estudiantes que poseen un nivel de dominio regular, presentan un desarrollo de aprendizaje en nivel medio con el (49.2%); Así mismo, los estudiantes que poseen un dominio alto, presentan un desarrollo alto de prácticas computacionales con el (50.0%). Esto evidencia la existencia de una relación directa proporcional entre el dominio del Scratch y el desarrollo de las prácticas computacionales.

### Contrastación de hipótesis

#### Hipótesis estadística

- (H1): Existe una relación directa entre el dominio del Software Educativo Scratch y el desarrollo de prácticas computacionales de los estudiantes.
- (H0): No existe una relación directa entre el dominio del Software Educativo Scratch y el desarrollo de prácticas computacionales de los estudiantes.

#### Decisión

De acuerdo al P valor Sig. = (0,000) < (0,05): corresponde rechazar la H0 y aceptar la (H1), lo cual se interpreta que “Existe una relación directa entre el dominio del software educativo Scratch y desarrollo de prácticas computacionales de los Estudiantes de la IES José Gálvez de Yunguyo”. Es decir, una mejora en el dominio del Scratch permitirá un mejor desarrollo del aprendizaje de prácticas computacionales.

### Resultados para el objetivo 03

**Tabla 14**

*Grado de relación entre el dominio del Scratch y perspectivas computacionales*

Correlaciones - Rho de Spearman	
Dominio del Scratch	Perspectivas Computacionales
Coefficiente de correlación	,169
Sig. (bilateral)	,061
N	124

**2** Nota. La correlación es significativa en el nivel 0,01 (bilateral).

En la tabla 14 se evidencia el valor de Rho = (0,169), cuyo resultado indica que “existe una correlación – positiva muy baja entre el dominio del Scratch y el desarrollo de perspectivas computacionales”. Así mismo, los resultados estadísticos indican que existe una relación proporcional en la categoría alto, entre el dominio y el desarrollo de la perspectiva computacional. Sin embargo, las demás categorías no establecen relación directamente proporcional.

#### **8** Contrastación de hipótesis

##### **Hipótesis estadística**

- **(H1):** **1** Existe una **relación directa** entre el dominio del Software Educativo Scratch y el desarrollo de perspectivas computacionales de los estudiantes.
- **(H0):** **9** No existe una **relación directa** entre el dominio del Software Educativo Scratch y **1** el desarrollo de perspectivas computacionales de los estudiantes.

##### **Decisión**

De acuerdo al P valor Sig. = (0,061) > **26** (0,05) se acepta la H0 y se rechaza la (H1), lo cual se interpreta que “No existe una relación directa entre el dominio del software educativo Scratch y desarrollo de perspectivas computacionales de los Estudiantes de la IES José Gálvez de Yunguyo”. Es decir, el dominio del Scratch no determina en su totalidad para el desarrollo del aprendizaje de las perspectivas computacionales.

#### IV. DISCUSIÓN

En el presente estudio se logró determinar la existencia de una relación positiva baja entre el dominio del software educativo Scratch y el nivel del pensamiento computacional de los Estudiantes de la IES José Gálvez de Yunguyo. Esta afirmación se basa de acuerdo a lo establecido en la (tabla 11) donde se evidencia el valor de  $Rho = 0,326$  y  $P \text{ valor Sig.} = (0,000) < 0,05$ . Lo cual significa que, si se mejora el dominio del Scratch, ésta se verá reflejado en el desempeño del aprendizaje del pensamiento computacional de los estudiantes. Por lo tanto, se concluye y queda demostrado que el Programa Scratch contribuye en el desarrollo del aprendizaje del pensamiento computacional. Por otro lado, con relación a los indicadores evaluados: los estudiantes mostraron un mejor desempeño en el indicador experiencia previa, con el 88.7%; características, con el 68.5%; funcionalidad con el 98.4%. Sin embargo, un 36.3% de los estudiantes desconocen sobre el uso de botones de la Barra de Herramientas y un 45.2% desconocen que a través del Scratch se puede crear historias interactivas.

Estos resultados establecen coincidencia parcial con los hallazgos de Condemayta (2022) quien logró concluir que “el uso del software de programación Scratch para promover el pensamiento computacional de los niños es eficiente” (p.80). Así mismo, se coincide con Araujo (2021) donde “la enseñanza de clases de programación utilizando Scratch tuvo un buen efecto en los estudiantes y que el lenguaje de programación Scratch influye en el pensamiento computacional” (p.46). A su vez Churata et al. (2021) lograron concluir que el uso del Scratch “tiene un impacto positivo en el desarrollo del pensamiento computacional de los alumnos”. De igual forma Arranz de la Fuente y Pérez (2017) lograron demostrar que el uso de Scratch influye en el crecimiento de los conceptos, las prácticas y los puntos de vista del pensamiento computacional en estudiantes con experiencia en el manejo del Scratch.

Lo antes mencionado refuerza los aportes teóricos realizados por Márquez (2016) quien refirió que los programas educativos informatizados tienen como finalidad de ser utilizados como herramienta educativos para conseguir importantes mejoras en la enseñanza y el aprendizaje. Así mismo, en cuanto a la importancia de los entornos de aprendizaje se confirma lo establecido por Piaget, quien indicó que el éxito de la aplicación de los programas informáticos, “se logra mediante el crecimiento cognitivo y las interacciones entre el sujeto y el objeto en las que el objeto llega al sujeto”.

En ese sentido, lo mencionado refuerza la contribución teórica de Begoña (2018) quien señaló que “para que los niños puedan utilizar las herramientas y los programas informáticos, deben tener los conocimientos y las habilidades necesarias para reconocer y relacionarla con la nueva información y utilizarla en su beneficio” (p.22). Por otro lado, desde la perspectiva del nuevo enfoque de enseñanza mediante la incorporación de sistemas digitales interactivos en la educación, se respalda la importancia de un dominio significativo del Scratch para contribuir un mejor desarrollo de capacidades, competencias y habilidades del estudiante donde Scratch ofrece como medio alternativo para estimular y desarrollar diversas inteligencias, como la lingüística, espacial, musical, interpersonal, corporal, lógico-matemática y la intrapersonal. Además, inspira a los estudiantes durante el proceso de aprendizaje y fomenta el desarrollo de la creatividad. Esto nos permite corroborar la afirmación de Serna et al. (2018) que la aplicación Scratch es un método para utilizar la tecnología con un enfoque pedagógico.

Con respecto al objetivo 01, se logró determinar la existencia de una relación positiva baja entre el dominio del software educativo Scratch y el desarrollo de conceptos computacionales de los Estudiantes de la IES José Gálvez de Yunguyo. Esta afirmación se basa de acuerdo a lo establecido en la (tabla 12) donde se evidencia el valor de Rho = 0,210 y P valor Sig. = (0,020) < 0,05. Así mismo, estadísticamente se identificó una relación directa proporcional donde predomina el dominio alto con el (55.6%) y en cuanto al desarrollo de conceptos computacionales predomina un desarrollo alto con el (66.9%). Evidenciando que la deficiencia existente en el dominio del Scratch repercute en el desarrollo de aprendizaje sobre conceptos computacionales. Es decir, los estudiantes mostraron mejores aprendizajes sobre el concepto de la utilidad de secuencias y ciclos. Sin embargo, existe cierto desconocimiento sobre la opción de almacenamiento y recuperación de información.

Con respecto al objetivo 02, se logró determinar la existencia de una relación positiva baja entre el dominio del software educativo Scratch y el desarrollo de prácticas computacionales de los Estudiantes de la IES José Gálvez de Yunguyo. Esta afirmación se basa de acuerdo a lo establecido en la (tabla 13) donde se evidencia el valor de Rho = 0,321 y P valor Sig. = (0,000) < 0,05. Así mismo, los resultados estadísticos indican que los estudiantes que poseen un nivel de dominio regular, presentan un desarrollo de aprendizaje en nivel medio con el (49.2%); Así mismo, los estudiantes que poseen un

dominio alto, presentan un desarrollo alto de prácticas computacionales con el (50.0%). Esto evidencia <sup>2</sup> la existencia de una relación directa proporcional entre el dominio del Scratch y el desarrollo de las prácticas computacionales. es decir, los estudiantes mostraron mejores aprendizajes en los indicadores sobre búsqueda de solución poco a poco, probando opciones y adaptándolas en cada momento. Así mismo, sobre búsqueda y detección de algún problema, realizando diversas pruebas. Sin embargo, existe cierta deficiencia en la práctica sobre reusar y remezclar algunas instrucciones.

Con respecto al objetivo 03, <sup>31</sup> se logró determinar la existencia de una relación positiva muy baja <sup>1</sup> entre el dominio del software educativo Scratch y el desarrollo de perspectivas computacionales de los Estudiantes de la IES José Gálvez de Yunguyo. Esta afirmación se basa de acuerdo a lo establecido en la (tabla 14) donde se evidencia el valor de  $Rho = 0,169$  y  $P \text{ valor Sig.} = (0,061) > 0,05$ . Lo cual se interpreta que “No existe una relación directa entre el dominio del software educativo Scratch y desarrollo de perspectivas computacionales de los Estudiantes de la IES José Gálvez de Yunguyo”. Es decir, el dominio del Scratch no determina en su totalidad para el desarrollo del aprendizaje de las perspectivas computacionales. Así mismo, los estudiantes mostraron mejores aprendizajes en el desarrollo eficiente de la capacidad creativa, la autoexpresión, la producción y el diseño. Sin embargo, aún presentan deficiencias en cuanto a la perspectiva del cuestionamiento.

Estos resultados establecen coincidencia con <sup>13</sup> los hallazgos obtenidos por Gamito et al. (2022) donde llegaron demostrar que “la experiencia en el manejo del Scratch tuvo una valoración significativa y su uso contribuye en el desarrollo de procesos de pensamiento computacional en estudiantes de Educación Primaria” (p.71). De igual forma, se confirma lo mencionado por Ílic (2021) donde los maestros en formación afirmaron que “las aplicaciones Scratch contribuyen en la adquisición de habilidades de pensamiento computacional” (p.46). Así mismo, Condemayta (2022) logró revelar que “el uso eficiente del software de programación Scratch contribuye para promover el pensamiento computacional de los niños” (p.80). A su vez Kyza et al. (2022) lograron revelar que el dominio del apoyo instructivo es necesario para apoyar <sup>1</sup> el desarrollo de las prácticas de codificación y el pensamiento computacional de los niños de primaria.

Esto permite corroborar lo determinado por Barahona et al. (2022) de que es necesario incluir contenidos curriculares relacionados con el pensamiento computacional independientemente de los elementos sociales, culturales y académicos que influyan en el alumno. De igual forma, Kyza et al. (2022) consideraron que el apoyo instructivo es necesario para contribuir en el desarrollo de las prácticas de codificación y el pensamiento computacional de los estudiantes. A su vez se coincide con Taco (2019) quien logró evidenciar en los estudiantes un desempeño regular y altas puntuaciones en la evaluación de las prácticas y perspectivas computacionales.

En tal sentido, Adelmo et al. (2022) con respecto al pensamiento computacional, lograron proponer un enfoque de evaluación basado en datos utilizando el lenguaje de programación Scratch. Por lo tanto, concluyen que a través de la programación Scratch se puede mejorar y orientar el diseño de los cursos, así como también apoyar estudios sobre la evaluación del pensamiento computacional. En la misma línea, Selamí y Gülhanım (2021) al observar las percepciones de los estudiantes sobre Tinkercad, lograron determinar que los estudiantes estaban muy motivados por el interés y la apreciación debido a que Tinkercad era generalmente útil y fácil de usar. Así mismo, lograron determinar que la percepción de los estudiantes sobre el pensamiento computacional fue moderada. Por lo tanto, lograron concluir que “existe una relación positiva de nivel moderado entre su percepción de Tinkercad y sus habilidades de pensamiento computacional” (p.35). En ese sentido, lo mencionado refuerza el aporte teórico establecido por Code learn (2021) donde refiere que “el pensamiento computacional es como un proceso mental que dirige a alguien a buscar las soluciones mejores, más rentables y abiertas” (p.44). Por otro lado, Vera et al. (2015) agregan que “para proponer la mejor solución, las personas pueden desarrollar sus habilidades de resolución de problemas mediante el uso del pensamiento computacional”.

Por otro lado, desde la perspectiva como recurso para el proceso de aprendizaje y enseñanza se respalda lo referido por Vera (2018) quien aseveró que las habilidades y cualidades personales permiten a un individuo abordar y solucionar un problema utilizando el pensamiento computacional como recurso alternativo. Además, según Lamoth et al. (2020) explican que con el pensamiento computacional se logra incrementar y estimular las iniciativas educativas destinadas para resolver problemas de la vida cotidiana.

## V. CONCLUSIONES

**PRIMERA:** De acuerdo al objetivo general se logró determinar la existencia de una relación positiva baja entre el dominio del software educativo Scratch y el nivel del pensamiento computacional de los Estudiantes de la IES José Gálvez de Yunguyo. Así mismo, de acuerdo a lo establecido en el valor de  $Rho = 0,326$  y P valor  $Sig. = (0,000) < 0,05$ ; corresponde rechazar la  $H_0$  y aceptar la ( $H_1$ ), demostrado de este modo que el Programa Scratch contribuye en el desarrollo del aprendizaje del pensamiento computacional. Es decir, si se mejora el dominio del Scratch, ésta se verá reflejado en el desempeño del aprendizaje del pensamiento computacional de los estudiantes.

**SEGUNDA:** Se logró determinar la existencia de una relación positiva baja entre el dominio del software educativo Scratch y el desarrollo de conceptos computacionales de los Estudiantes de la IES José Gálvez de Yunguyo. Esta afirmación se basa de acuerdo a lo establecido en la (tabla 12) donde se evidencia el valor de  $Rho = 0,210$  y P valor  $Sig. = (0,020) < 0,05$ . Así mismo, estadísticamente se identificó una relación directa proporcional donde predomina el dominio alto con el (55.6%) y en cuanto al desarrollo de conceptos computacionales predomina un desarrollo alto con el (66.9%).

**TERCERA:** Se logró determinar la existencia de una relación positiva baja entre el dominio del software educativo Scratch y el desarrollo de prácticas computacionales de los Estudiantes de la IES José Gálvez de Yunguyo. Esta afirmación se basa de acuerdo a lo establecido en la (tabla 13) donde se evidencia el valor de  $Rho = 0,321$  y P valor  $Sig. = (0,000) < 0,05$ . Así mismo, los resultados estadísticos indican que los estudiantes que poseen un nivel de dominio regular, presentan un desarrollo de aprendizaje en nivel medio con el (49.2%); Así mismo, los estudiantes que poseen un dominio alto, presentan un desarrollo alto de prácticas computacionales con el (50.0%).

**CUARTA:** Se logró determinar la existencia de una relación positiva muy baja entre el dominio del software educativo Scratch y el desarrollo de perspectivas computacionales de los Estudiantes de la IES José Gálvez de Yunguyo. Esta afirmación se basa de acuerdo a lo establecido en la (tabla 14) donde se evidencia el valor de  $Rho = 0,169$  y P valor  $Sig. = (0,061) > 0,05$ . Es decir, el dominio del Scratch no determina en su totalidad para el desarrollo del aprendizaje de las perspectivas computacionales.

## **VI. RECOMENDACIONES**

**PRIMERA:** Se sugiere a los docentes de la institución a incorporar el uso del Scratch desde los primeros ciclos de educación secundaria a través sesiones de aprendizaje con la finalidad de mejorar el dominio sobre su uso, características y funcionalidad que ofrece dicho programa educativo.

**SEGUNDA:** Se sugiere a los estudiantes aplicar con mayor frecuencia los conocimientos aprendidos mediante el uso del Scratch al momento de resolver algún tipo de problema relacionado a conceptos, lógicos, criterios, análisis y perspectivas. Puesto que se logró demostrar que a través del dominio del Scratch se puede mejorar las habilidades y competencias del desarrollo del aprendizaje.

**TERCERA:** Se sugiere a las autoridades educativas del ámbito local a gestionar capacitaciones para la plana docente sobre el uso adecuado del Scratch como herramienta y recurso alternativo para transmitir conocimientos en las diferentes áreas del sistema curricular.



## INFORME DE ORIGINALIDAD

13%

INDICE DE SIMILITUD

13%

FUENTES DE INTERNET

2%

PUBLICACIONES

2%

TRABAJOS DEL  
ESTUDIANTE

## FUENTES PRIMARIAS

1	<a href="http://repositorio.unap.edu.pe">repositorio.unap.edu.pe</a> Fuente de Internet	3%
2	<a href="http://repositorio.ucv.edu.pe">repositorio.ucv.edu.pe</a> Fuente de Internet	2%
3	<a href="http://repositorio.upt.edu.pe">repositorio.upt.edu.pe</a> Fuente de Internet	1%
4	<a href="http://tesis.ucsm.edu.pe">tesis.ucsm.edu.pe</a> Fuente de Internet	1%
5	<a href="http://repositorio.uct.edu.pe">repositorio.uct.edu.pe</a> Fuente de Internet	1%
6	<a href="http://es.slideshare.net">es.slideshare.net</a> Fuente de Internet	<1%
7	<a href="http://repositorio.une.edu.pe">repositorio.une.edu.pe</a> Fuente de Internet	<1%
8	<a href="http://repositorio.uigv.edu.pe">repositorio.uigv.edu.pe</a> Fuente de Internet	<1%
9	<a href="http://hdl.handle.net">hdl.handle.net</a> Fuente de Internet	<1%

10	<a href="http://revistas.uma.es">revistas.uma.es</a> Fuente de Internet	<1 %
11	<a href="http://repositorio.autonmadeica.edu.pe">repositorio.autonmadeica.edu.pe</a> Fuente de Internet	<1 %
12	<a href="http://www.researchgate.net">www.researchgate.net</a> Fuente de Internet	<1 %
13	<a href="http://anfei.mx">anfei.mx</a> Fuente de Internet	<1 %
14	<a href="http://revistas.uned.es">revistas.uned.es</a> Fuente de Internet	<1 %
15	<a href="http://docplayer.es">docplayer.es</a> Fuente de Internet	<1 %
16	<a href="http://repositorio.upd.edu.pe">repositorio.upd.edu.pe</a> Fuente de Internet	<1 %
17	<a href="http://www.sagarpa.gob.mx">www.sagarpa.gob.mx</a> Fuente de Internet	<1 %
18	<a href="http://www.intechopen.com">www.intechopen.com</a> Fuente de Internet	<1 %
19	Submitted to Liberty University Trabajo del estudiante	<1 %
20	<a href="http://www.theibfr.com">www.theibfr.com</a> Fuente de Internet	<1 %
21	<a href="http://conrado.ucf.edu.cu">conrado.ucf.edu.cu</a> Fuente de Internet	<1 %

22	<a href="http://issuu.com">issuu.com</a> Fuente de Internet	<1 %
23	<a href="http://www.iccat.int">www.iccat.int</a> Fuente de Internet	<1 %
24	<a href="http://www.clubensayos.com">www.clubensayos.com</a> Fuente de Internet	<1 %
25	<a href="http://produccioncientificaluz.org">produccioncientificaluz.org</a> Fuente de Internet	<1 %
26	<a href="http://repositorio.upla.edu.pe">repositorio.upla.edu.pe</a> Fuente de Internet	<1 %
27	<a href="http://www.sgsgroup.com.ar">www.sgsgroup.com.ar</a> Fuente de Internet	<1 %
28	(7-6-03) <a href="http://130.206.130.211/~admin/aenui/ProcWeb/resume">http://130.206.130.211/~admin/aenui/ProcWeb/resume</a> Fuente de Internet	<1 %
29	<a href="http://1library.co">1library.co</a> Fuente de Internet	<1 %
30	Submitted to Universidad Maria Auxiliadora SAC Trabajo del estudiante	<1 %
31	<a href="http://pesquisa.bvsalud.org">pesquisa.bvsalud.org</a> Fuente de Internet	<1 %
32	<a href="http://repositorio.uac.edu.co">repositorio.uac.edu.co</a> Fuente de Internet	<1 %

33	<a href="http://repositorio.udh.edu.pe">repositorio.udh.edu.pe</a> Fuente de Internet	<1 %
34	<a href="http://revistas.um.es">revistas.um.es</a> Fuente de Internet	<1 %
35	<a href="http://sanosila.blogspot.com">sanosila.blogspot.com</a> Fuente de Internet	<1 %
36	<a href="http://catalogo.ucateci.edu.do">catalogo.ucateci.edu.do</a> Fuente de Internet	<1 %
37	<a href="http://es.unionpedia.org">es.unionpedia.org</a> Fuente de Internet	<1 %
38	<a href="http://lareferencia.info">lareferencia.info</a> Fuente de Internet	<1 %
39	<a href="http://renati.sunedu.gob.pe">renati.sunedu.gob.pe</a> Fuente de Internet	<1 %
40	<a href="http://repositorio.cientifica.edu.pe">repositorio.cientifica.edu.pe</a> Fuente de Internet	<1 %
41	<a href="http://www.dspace.uce.edu.ec">www.dspace.uce.edu.ec</a> Fuente de Internet	<1 %
42	<a href="http://www.infoagenda.net">www.infoagenda.net</a> Fuente de Internet	<1 %
43	<a href="http://www.lexureditorial.com">www.lexureditorial.com</a> Fuente de Internet	<1 %
44	<a href="http://www.mysciencework.com">www.mysciencework.com</a> Fuente de Internet	<1 %

Excluir citas

Activo

Excluir coincidencias < 9 words

Excluir bibliografía

Activo