



UNIVERSIDAD
BOLIVARIANA
DEL ECUADOR

TRABAJO DE TITULACIÓN

UNIVERSIDAD
BOLIVARIANA
DEL ECUADOR



UNIVERSIDAD BOLIVARIANA DE ECUADOR

MAESTRÍA EN EDUCACIÓN MENCIÓN EN PEDAGOGÍA ENTORNO DIGITALES

TRABAJO DE TITULACIÓN

PREVIO A LA OBTENCIÓN DEL TÍTULO DE

MAESTRÍA EN EDUCACIÓN MENCIÓN EN PEDAGOGÍA ENTORNO DIGITALES

Tema:

ESTRATEGIAS DIDÁCTICAS BASADAS EN SCRATCH PARA LA ENSEÑANZA DE
PROGRAMACIÓN EN ESTUDIANTES DE OCTAVO GRADO

Autores:

RONY DERECK CERVANTES SANCHEZ
JUAN CARLOS SALAZAR TORRES

Tutor/a:

MSC. ENRIQUE GUSTAVO GUEVARA CAIZAPANTA

ECUADOR

2026



La Universidad para todos





DEDICATORIA

Dedico este proyecto a mi abuela que me ha acompañado a lo largo de este proceso y a mi estudiante Kristel Doménica Travéz que en paz descansa.

RONY DERECK CERVANTES SÁNCHEZ

Dedico este proyecto a todos los docentes que día a día se esfuerza por inspirar y guiar a sus alumnos, sembrando en ellos las semillas del conocimiento, la curiosidad y la creatividad. Su dedicación y compromiso son fundamentales para transformar la educación y preparar nuevas generaciones para los desafíos del futuro.

JUAN CARLOS SALAZAR TORRES





AGRADECIMIENTO

Agradezco a los docentes que impartieron cada uno de los módulos de esta maestría ya que han dejado una huella en mí como persona y como profesional, también agradezco a mi compañero con el cual trabajamos juntos en varios módulos hasta el final.

RONY DERECK CERVANTES SÁNCHEZ

Quiero expresar mi más sincero agradecimiento a mi colega docente, cuya colaboración y apoyo constante han sido fundamentales para el desarrollo y la implementación de este trabajo educativo. Agradezco también a los estudiantes, a mi familia y amigos por su apoyo incondicional y comprensión durante el proceso de este proyecto.

JUAN CARLOS SALAZAR TORRES



RESUMEN

La presente investigación tuvo como propósito desarrollar y evaluar una estrategia didáctica basada en el uso de Scratch para la enseñanza de programación en estudiantes de octavo grado de la Escuela Particular de Educación Básica “Pablo Palacio”. La investigación se desarrolló bajo un enfoque mixto, combinando métodos cuantitativos y cualitativos, lo que permitió analizar de manera integral la incidencia de la propuesta en el aprendizaje y la motivación de los estudiantes. Desde el punto de vista metodológico, el estudio se sustentó en el método explicativo, orientado a identificar los efectos de la estrategia didáctica implementada en el contexto educativo.

La población estuvo conformada por 30 estudiantes de octavo grado, quienes participaron en una secuencia didáctica estructurada en cuatro sesiones, apoyadas en cápsulas educativas y actividades prácticas en el entorno de programación Scratch. En el componente cuantitativo, se aplicaron cuestionarios tipo Likert antes y después de la implementación, con el fin de medir los niveles de motivación y percepción del aprendizaje. En el componente cualitativo, se emplearon técnicas como la observación directa y entrevistas, que permitieron recoger información sobre la participación, el interés y las dinámicas de trabajo durante el desarrollo de las sesiones.

Asimismo, se utilizó una rúbrica de evaluación para valorar los proyectos finales elaborados por los estudiantes, centrados en la creación de un videojuego como evidencia de aprendizaje. Los resultados evidenciaron que la estrategia didáctica basada en Scratch contribuyó de manera significativa al fortalecimiento del interés por la programación y al desarrollo de habilidades básicas de pensamiento computacional. En conclusión, la propuesta se constituye en una alternativa pedagógica viable e innovadora para la enseñanza de programación en educación básica.

Palabras clave: Programación, Scratch, Enseñanza, Estrategia Didáctica, Metodología.



ABSTRACT

This research focuses on the development and evaluation of Scratch-based teaching strategies for programming instruction, aimed at eighth-grade students at the “Pablo Palacio” Private Basic Education School. The study employs a mixed-methods approach, combining quantitative and qualitative methods, which allows for a comprehensive understanding of the impact of these strategies on student learning and motivation.

Learning to program has become a crucial topic in today's educational context, given the advancement of technology and the need to prepare new generations for an increasingly complex digital environment. Scratch, as an educational tool, offers an accessible and engaging approach for students, allowing them to learn programming concepts through the creation of interactive projects. This aligns with current trends in education, which promote the use of technology to motivate students and make learning more dynamic.

The study adopts a constructivist approach, based on the idea that students construct their own knowledge through interaction with their environment and with others. This is reflected in the use of teaching strategies that foster active learning and collaboration. Students will work on group projects, which will not only allow them to apply programming concepts but will also promote social and teamwork skills. This methodology is key to the students' holistic development, as it aims for them to acquire not only technical knowledge but also interpersonal skills.

The research methodology incorporates various data collection techniques. Surveys and interviews will be conducted with students and teachers to gather information about their perceptions of motivation and interest in programming. In addition, classroom observations will be carried out to evaluate the implementation of the teaching strategies and their effectiveness. The use of rubrics will allow for measuring the quality of the projects developed by the students, as well as their understanding of the programming concepts taught.

Keywords: Programming, Scratch, Teaching, Didactic Strategy, Methodology.



ÍNDICE GENERAL

FICHA SENESCYT PARA EL REPOSITORIO	i
COPIA INFORME DE SIMILITUD (ANTIPLAGIO)	iii
CERTIFICACIÓN DE AUTORÍA Y CESIÓN DE DERECHOS DEL AUTOR (ES)	iv
AVAL DEL TUTOR DE LA TESIS	v
DEDICATORIA.....	vi
AGRADECIMIENTO.....	vii
RESUMEN.....	viii
ABSTRACT	ix
Índice de Anexos	xvi
INTRODUCCIÓN	1
Presentación y contextualización	1
Justificación del Problema.....	2
Planteamiento del Problema.....	2
Precisión del tema	3
Objeto de la Investigación.....	3
Objetivo General	3
Preguntas científicas.....	4
Categorías de la Investigación.....	4
Proceso de enseñanza-aprendizaje de la asignatura informática.....	4
Motivación.....	5
Estudio del Lenguaje de programación Scratch	6
Variables de la investigación.....	7
Objetivos específicos de la investigación.....	7
Metodología a Emplear	7
Población y Muestra.....	7
Tipo de Investigación	8
Importancia de la Investigación	8
Coherencia entre los elementos del diseño teórico – metodológico.....	9
Fundamentación Teórica	9
Metodología	10
Instrumentos de Recolección.....	10
Implementación de la Propuesta.....	10
Evaluación y retroalimentación.....	10
Descripción breve del contenido de los capítulos	10
Capítulo I: Marco Teórico	10





Capítulo II: Metodología para el Desarrollo de la Investigación y Estudio Diagnóstico	11
Capítulo III: Presentación y Validación de la Propuesta	11
CAPÍTULO I: MARCO TEÓRICO.....	12
1.1 Antecedentes	12
1.2 Fundamentos Teóricos	13
1.2.1 Fundamentos de la Enseñanza de la Programación	13
1.2.2 Motivación y Aprendizaje	15
1.2.3 Scratch como Herramienta Didáctica	16
1.2.4 Aplicación de Estrategias de Enseñanza en Entornos Digitales	18
1.2.5 Aprendizaje Basado en Proyectos	21
1.2.6 Scaffolding progresivo y rúbricas con Scratch.....	23
1.2.7 Trabajo Colaborativo y Roles en Scratch	24
1.3 Bases Legales	25
1.3.1 La Declaración Universal de los Derechos Humanos.	25
1.3.2 Constitución de la República Del Ecuador	26
CAPÍTULO II: METODOLOGÍA PARA EL DESARROLLO DE LA INVESTIGACIÓN Y ESTUDIO DIAGNÓSTICO	28
2.1 Conceptualización y operacionalización de las categorías y variables	28
2.2 El enfoque de la investigación.....	20
2.3 Justificación del enfoque mixto.....	21
2.4 Alcance de la investigación.....	21
2.5 Declaración y justificación del tipo de investigación.....	22
2.6 Métodos empleados y sus propósitos en el contexto de la investigación.....	22
2.7 Instrumentos de la Investigación.....	23
2.8 Delimitación de la Población Y Muestra.....	24
2.8.1 Delimitación de la Población.....	24
2.9 Descripción de las Etapas de la Investigación.....	25
2.9.1 Etapa de estudio teórico.....	25
2.9.2 Etapa de diagnóstico inicial	26
2.9.3 Etapa de la modelación de la propuesta.....	28
2.9.4 Fase del diseño.....	28
2.9.5 Etapa del diagnóstico final.....	34





2.10 Presentación de los resultados del estudio diagnóstico	34
2.10.1 Resultados.....	43
CAPÍTULO III: PRESENTACIÓN Y VALIDACIÓN DE LA PROPUESTA	46
3.1 Propuesta	46
3.2 Fundamentación	47
3.3 Objetivos Generales y Específicos	47
3.3.1 Objetivos Generales.....	47
3.3.2 Objetivos Específicos	47
3.4 Característica de la Propuesta.....	48
3.5 Estructura y Dinámica de sus Componentes	48
3.6.1 Exigencias.....	49
3.6.2 Requisitos	50
3.6.3 Condiciones	51
3.7 Demostraciones	53
3.8 Formas de aplicación, implementación y evaluación, recursos y beneficiarios	54
3.8.1 Formas de aplicación.....	54
3.8.2 Implementación y evaluación.....	56
3.8.3 Recursos.....	56
3.8.4 Beneficiarios.....	57
58	
3.9 Validación de la Propuesta	58
3.9.1 Análisis de los profesionales	59
3.10 Descripción del Proceso	63
3.11 Instrumentos de Validación.....	65
3.12 Resultados de la Validación	65
3.12.1 Encuestas pre y post proyecto.....	66
3.12.2 Evaluación oral.....	66
3.12.3 Rubrica de evaluación	66
3.12.4 Observaciones directas	67
3.12.5 Presentaciones finales.....	67
3.12.6 Entrevista.....	67



3.12.7 Análisis de proyecto	67
3.13 CONCLUSIONES.....	68
3.14 RECOMENDACIONES	68
BIBLIOGRAFIA.....	70
ANEXOS.....	74

Índice de Figuras

Figura 1 Te sientes motivado por aprender programación.....	35
Figura 2¿Crees que tus profesores tienen claro el contenido a enseñar?	35
Figura 3¿Te parece que las actividades de programación deben ser solamente de manera escrita?.....	36
Figura 4¿Sientes que tus profesores te apoyan cuando tienes dificultades al programar?.....	36
Figura 5¿Crees que la manera en la cual estás aprendiendo te será útil en el futuro?	36
Figura 6¿Puedes elegir cómo quieres trabajar tus proyectos de programación?.....	37
Figura 7¿Tienes oportunidad de elegir temas o proyectos que te interesen?	37
Figura 8¿Puedes trabajar a tu propio ritmo en tus proyectos de programación?	37



Figura 9¿ Tienes la libertad de probar diferentes enfoques en tus proyectos de programación?	38
Figura 10¿ Consideras que tus profesores están dispuestos a adaptar sus planes de enseñanzas a tus necesidades?	38
Figura 11¿ Crees que los proyectos de programación están orientados con tus intereses y hobbies?	39
Figura 12¿ Te parece que los ejemplos utilizados en clases son relevantes en la vida diaria?	39
Figura 13¿ Crees que la programación te ayudará a resolver problemas o situaciones que enfrentas a diario?	40
Figura 14¿ Crees que la programación te ayudará a resolver problemas o situaciones que enfrentas a diario?	40
Figura 15¿ Crees que las evaluaciones y retroalimentaciones que recibes en clases de programación te ayuda a mejorar tus habilidades?	40
Figura 16¿ Te parece que las evaluaciones de tus proyectos son justas y objetivas?	41
Figura 17¿ Crees que las retroalimentaciones que recibes de tus profesores te ayudan a entender que debes mejorar en tus proyectos de programación?	41
Figura 18¿ Te sientes motivado para mejorar tus habilidades en programación debido a las evaluaciones y retroalimentaciones que recibes?	42
Figura 19¿ Crees que las evaluaciones de tus proyectos de programación se reflejan con precisión tus habilidades y conocimientos en programación?	42
Figura 20 Los objetivos de aprendizaje de las cápsulas educativas están claramente definidos.	59
Figura 21 El contenido de las cápsulas es relevante para los estudiantes de octavo grado que aprenden programación	59



Figura 22 La estructura de cada cápsula facilita el aprendizaje de los conceptos de programación.....	60
Figura 23 Los recursos multimedia incluidos en las cápsulas son efectivos para captar la atención de los estudiantes.	60
Figura 24 Las actividades interactivas propuestas fomentan la participación activa de los estudiantes.	61
Figura 25 Las herramientas de evaluación propuestas son adecuadas para medir el aprendizaje de los estudiantes.....	61
Figura 26 La propuesta permite la personalización del aprendizaje según las necesidades de los estudiantes.	62
Figura 27 La propuesta es adaptable a diferentes áreas educativas.	62
Figura 28 Creo que esta propuesta tiene un potencial significativo para mejorar la enseñanza de programación.....	63

Índice de tablas

Tabla 1 Estrategias Didácticas	20
Tabla 2 Estrategias didácticas 2	21
<i>Tabla 3 Operacionalización.....</i>	20
Tabla 4 Rúbrica	33
Tabla 5Exigencias	49
Tabla 6 Requisitos.....	50
Tabla 7 Condiciones a cumplir.....	51
Tabla 8 Descripción del video 1.....	53



Tabla 9 Descripción del video 2.....	53
Tabla 10 Descripción del video 3.....	53
Tabla 11 Descripción del video 4.....	54
Tabla 12 Formas De Aplicación, Implementación y Evaluación.....	55
Tabla 13 Recursos	57
Tabla 14 Beneficios.....	58
Tabla 15 Descripción del proceso	64
Tabla 16 Instrumentos de validación	65

Índice de Anexos

Anexo 1 Presentación de la propuesta a la rectora Joselyn Villanueva	74
Anexo 2 Introducción a Scratch	74
Anexo 3 Primera Capsula Educativa.....	74
Anexo 4 Presentación e la segunda cápsula educativa.....	75
Anexo 5 Presentación de la tercera cápsula educativa.....	75
Anexo 6 Presentación de la cuarta cápsula educativa	75
Anexo 7 Clausura de la propuesta.....	75
Anexo 8 Análisis por parte de los profesionales 1	76
Anexo 9 Análisis por parte de los profesionales 1	76



Anexo 10 Análisis por parte de los profesionales 3 77

Anexo 11 Análisis por parte de los profesionales 4 77





INTRODUCCIÓN

Presentación y contextualización

La enseñanza de la programación se ha vuelto esencial en el contexto educativo actual, dada la creciente digitalización de la sociedad. En este sentido, las estrategias didácticas basadas en Scratch se proponen como una innovadora alternativa para captar el interés de los estudiantes de octavo grado, facilitando su comprensión de conceptos fundamentales de programación.

Scratch, desarrollado por el MIT, es un lenguaje de programación visual que permite a los estudiantes crear proyectos interactivos mediante bloques gráficos. Esta herramienta reduce las barreras técnicas de los lenguajes de programación tradicionales, haciéndola accesible y atractiva para estudiantes en etapas iniciales de aprendizaje. Al utilizar Scratch, se promueve el aprendizaje activo, donde los estudiantes pueden experimentar, crear y resolver problemas de forma creativa.

Las estrategias didácticas que se implementarán incluyen el aprendizaje basado en proyectos, donde los estudiantes trabajan en equipos para desarrollar juegos y animaciones, fomentando la colaboración y habilidades interpersonales. Asimismo, se incorporarán cápsulas educativas que presentan de manera concisa conceptos clave, facilitando una comprensión rápida y efectiva. Según Montoya & López, (2014) considera que Scratch es un entorno de programación visual que permite a los usuarios crear proyectos multimedia interactivos. Un gran número de personas ha creado una amplia variedad de proyectos, utilizando Scratch, como videos musicales, presentaciones, juegos de ordenador y otro tipo de animaciones (p. 3).

La contextualización de estas estrategias es fundamental, ya que los estudiantes de octavo grado se encuentran en una etapa crucial de desarrollo cognitivo y social. Al integrar el uso de tecnología en el aula, se busca no solo mejorar la comprensión de la programación, sino también aumentar la motivación y el compromiso de los estudiantes con su aprendizaje.

Las estrategias didácticas basadas en Scratch no solo abordan la enseñanza de la programación de manera efectiva, sino que también contribuyen al desarrollo integral de los estudiantes, preparándolos para enfrentar los desafíos de un mundo digital en constante evolución.



Justificación del Problema

La tecnología avanza muy rápido, por lo que es importante tener conocimientos básicos, para que en un futuro poder tener los conocimientos necesarios para desempeñarse en el mundo, esta herramienta ayuda a que puedan comprender de forma fácil y sin tantas complicaciones de cómo se puede desarrollar un programa, aprovechando la primicia de que son jóvenes y a la vez motivándolos a seguir aprendiendo sobre la programación.

Un punto importante a considerar es que no solo desarrollarían habilidades digitales, sino también un pensamiento crítico y de resolución sistemática de problemas, al implementar esa motivación de aprender, en esta ocasión, en la introducción de los conceptos de programación de forma sencilla y comprensible a través de Scratch como parte de su formación tecnológica temprana, destaca la relevancia en la educación actual.

Según Kim & Dinç, (2025) las habilidades de resolución de problemas, razonamiento y habilidades digitales son fundamentales para el mundo actual, el cual está impulsado por la tecnología; en su estudio, el cual se basó en la enseñanza por bloques, implementado en la herramienta Scratch, y además que se enfocó en diferentes materias escolares, en este reconoce que adaptar las actividades entorno a dicha herramienta refleja interés en los estudiantes, siempre que se realice una retroalimentación frecuente, para que el estudiante comprenda el tema dado y aprenda sobre el mismo, ya que permite también al docente adaptarse según las necesidades y capacidades de los estudiantes.

Planteamiento del Problema

En la Escuela Particular de Educación Básica “Pablo Palacio”, ubicada en el cantón Guayaquil, provincia del Guayas, específicamente en el segundo callejón 25 entre la P y N, en el suburbio suroeste de la ciudad, se imparte la asignatura de Computación a los estudiantes de octavo grado en un contexto social caracterizado por situaciones de vulnerabilidad y conflictividad asociadas a la criminalidad. Este entorno influye de manera directa en los procesos educativos, generando desafíos adicionales para mantener la motivación, la concentración y el interés de los estudiantes durante el desarrollo de las clases.

En el área de Computación, se ha evidenciado que los estudiantes de octavo grado presentan bajos niveles de motivación y dificultades en la comprensión de conceptos básicos de programación, lo que se refleja en una participación limitada y en la escasa aplicación práctica de los contenidos abordados en clase. Estas dificultades se relacionan con el uso predominante de metodologías tradicionales, centradas en la explicación teórica, con poco énfasis en



actividades interactivas y creativas, a pesar de que la institución cuenta con acceso a internet y un laboratorio de computación. Como resultado, los estudiantes perciben la programación como un contenido complejo y poco significativo para su realidad educativa.

De no intervenir en esta problemática, existe el riesgo de que los estudiantes continúen mostrando desinterés por la asignatura de Computación, limitando el desarrollo de habilidades de pensamiento computacional, resolución de problemas y competencias digitales fundamentales para su formación académica y futura inserción en entornos tecnológicos.

Ante esta realidad, se evidencia la necesidad de analizar estrategias didácticas innovadoras que contribuyan a mejorar la motivación y el aprendizaje de la programación en los estudiantes de octavo grado, considerando las características del contexto educativo y social en el que se desarrolla la investigación.

Precisión del tema

Implementación de estrategias didácticas basadas en Scratch para fortalecer la motivación y el aprendizaje de la programación en estudiantes de octavo grado.

Objeto de la Investigación

La enseñanza de programación en estudiantes de octavo año con el uso de Scratch servirá como estrategia didáctica, buscando motivar a los estudiantes en el proceso educativo, haciendo que sea una experiencia más adaptable y accesible, además de facilitar el aprendizaje de conceptos complejos de la programación.

El aprendizaje de los estudiantes, se lo realizará con estrategias didácticas, como:

- Aprendizaje basado en proyectos: Aplicación de casos reales en los que permitan aplicar los conceptos básicos de la programación.
- Trabajo colaborativo: Interacción entre los estudiantes, trabajo en equipo y retroalimentación constante.
- Capsulas educativas: Sesiones dinámicas, guía breve y concisa de los conceptos básicos de la programación.

La investigación medirá el impacto de las estrategias antes mencionadas en los estudiantes de octavo año hacia la programación y a su vez, su rendimiento académico.

Objetivo General

Analizar el impacto de la implementación de una estrategia didáctica basada en Scratch, mediante cápsulas educativas, en la motivación y en la comprensión de los conceptos de



programación de los estudiantes de octavo grado de la asignatura de Computación de la Escuela Particular de Educación Básica “Pablo Palacio”, ubicada en el cantón Guayaquil, provincia del Guayas.

Preguntas científicas

- ¿Cómo contribuye la estrategia didáctica basada en cápsulas educativas en Scratch al desarrollo de la motivación en los estudiantes de octavo grado?
- ¿Qué efectos tiene la implementación de la estrategia didáctica basada en cápsulas educativas en Scratch sobre el aprendizaje de conceptos básicos de programación?
- ¿En qué medida dicha estrategia influye en la participación y el compromiso de los estudiantes durante las clases de programación?

Categorías de la Investigación

Proceso de enseñanza-aprendizaje de la asignatura informática

El proceso de enseñanza aprendizaje es un proceso complejo, que supone un importante reto para el profesor debido a la presencia de mecanismos convergentes y divergentes utilizados por los diferentes actores del proceso en función de la transmisión de contenidos de enseñanza por una parte y la asimilación de ese contenido por la otra (Navarro & Samón, 2017)

La interacción y la complejidad de estos mecanismos crea la necesidad de armonizar la planificación docente, en especial teniendo en cuenta las particularidades cognitivas, emocionales y sociales de los estudiantes; el docente, al cumplir un rol activo debe ser también flexible y ser capaz de responder a la diversidad de su alumnado, acogiendo a estrategias diferenciadas y a metodologías innovadoras que favorezcan a la adquisición de conocimientos de sus estudiantes, así como también el desarrollo de habilidades críticas, creativas y de razonamiento lógico.

El proceso de enseñanza aprendizaje, como estrategia que propicia espacios con ambientes adecuados al medio educativo, para estimular la investigación, además indicaron que su utilización promueve entre los estudiantes, sentido de pertenencia al aprender las componentes esenciales guiadas por su docente y que este haya realizado una experiencia única de aprendizaje, (Kim & Dinç, 2025)

Los estudiantes, al tener ese sentido de pertenencia fortalece el aprendizaje, ya que responde a sus necesidades reales, permitiéndoles explorar y experimentar en forma activa, haciendo que tenga más opciones de aprendizaje, ayudándolos así en su propio desarrollo académico. Un



docente que diseña experiencias únicas no solo transmite conocimiento teórico, sino que también impulsa a la reflexión de los hechos, al análisis y aplicación de los temas vistos, conectándolos con los problemas de la vida cotidiana.

“Un proceso de enseñanza -aprendizaje se conceptualiza como un ciclo social que se desarrolla en un espacio específico a través de la interacción, donde se evidencia la didáctica como punto de partida para efectuar lo antes mencionado” (Ordoñez y otros, 2020). Los avances tecnológicos y las transformaciones culturales no son estáticos, fluctúan en cambios en el entorno educativo, el eje principal es la estrategia didáctica, que permite seleccionar el método, la técnica y los recursos adecuados para orientar el aprendizaje de los estudiantes.

El éxito depende de la capacidad del docente de integrar la teoría, la práctica y la adaptación constante de un grupo de estudiantes, fomentando la colaboración y promoviendo el respeto mutuo, recordando que todos aprenden a diferentes ritmos y que se deben de ayudar mutuamente; considerando el proceso de enseñanza – aprendizaje como un sistema constructivo implica también reconocer que su mejoría implica la reflexión, actualización constante de conocimientos tanto en la materia dada como en la investigación pedagógica.

Motivación

La baja motivación puede estar relacionada con distintos factores como el uso de métodos convencionales para enseñar a programar generando poco interés por parte de los estudiantes. La falta de materiales de aprendizaje especialmente en el idioma nativo, el poco tiempo que se otorga a la interacción entre estudiante-docente y la complejidad del curso de programación también influyen en la desmotivación de los alumnos (Cuervo, 2021). En este conjunto de factores crea un escenario donde el estudiante percibe la programación como una actividad de gran dificultad, poco atractiva e irrelevante para sus intereses de ese momento.

La motivación, suele ser incluido como elemento de diseño en herramientas de aprendizaje bajo la forma de contenidos atractivos, interactivos y lúdicos. Sin embargo, es poco frecuente encontrar estudios cuantitativos que registren la motivación (intrínseca o extrínseca) del estudiante de programación hacia una herramienta específica en la forma de intención de uso (Argelio y otros, 2013).

El desafío que existe para el docente aún con los recursos tecnológicos que existe actualmente, es que no siempre se alinean con el contexto sociocultural, lo cual hace que sea poco atractivo para los estudiantes y no le tomen como motivación para aprender de la programación.

La motivación se posiciona como un factor determinante en el proceso de aprendizaje y el





rendimiento académico, impulsando a los docentes a explorar estrategias, métodos y herramientas que eviten que la experiencia de aprendizaje se vuelva monótona y desinteresante (Salgado, 2024) En este sentido, es fundamental que el proceso del estudiante en su aprendizaje de la programación este basada en metodologías activas, que promuevan la interacción reciproca entre docente y el alumnado, experimentando el trabajo colaborativo, permitiendo construir el conocimiento de forma significativa.

Estudio del Lenguaje de programación Scratch

Scratch es un entorno de programación visual que permite a los usuarios crear proyectos multimedia interactivos. Un gran número de personas ha creado una amplia variedad de proyectos, utilizando Scratch, como videos musicales, presentaciones, juegos de ordenador y otro tipo de animaciones. (López & Sánchez, 2015)

El objetivo original de Scratch era desarrollar un enfoque de programación que atrajera a las personas, sin importar la edad, origen social, o educacional, al desarrollo de soluciones algorítmicas sin las complejidades de sintaxis y semántica de los lenguajes de programación tradicional, esto es, hacer de Scratch un lenguaje para programar historias interactivas, juegos, animaciones y simulaciones fácil para todos sus usuarios, quienes pueden además compartir sus creaciones con otros.

De esta forma, el objetivo principal de Scratch no es preparar a las personas para carreras profesionales o técnicas del área de programación, sino para nutrir una nueva generación de pensadores creativos y sistemáticos utilizando la programación para expresar sus ideas (Vidal y otros, 2019). Esta herramienta de programación visual, permite que los usuarios construyan historias interactivas, animaciones, juegos y simulaciones sin necesidad de aprender un lenguaje de programación complejo, lo que la convierte en una plataforma accesible para niños, jóvenes y adultos.

Scratch apoya el desarrollo de habilidades de aprendizaje para el siglo XXI, ya que desarrolla habilidades de procesamiento de información, comunicación, pensamiento creativo y solución de problemas, mediante la creación y edición de diversos medios digitales. (García A. , 2022) Además, promueve el trabajo colaborativo, ya que su comunidad en línea ofrece un espacio para compartir proyectos, recibir retroalimentación y aprender de las creaciones de otros usuarios. Este enfoque colaborativo estimula la motivación intrínseca y fortalece la autoconfianza en el aprendizaje.

El uso de Scratch contribuye al desarrollo del pensamiento computacional, entendiendo este





como la capacidad de formular problemas y soluciones de manera que puedan ser ejecutadas por un ordenador, lo que no solo beneficia el ámbito tecnológico, sino que también se transfiere a otras áreas del conocimiento. En entornos educativos, su aplicación no se limita únicamente a la informática, sino que se integra en asignaturas como matemáticas, ciencias, arte o literatura, permitiendo experiencias interdisciplinarias que enriquecen el proceso formativo.

Variables de la investigación

Las variables de la investigación son las siguientes:

- **Variable independiente:** Estrategias didácticas
- **Variable dependiente:** Aprendizaje de la programación.

Objetivos específicos de la investigación

- Examinar los fundamentos teóricos sobre la motivación académica y su influencia en el proceso de enseñanza-aprendizaje.
- Sustentar teórica y metodológicamente el diseño de la investigación de enfoque mixto mediante la definición y operacionalización de instrumentos cuantitativos y cualitativos.
- Diseñar una estrategia didáctica basada en cápsulas educativas para potenciar la motivación y el aprendizaje de la programación.
- Validar la propuesta de estrategia didáctica basada en cápsulas educativas para la enseñanza de la programación mediante instrumentos cuantitativos y cualitativos.

Metodología a Emplear

El método a utilizar será el análisis y la síntesis ya que se relacionan con el pensamiento crítico, la capacidad resolutoria de problemas, y entre otras capacidades; el método inductivo y deductivo servirá para obtener conclusiones generales y específicas; el pre test y el post test será necesario para visualizar si el programa de Scratch ayuda con la enseñanza – aprendizaje de las definiciones básicas de la programación a los estudiantes de octavo año. La metodología será empírica, ya que se usará la observación, la entrevista a docentes y pruebas pedagógicas a los estudiantes, para determinar su rendimiento académico antes, durante y después de la introducción a la programación.

Población y Muestra

La población del estudio estará compuesta por los 30 estudiantes del octavo año de la Escuela Particular de Educación Básica “Pablo Palacio”, ubicado en la provincia del Guaya cantón Guayaquil, en las calles 26 ava y callejon N. En este caso, dado que el número total de



estudiantes de octavo año es 30, la muestra también será de 30 estudiantes, ya que toda la población de octavo será incluida en la investigación.

En este contexto, la población y la muestra son equivalentes, dado que todos los estudiantes de octavo año forman parte de la investigación.

Tipo de Investigación

Se desarrollará bajo un enfoque mixto con un alcance transversal, lo que permitirá un desempeño óptimo para el proceso de enseñanza – aprendizaje a los estudiantes de octavo año de la Escuela Particular de Educación Básica “Pablo Palacio”, esperando el favorecimiento del rendimiento académico y que desarrollen habilidades y destrezas en el área de informática, permitiendo establecer una dinámica con la herramienta Scratch y los temas sobre la introducción a la programación.

Importancia de la Investigación

En la actualidad, la enseñanza de la programación ha adquirido un papel protagónico dentro de la educación básica, debido a que el dominio de habilidades digitales y computacionales se considera fundamental para desenvolverse en un mundo cada vez más interconectado y tecnológicamente avanzado. La programación no solo fomenta competencias técnicas, sino que también promueve el desarrollo del pensamiento lógico, la creatividad, la resolución de problemas y la capacidad de trabajo colaborativo (Rodríguez, 2022). En este contexto, el uso de estrategias didácticas innovadoras que se adapten a las características cognitivas y motivacionales de los estudiantes de octavo grado se vuelve una necesidad pedagógica.

Scratch, como entorno de programación visual, ha demostrado ser una herramienta eficaz para introducir a los estudiantes en los conceptos básicos de la programación de manera sencilla, intuitiva y lúdica. Su interfaz basada en bloques elimina la barrera de la sintaxis compleja presente en otros lenguajes, lo que permite que los estudiantes se concentren en la lógica y la secuencia de instrucciones. Esto es especialmente importante en la educación básica, ya que facilita la adquisición de conocimientos a través de la experimentación y el aprendizaje activo. El diseño de estrategias didácticas que integren Scratch, como el uso de cápsulas educativas breves, posibilita la creación de experiencias de aprendizaje dinámicas y motivadoras (Kim & Dinç, 2025). Estas cápsulas, con una duración de entre tres y cinco minutos, permiten presentar conceptos de programación de forma segmentada, clara y práctica, lo que contribuye a mantener la atención de los estudiantes y a reforzar el aprendizaje progresivo. Además, el formato breve



favorece la retroalimentación inmediata, un factor clave para afianzar el conocimiento y corregir errores oportunamente.

La importancia de implementar este tipo de estrategias radica también en su capacidad para generar un cambio positivo en la percepción de los estudiantes hacia la programación. Diversos estudios han evidenciado que cuando los contenidos se presentan de forma interactiva y contextualizada, se incrementa la motivación intrínseca y la disposición para aprender. En este sentido, la motivación se convierte en un motor que impulsa el rendimiento académico y facilita el desarrollo de habilidades cognitivas superiores como el razonamiento lógico y la creatividad. Por otro lado, estas estrategias no solo contribuyen al aprendizaje técnico, sino que también fortalecen competencias transversales necesarias en el siglo XXI, tales como la comunicación, la colaboración y la adaptabilidad. En un entorno educativo como el de octavo grado, donde los estudiantes comienzan a definir sus intereses vocacionales, la enseñanza de programación mediante metodologías activas puede despertar el interés por carreras relacionadas con la tecnología y las ciencias computacionales.

La implementación de estrategias didácticas basadas en Scratch representa una oportunidad valiosa para mejorar el proceso de enseñanza-aprendizaje de la programación. No solo facilita la comprensión de conceptos complejos, sino que también motiva a los estudiantes y fomenta el desarrollo de habilidades esenciales para su formación académica y futura inserción en un mundo laboral digitalizado. De esta forma, el tema adquiere relevancia no solo por su aporte a la educación tecnológica, sino también por su contribución al desarrollo integral de los estudiantes.

Coherencia entre los elementos del diseño teórico – metodológico.

Se conforma por una estructura coherente que integra teoría, práctica y evaluación.

Fundamentación Teórica

La base teórica del estudio se centra en enfoques pedagógicos contemporáneos que respaldan la enseñanza de la programación, como el aprendizaje activo y colaborativo. La elección de Scratch se justifica a través de su capacidad para hacer accesible y entretener el aprendizaje de conceptos complejos. El marco teórico proporciona la justificación necesaria para la implementación de estrategias didácticas que fomentan la motivación y el desarrollo de habilidades técnicas.



Metodología

La metodología adoptada, que combina métodos cualitativos y cuantitativos, asegura que los datos obtenidos proporcionen una visión integral de la efectividad de las estrategias didácticas. El enfoque mixto permite una evaluación más completa, incorporando tanto la perspectiva de los estudiantes a través de encuestas y entrevistas, como la observación directa del desempeño en el aula. Esta metodología se alinea con los principios del aprendizaje activo, promoviendo la participación activa y la reflexión.

Instrumentos de Recolección

Los instrumentos de recolección de datos, como encuestas y rúbricas de evaluación, están diseñados para medir la motivación, la comprensión conceptual y el desempeño en proyectos en Scratch. Estos instrumentos son coherentes con los objetivos de investigación y permiten recopilar información relevante que contribuye al análisis de la efectividad de las estrategias implementadas.

Implementación de la Propuesta

La propuesta de estrategias didácticas se estructura de manera que cada actividad esté alineada con los conceptos teóricos y los hallazgos de la investigación. Las cápsulas educativas y los proyectos prácticos no solo están diseñados para ser motivadores, sino también para facilitar el aprendizaje de conceptos específicos de programación. La integración de estas actividades garantiza que los estudiantes no solo adquieran conocimientos técnicos, sino que también desarrollarán habilidades de colaboración y resolución de problemas.

Evaluación y retroalimentación

Finalmente, el proceso de evaluación está íntimamente relacionado con los métodos y objetivos de enseñanza. La evaluación continua, tanto formativa como sumativa, proporciona retroalimentación valiosa sobre la efectividad de las estrategias didácticas y permite ajustes en tiempo real, asegurando que la enseñanza se adapta a las necesidades de los estudiantes.

Descripción breve del contenido de los capítulos

Capítulo I: Marco Teórico

Este capítulo revisa la literatura y estudios previos relacionados con la enseñanza de la programación, especialmente en el contexto de estudiantes de octavo grado. Se analizan enfoques pedagógicos relevantes que han demostrado ser efectivos en la educación de programación, con un énfasis particular en el uso de plataformas como Scratch. Se discuten



también las experiencias previas de implementación de estrategias didácticas innovadoras en el aula, así como los desafíos y resultados observados en estudios anteriores. El capítulo establece un marco teórico que contextualiza la investigación y justifica la necesidad de desarrollar nuevas prácticas educativas adaptadas a las necesidades contemporáneas de los estudiantes.

Capítulo II: Metodología para el Desarrollo de la Investigación y Estudio Diagnóstico

Este capítulo detalla el diseño metodológico utilizado en la investigación. Se describe el enfoque, que combina métodos cualitativos y cuantitativos, y se justifica la selección de la muestra de estudiantes de octavo grado. Se incluyen herramientas de recolección de datos, como encuestas, entrevistas y observaciones en el aula, empleadas para evaluar la motivación y el aprendizaje en programación. Además, se presenta un estudio de diagnóstico que identifica las habilidades previas de los estudiantes y sus actitudes hacia la programación, sirviendo como base para la implementación de las estrategias didácticas en Scratch.

Capítulo III: Presentación y Validación de la Propuesta

El tercer capítulo presenta la propuesta de estrategias didácticas basadas en Scratch, detallando las actividades, recursos y metodologías diseñadas para mejorar el aprendizaje de programación. Se explica cómo se integrarán cápsulas educativas y proyectos prácticos en el aula. Se agrega un análisis sobre la validación de la propuesta, que incluye la retroalimentación obtenida de expertos en educación y de los propios estudiantes, así como la planificación de un proceso de evaluación que permitirá medir la efectividad de las estrategias implementadas. Este capítulo concluye con reflexiones sobre la aplicabilidad y el potencial impacto de la propuesta en el entorno educativo.

CAPÍTULO I: MARCO TEÓRICO

El capítulo 1 titulado Marco teórico, tiene como propósito analizar estudios previos relacionados con la enseñanza de la programación y el uso de estrategias didácticas apoyadas en Scratch, con el fin de contextualizar la presente investigación. En este capítulo se sistematizan aportes relevantes que permiten identificar enfoques, métodos y resultados significativos en investigaciones similares. Estos antecedentes fundamentan el desarrollo del Capítulo II, donde se construye el marco teórico y metodológico del estudio, y se articulan con el Capítulo III, en el que se presenta la propuesta didáctica y su proceso de validación, asegurando la coherencia integral de la investigación.

1.1 Antecedentes

En el ámbito internacional existe varios casos en lo que el uso de la plataforma Scratch aumenta ciertas habilidades en estudiantes de edad escolar, como el estudio realizado en Marruecos por Binaoui, Moubtassime & Belfakir, (2022), en donde realizaron un estudio cuasiexperimental con alumnos de sexto grado en una zona rural de Marruecos para evaluar la eficacia de Scratch en el desarrollo de competencias en programación. El grupo experimental recibió enseñanza práctica con Scratch, mientras que el control no. Los resultados mostraron una clara mejora en las competencias digitales del grupo que utilizó Scratch, además de una percepción positiva por parte de los docentes sobre su utilidad para integrar la codificación en la educación básica.

En un análisis sistemático que Gómez & Montiel, (2021), en donde reviso más de 30 artículos científicos de países como: Turquía, Reino Unido, Nueva Zelanda y Dinamarca; y que habían integrado Scratch como herramienta innovadora para desarrollar pensamiento computacional en jóvenes, a menudo complementada por formación docente y modelos pedagógicos activos orientados a conceptos clave como secuencia, bucles y variables, teniendo un nuevo enfoque pedagógico hacia la comprensión computacional.

Meegamma, Suriyaarachchi, Denny, & Nanayakkara, (2025), replicaron talleres de programación basada en sensores junto con Scratch en zonas rurales de una ciudad de Asia del Sur. Comparando grupos con y sin explicación previa de la herramienta, encontraron que ambos ganaron confianza al programar, pero el grupo que recibió una explicación previamente reportó mayor autoeficacia, creatividad e interés en carreras de TIC, sugiriendo que combinar lo didáctico con Scratch amplifica su impacto motivacional.

Un estudio realizado en la Universidad Estatal de Milagro comparó el aprendizaje de programación entre dos grupos universitarios. El grupo que usó Scratch mejoró

significativamente en conceptos como variables, estructuras condicionales y bucles, tanto en percepción como en calificaciones, en comparación con quienes usaron Java exclusivamente, (Cárdenas y otros, 2021).

Pagllacho, Egüez, & Reyes, (2024) en su investigación exploraron la enseñanza del pensamiento computacional en Ecuador utilizando Scratch Jr. mediante una metodología descriptiva de enfoque mixto. Con base en encuestas a docentes, destacó la necesidad de capacitación docente y resaltó que Scratch Jr. facilita el desarrollo de competencias críticas como resolución de problemas, pensamiento analítico y creatividad entre estudiantes de los primeros años.

Cun & Encalada, (2019) elaboraron una guía de aplicación de Scratch como herramienta pedagógica para estudiantes con discalculia. Se observó que su uso mejora la comprensión matemática, el razonamiento lógico, la resolución de problemas y aspectos socioemocionales como autoestima y trabajo en equipo. Los resultados que obtuvieron indican que Scratch ayuda a reducir desigualdades cognitivas y promueve habilidades transversales en estudiantes con necesidades educativas especiales.

1.2 Fundamentos Teóricos

1.2.1 Fundamentos de la Enseñanza de la Programación

La enseñanza de la programación se ha convertido en un eje estratégico para el desarrollo de competencias clave del siglo XXI, como la resolución de problemas, el pensamiento computacional y la llamada: alfabetización digital; fortaleciendo habilidades y competencias desde una edad temprana, (Caballero & García, 2020).

El pensamiento computacional no se limita a la informática, sino que concibe una forma de pensar y estructurar soluciones que se pueden aplicar en la vida cotidiana y en diferentes disciplinas; sus principales elementos incluyen, la descomposición de problemas complicados en partes más sencillas, se usa la perspectiva de menor a mayor, identificando así la información más relevante, reconociendo patrones y formular algoritmos que permitan resolver dichas situaciones de manera general. (Wing, 2017)

La integración del pensamiento computacional de forma general en el currículo no solo favorece a la adquisición de competencias digitales, sino que también potencia la transferencia de habilidades a otras áreas, como las artes, ciencias y matemáticas, haciendo que sea un aprendizaje más profundo y significativo, contribuyendo a la mejora del aprendizaje no solo



computacional, sino de varios ámbitos.

Zhang & Nouri, (2019) en su investigación señalan que el uso de entornos de programación visual por bloques, como Scratch, reduce significativamente la carga cognitiva que generan los lenguajes de programación textuales tradicionales. Esta simplificación en el aprendizaje permite que los estudiantes se concentren en la lógica, la secuencia y el razonamiento algorítmico, en lugar de invertir esfuerzos en memorizar y aplicar reglas complejas.

La enseñanza de la programación debería ser inclusiva y efectiva, para ello primero se debe formar al docente, obtener los conocimientos necesarios para poder enseñar a sus estudiantes, muchos profesores carecen de una preparación específica para integrar el pensamiento computacional en sus clases, lo que dificulta su alineación con los objetivos curriculares. Morales, Muñoz, & MacCann, (2025) señala que es esencial que se complete la enseñanza de la argumentación en la formación docente, ya que sería la base esencial que ayude a construir el conocimiento y el razonamiento argumentativo de los estudiantes.

Después de que el docente se prepare, lo siguiente es el aprendizaje del estudiante, las prácticas tradicionales suelen centrarse en el producto final, en que termine un curso, o un módulo, o un año escolar, dejando a un lado el proceso como el razonamiento, la colaboración entre compañeros, la depuración de la información; los instrumentos como el análisis permiten una valoración más completa del aprendizaje, (Liu y otros, 2024)

El uso de estrategias didácticas innovadoras y culturalmente pertinentes que faciliten la enseñanza de la programación es lo ideal, para esto el uso de las capsulas educativas, que comprende el micro aprendizaje, (Santana y otros, 2025), es decir, segmentos breves de no más de 5 minutos, que presenten conceptos básicos pero concretos de la programación, que incluya ejemplos del tema tratado y una pequeña retroalimentación al final de cada capsula, ayuda a mantener la atención del estudiante y que el aprendizaje sea más adaptable al ritmo del estudiante, además que al ser progresivo, el nivel de conocimiento y el ritmo de aprendizaje se entrelazan.

Los fundamentos de la enseñanza de la programación en la educación básica se sustentan en la necesidad de desarrollar pensamiento computacional, favorecer el aprendizaje interdisciplinario y garantizar la inclusión digital. Para lograrlo, es indispensable fortalecer la formación docente, diversificar las estrategias de evaluación y asegurar el acceso equitativo a recursos tecnológicos,



integrando enfoques como las cápsulas educativas que potencien la motivación y la comprensión en entornos visuales accesibles como Scratch.

1.2.2 Motivación y Aprendizaje

La motivación es un elemento central en el proceso de enseñanza-aprendizaje, actuando como motor que impulsa el compromiso, la persistencia y el rendimiento académico de los estudiantes, especialmente en áreas percibidas como complejas, como la programación. En este campo, la motivación no solo favorece la participación activa, sino que también influye directamente en la capacidad de los alumnos para superar retos cognitivos y técnicos (Chuchico y otros, 2025)

Según Bermúdez (2021), afirma que el aprendizaje basado en problemas, el aprendizaje basado en proyectos y en estudios de casos, generan una motivación en los estudiantes, siempre que se complementen con estrategias de apoyo, como la retroalimentación y la definición clara de los objetivos a conseguir; este tipo de enfoque ayuda que los estudiantes tengan un sentido de responsabilidad y de autonomía, parte esencial que favorece a la motivación y al interés genuino de realizar una tarea por medio del pensamiento lógico y la creatividad.

Parte de la autoeficacia del estudiante, es creer en la capacidad que tiene para realizar con éxito una tarea, lo percibe como relevante y con la retroalimentación adecuada se valora su esfuerzo de manera positiva, (Llanga y otros, 2019). Estos elementos que ayudan con la motivación y la autodeterminación de concluir una tarea, hechos que fortalecen la necesidad de competencia e independencia.

Como parte de potenciar la motivación está la estrategia del micro aprendizaje, el diseño instruccional organiza el contenido que sea breve, focalizada y fácilmente digeribles, (Durán & Escudero, 2023). En el ámbito escolar, los objetivos educativos dependen de la alineación de los contenidos que el docente prepara, al realizar cápsulas educativas utilizando el micro aprendizaje, no solo hace más fácil la comprensión del tema, sino que se adapta a las necesidades de los estudiantes.

Estas cápsulas permiten introducir un concepto específico de programación, mostrar ejemplos prácticos, invitar a la experimentación inmediata y ofrecer retroalimentación rápida. Su brevedad y enfoque práctico facilitan la atención sostenida, la comprensión progresiva y la motivación intrínseca, ya que cada cápsula representa un reto alcanzable y significativo para el estudiante. Además, al estar integradas en un entorno visual como Scratch, reducen la ansiedad asociada a la sintaxis de los lenguajes textuales y promueven un aprendizaje más lúdico y

accesible.

La motivación en el aprendizaje de la programación depende de un conjunto de factores pedagógicos, emocionales y contextuales. Estrategias como el micro aprendizaje, combinadas con entornos visuales interactivos, constituyen un recurso valioso para despertar y mantener el interés de los estudiantes, al mismo tiempo que favorecen la adquisición gradual de competencias de pensamiento computacional y resolución de problemas.

1.2.3 Scratch como Herramienta Didáctica

Scratch es una herramienta desarrollada por el MIT Media Lab en el 2003, un entorno de programación visual basado en bloques que permite a los usuarios crear animaciones, juegos, simulaciones y presentaciones interactivas sin necesidad de escribir código textual. Su diseño intuitivo, basado en el arrastre y ensamblaje de bloques lógicos, elimina la complejidad sintáctica de los lenguajes tradicionales y facilita que estudiantes de todas las edades se concentren en la lógica, la secuencia y el razonamiento algorítmico (Zhang & Nouri, 2019).

Grané & Sánchez (2021) realizó una revisión metodología sobre los programas educativos que enseñan pensamiento computacional y determinó que el propósito principal de Scratch es acercar la programación a todas las edades y contextos, fomentando la creatividad, el pensamiento lógico y la capacidad de resolver problemas.

Al eliminar la complejidad, el estudiante puede concentrarse en el diseño de algoritmos y en la construcción de soluciones. Este enfoque lo convierte en una herramienta idónea para introducir conceptos de programación en la educación básica y media, tanto en entornos formales como no formales.

Scratch favorece el aprendizaje interdisciplinar. En matemáticas, apoya la visualización de coordenadas y figuras geométricas; en ciencias, la simulación de fenómenos; en lengua, la construcción de relatos interactivos; y en arte, la animación digital. Esta flexibilidad permite que los estudiantes integren conocimientos de distintas áreas, fortaleciendo así un aprendizaje significativo.

La comunidad en línea de Scratch constituye otro de sus valores educativos, ya que ofrece un espacio para publicar, compartir y comentar proyectos, lo que fomenta el aprendizaje colaborativo y la retroalimentación entre pares. (Pérez y otros, 2020)



Scratch no es solo un lenguaje de programación visual, sino una plataforma pedagógica que integra creación, colaboración y reflexión. Su impacto positivo en la enseñanza depende, no obstante, de su integración en estrategias didácticas bien planificadas, con actividades guiadas, objetivos claros y retroalimentación oportuna. Cuando se utiliza de esta manera, contribuye al desarrollo integral del estudiante, combinando habilidades técnicas, cognitivas y sociales que son esenciales en la sociedad actual.

Para estudiantes de octavo año, Scratch ofrece ventajas pedagógicas concretas, y las estrategias didácticas que se pueden implementar con esta plataforma son las siguientes:

- **Aprendizaje por proyectos:** Diseñar proyectos interdisciplinarios, es decir, simulación de fenómenos físicos, juegos que integren problemas matemáticos, relatos interactivos para lengua) donde los estudiantes planifiquen, programen y presenten su producto. La metodología por proyectos potencia la motivación y el aprendizaje situado. Además de utilizar casos de la vida cotidiana, ayuda a mejorar su pensamiento crítico en cualquier tipo de situación.
- **Scaffolding progresivo y rubricas:** El scaffolding según Exposito (2023), es una estrategia donde el docente proporciona apoyo necesario, adaptandose al ritmo de aprendizaje del estudiante para que pueda dominar nuevos conocimientos. De esa forma utilizando las capsulas educativa en conjunto con la plataforma de Scratch, las tareas enviadas son guiadas, con tutoria remota y presencial, avanzando en proyectos y con retroalimentacion formativa sobre estructuras algoritmicas y uso de variables.
- **Trabajo colaborativo y roles:** El trabajo en equipo es esencial, por lo que se le podría dar roles dentro del grupo de trabajo, como: los programadores, los diseñadores y los evaluadores. Al “jugar” están favoreciendo la comunicación y reforzando sus conocimientos, puesto que se concentran en una tarea específica tomando en cuenta la tarea anterior y posterior, es decir, comparten información del avance del proyecto que están trabajando, lo revisan y construyen de a poco el conocimiento de todos.

Scratch facilita la materialización de ideas abstractas (algoritmos, bucles, condicionales) en objetos manipulables, lo que baja la barrera cognitiva de la sintaxis textual y permite enfocarse en la lógica. El diseño por proyectos y la naturaleza lúdica incrementan la motivación y el



compromiso del alumnado, elementos cruciales en la adolescencia temprana. (Monjelat & San, 2015)

La eficacia depende en gran medida de la orientación pedagógica: la herramienta por sí sola no garantiza aprendizajes profundos si se usa de forma instrumental, como: repetir tutoriales sin contextualización. Las barreras más importantes no son tecnológicas sino pedagógicas y organizativas: formación insuficiente del profesorado, diseño de actividades de baja complejidad, falta de ajuste curricular y escasez de recursos informáticos. Superar estas limitaciones requiere formación docente específica y planificación institucional.

Scratch es una herramienta muy adecuada para alumnos de octavo año cuando se integra mediante secuencias didácticas bien diseñadas, con objetivos curriculares claros, evaluación formativa y formación docente. Su fuerza radica en la capacidad de transformar conceptos abstractos en objetos interactivos y en posibilitar enfoques interdisciplinarios y colaborativos; sus riesgos aparecen si se limita a tutoriales descontextualizados o falta acompañamiento pedagógico institucional.

1.2.4 Aplicación de Estrategias de Enseñanza en Entornos Digitales

La integración de entornos digitales en la enseñanza ha transformado significativamente las metodologías de trabajo en el aula, ofreciendo nuevas oportunidades para el aprendizaje activo, personalizado y colaborativo. En el ámbito de la programación, las estrategias didácticas más efectivas suelen combinar diversos enfoques para potenciar la motivación, la comprensión conceptual y la retención a largo plazo de los contenidos.

Una de las metodologías más destacadas es el aprendizaje basado en proyectos, que plantea a los estudiantes la resolución de problemas, este enfoque fomenta la investigación autónoma, la creatividad y la aplicación práctica de los conocimientos, favoreciendo un aprendizaje significativo (Larmer y otros, 2015). Cuando se integra en entornos digitales, el aprendizaje basado en proyectos permite la creación de productos finales interactivos, como juegos o simulaciones en Scratch, que pueden ser compartidos con la comunidad, lo que añade un componente motivador.

La gamificación ligera es otra estrategia relevante, que consiste en la aplicación de elementos propios de los juegos (como retos, niveles, insignias y recompensas simbólicas) con el fin de



incrementar el compromiso y la perseverancia. A diferencia de la ludificación intensiva, la gamificación ligera mantiene el foco en la maestría del contenido y en la superación personal, evitando la dependencia excesiva de recompensas externas (Domínguez y otros, 2013). En programación, esto puede aplicarse proponiendo misiones progresivas que conduzcan al dominio de conceptos clave, como bucles o condicionales.

La evaluación formativa es igualmente esencial, ya que permite a docentes y estudiantes monitorear el progreso de manera continua, identificar errores y corregirlos antes de la evaluación sumativa. En el desarrollo de pensamiento computacional, se recomienda el uso de rúbricas específicas que valoren no solo el producto final, sino también el proceso: planificación, pruebas, depuración y documentación del código (Brennan & Resnick, 2012).

Por otra parte, el micro aprendizaje se presenta como una estrategia idónea para dosificar contenidos técnicos en unidades breves y focalizadas, adaptadas a la capacidad de atención de los estudiantes y al ritmo de aprendizaje individual. En programación, las cápsulas de 3 a 5 minutos con ejemplos prácticos permiten introducir conceptos sin sobrecargar la memoria de trabajo, ofreciendo la posibilidad de repasar y reforzar el aprendizaje a demanda (Almenara & Ruiz, 2018).

En contextos de programación introductoria, el uso de bloques didácticos resulta fundamental. Estos incluyen plantillas prediseñadas, ejemplos mínimos viables y guías de depuración que orientan a los estudiantes en las primeras fases del aprendizaje, reduciendo la frustración y aumentando la confianza. Scratch, por su naturaleza visual e interactiva, favorece este proceso, ya que la ejecución inmediata de los proyectos y los mensajes de error implícitos facilitan la comprensión y el ajuste de las instrucciones.

La retroalimentación inmediata es un factor clave para fortalecer la autoeficacia, entendida como la creencia del estudiante en su capacidad para alcanzar con éxito los objetivos de aprendizaje. En entornos digitales, esta retroalimentación puede generarse de forma automática (por la propia herramienta) o a través de la interacción directa con el docente, combinando lo mejor de la automatización y el acompañamiento humano.

El diseño de estrategias de enseñanza en entornos digitales para la programación debe ser integral, combinando metodologías activas, recursos de gamificación, evaluación formativa,

Estrategia Didáctica	Descripción	Ventajas	Desventajas
Aprendizaje Basado en Proyectos (ABP)	Enfoque centrado en la resolución de problemas reales mediante proyectos. Los estudiantes trabajan en grupos para investigar, diseñar y presentar soluciones.	<ul style="list-style-type: none"> Fomenta el trabajo en equipo y la colaboración. Desarrolla habilidades prácticas y críticas. Aumenta la motivación y el compromiso. 	<ul style="list-style-type: none"> Requiere tiempo significativo para planificación y ejecución. Puede ser difícil evaluar el aprendizaje individual.
Microlearning	Método de enseñanza que ofrece contenido en pequeñas dosis, facilitando el aprendizaje en sesiones cortas y enfocadas. Se puede utilizar a través de videos, infografías o módulos breves.	<ul style="list-style-type: none"> Flexibilidad en el aprendizaje. Facilita la retención de información. Ideal para temas específicos y rápidos. 	<ul style="list-style-type: none"> Puede no ser suficiente para conceptos complejos. Riesgo de fragmentación del conocimiento si no se organiza bien.
Scaffolding	Estrategia que proporciona apoyo temporal a los estudiantes mientras adquieren nuevas habilidades. A medida que los estudiantes se vuelven más competentes, el apoyo se retira gradualmente.	<ul style="list-style-type: none"> Promueve la autonomía y la confianza en el aprendizaje. Ayuda a los estudiantes a superar dificultades. Facilita la comprensión de conceptos complejos. 	<ul style="list-style-type: none"> Requiere una planificación cuidadosa y atención individualizada. Puede ser difícil determinar el momento adecuado para retirar el apoyo.
Colaboración	Enfoque que fomenta el trabajo en equipo entre los estudiantes para lograr objetivos comunes. Se basa en la interacción y el intercambio de ideas.	<ul style="list-style-type: none"> Mejora la comunicación y las habilidades interpersonales. Enriquecimiento del aprendizaje a través del intercambio de perspectivas. Fomenta un sentido de comunidad en el aula. 	<ul style="list-style-type: none"> Puede haber desigualdades en la participación. Dificultades en la coordinación y resolución de conflictos dentro del grupo.

micro aprendizaje y andamiaje pedagógico, de manera que se optimicen tanto los resultados de aprendizaje como la motivación del estudiante.

Esta tabla comparativa de las estrategias didácticas: Aprendizaje Basado en Proyectos (ABP), Microlearning, Scaffolding y Colaboración. Describe la ventaja y desventaja. permite a los docentes a elegir lo más adecuada para alcanzar los objetivos de aprendizaje.

Tabla 1 Estrategias Didácticas

Cada una de estas estrategias didácticas ofrece enfoques únicos para facilitar el aprendizaje y

la enseñanza. Al considerar sus ventajas y desventajas, los docentes pueden seleccionar y combinar estrategias que mejor se adapten a las necesidades de sus estudiantes y al contexto educativo, optimizando así el proceso de aprendizaje.

Esta tabla de estrategia didáctica tiene características únicas que pueden ser efectivas en diferentes contextos educativos. Comprender sus ventajas y limitaciones permite a los educadores elegir la más adecuada para alcanzar los objetivos de aprendizaje.

Estrategia Didáctica	Descripción	Ventajas	Limitaciones
Aprendizaje Basado en Proyectos (ABP)	Enfoque centrado en la resolución de problemas reales mediante proyectos.	<ul style="list-style-type: none"> Fomenta el trabajo en equipo y la colaboración. Desarrolla habilidades prácticas y críticas. Aumenta la motivación y el compromiso. 	<ul style="list-style-type: none"> Requiere tiempo significativo para planificación y ejecución. Puede ser difícil evaluar el aprendizaje individual.
Gamificación	Incorporación de elementos de juego en entornos educativos para aumentar la motivación y el compromiso.	<ul style="list-style-type: none"> Incrementa la participación y el interés. Facilita el aprendizaje a través de la diversión. Promueve la competencia sana y el reconocimiento. 	<ul style="list-style-type: none"> Puede distraer si no se implementa adecuadamente. Dependencia de la tecnología y recursos adicionales.
Microlearning	Método de enseñanza que ofrece contenido en pequeñas dosis, facilitando el aprendizaje en sesiones cortas y enfocadas.	<ul style="list-style-type: none"> Flexibilidad en el aprendizaje; se adapta a horarios ocupados. Facilita la retención de información. Ideal para temas específicos y rápidos. 	<ul style="list-style-type: none"> Puede no ser suficiente para conceptos complejos. Riesgo de fragmentación del conocimiento si no se organiza bien.

Tabla 2 Estrategias didácticas 2

Comprender sus ventajas y limitaciones permite a los educadores seleccionar la estrategia más adecuada para alcanzar los objetivos de aprendizaje y adaptarse a las necesidades de sus estudiantes.

1.2.5 Aprendizaje Basado en Proyectos

El aprendizaje basado en proyectos sitúa a los estudiantes ante un reto auténtico que culmina en un producto público. La literatura en español reporta mejoras en motivación, trabajo en



equipo y organización del aprendizaje cuando se aplica el aprendizaje basado en proyectos en educación obligatoria. Estos autores subrayan la necesidad de fases, pautas y retroalimentación docente para asegurar calidad del proceso y del producto (García y otros, 2017).

Aplicado a Scratch, el aprendizaje basado en proyectos permite que contenidos de diversas áreas se integren en artefactos digitales (historias interactivas, simulaciones, videojuegos educativos). Experiencias de propuesta didáctica en secundaria han mostrado la viabilidad de combinar el aprendizaje basado en proyectos con Scratch, incluso con estudiantes con necesidades específicas, incorporando actividades de complejidad creciente y rúbricas de evaluación. (Barberán, 2016)

El aprendizaje basado en proyectos se ha consolidado como una de las metodologías más significativas dentro de los enfoques pedagógicos activos, al situar al estudiante en el centro del proceso formativo. Este modelo no solo promueve la adquisición de conocimientos conceptuales, sino que también favorece el desarrollo de competencias transversales como la comunicación efectiva, la resolución de problemas y el pensamiento crítico. Una de sus principales fortalezas radica en que los estudiantes no trabajan con tareas fragmentadas o descontextualizadas, sino que enfrentan retos integrales, ligados a situaciones reales o simuladas de gran relevancia, lo que incrementa la pertinencia y la motivación hacia el aprendizaje.

Asimismo, el aprendizaje basado en proyectos potencia la colaboración entre pares, ya que los proyectos suelen implicar tareas interdependientes que requieren coordinación, planificación y negociación. De esta forma, los alumnos aprenden a valorar las aportaciones de los demás, a asumir responsabilidades compartidas y a distribuir roles, acercándose a la dinámica propia de los entornos profesionales. A ello se suma que los productos resultantes de los proyectos trascienden lo meramente escolar, pues suelen ser presentados en espacios públicos o compartidos con la comunidad, lo que otorga un mayor sentido de logro y compromiso.

Sin embargo, la implementación del aprendizaje basado en proyectos demanda una planificación cuidadosa por parte de los docentes. No basta con asignar un proyecto; es necesario diseñar fases claras que orienten el trabajo de los estudiantes, establecer criterios de evaluación transparentes y brindar retroalimentación continua que ayude a corregir errores y a mejorar la calidad del proceso. La labor del profesor se transforma en la de un facilitador y guía

que acompaña el aprendizaje, equilibra la autonomía del grupo y garantiza que los objetivos curriculares sean alcanzados. De esta manera, el aprendizaje basado en proyectos se convierte en una estrategia poderosa para vincular teoría y práctica en la educación contemporánea.

1.2.6 Scaffolding progresivo y rúbricas con Scratch

Scaffolding, también conocido como andamiaje. Consiste en ofrecer apoyos temporales ajustados a la “zona de desarrollo próximo”, retirándolos gradualmente a medida que el estudiante gana competencia. En la tradición constructivista hispana, se enfatiza el papel del profesor como mediador que ajusta ayudas, pauta metas intermedias y fomenta la autorregulación. (Exposito, 2023)

Además, herramientas como Scratch proporcionan retroalimentación automática del “nivel de pensamiento computacional” presente en un proyecto ya sea abstracción, paralelismo, control de flujo, representación de datos, etc., útil como complemento a la rúbrica para centrar la reflexión del alumnado en aspectos computacionales.

El concepto de apoyos temporales en el aprendizaje responde a la idea de que los estudiantes requieren acompañamiento diferenciado mientras consolidan nuevas habilidades. Este enfoque, también conocido como andamiaje, reconoce que el aprendizaje no es un proceso lineal ni homogéneo, sino que avanza en función de las capacidades previas y del contexto de cada estudiante. El rol del docente, en este sentido, es crucial: debe identificar las necesidades inmediatas, ofrecer ayudas pertinentes y, de manera estratégica, ir retirándolas para que el alumno logre autonomía y confianza en su desempeño.

El andamiaje no se limita a ofrecer explicaciones adicionales, sino que abarca la estructuración de tareas, la división en metas alcanzables, la modelación de procedimientos y la provisión de herramientas que faciliten la comprensión. Al mismo tiempo, fomenta la autorregulación, ya que los estudiantes aprenden a monitorear su propio progreso, reconocer sus dificultades y buscar soluciones por sí mismos. Este proceso resulta particularmente valioso en la enseñanza de habilidades complejas como la programación y el pensamiento computacional.

Las plataformas educativas como Scratch representan un apoyo significativo, pues ofrecen un entorno visual e intuitivo que facilita el aprendizaje progresivo de conceptos computacionales. Su capacidad de proporcionar retroalimentación inmediata permite que los estudiantes tomen



conciencia de la lógica de sus proyectos y reflexionen sobre aspectos como la abstracción, la secuenciación de instrucciones, el uso de eventos o la organización de datos. Estos elementos no solo enriquecen el proceso de aprendizaje, sino que también permiten a los docentes contar con evidencias objetivas para evaluar el nivel de desarrollo alcanzado.

Así, la combinación de un andamiaje pedagógico bien diseñado y de herramientas tecnológicas adecuadas ofrece un marco potente para potenciar el aprendizaje autónomo y significativo. Se trata de un equilibrio dinámico entre guía docente, autonomía estudiantil y aprovechamiento de recursos digitales.

1.2.7 Trabajo Colaborativo y Roles en Scratch

La revisión sistemática de Revelo, Collazos, & Jimenez (2018) muestra que el trabajo colaborativo en cursos de programación incrementa la participación, favorece el aprendizaje y aporta beneficios en aspectos como la resolución de problemas y la motivación, cuando se estructura explícitamente como: objetivos, roles, interdependencia positiva y evaluación de procesos.

Según, Revelo, Collazos, & Jimenez (2018) el trabajo colaborativo con roles hace operativos los procesos sociales del aprendizaje: obliga a explicitar ideas, negociar decisiones de diseño, distribuir responsabilidades y sostener la interdependencia positiva. La evidencia de la revisión de 2018 sugiere que, en programación, la colaboración sistematizada, con estructura y evaluación, incrementa la participación y mejora el rendimiento, lo cual es especialmente relevante en edades de 12–13 años, donde el clima de aula y la motivación social inciden fuertemente en el compromiso con la tarea.

El trabajo colaborativo en el ámbito educativo constituye una estrategia clave para potenciar el aprendizaje significativo, especialmente en áreas complejas como la programación. A través de la colaboración, los estudiantes no solo adquieren conocimientos técnicos, sino que también desarrollan habilidades socioemocionales fundamentales para desenvolverse en entornos académicos y profesionales. La distribución de roles, por ejemplo, asegura que cada integrante asuma responsabilidades concretas, lo que evita la pasividad y promueve la participación activa de todos los miembros del grupo. Este reparto equitativo fomenta además la interdependencia positiva, ya que el éxito colectivo depende de la contribución individual.



La colaboración en programación impulsa la verbalización de ideas, lo que obliga a los estudiantes a estructurar su pensamiento y a comunicarlo con claridad. Este ejercicio de explicitación favorece la comprensión mutua, el contraste de perspectivas y la construcción de soluciones más robustas a problemas complejos. La negociación de decisiones, por su parte, enseña a llegar a consensos y a respetar opiniones diversas, fortaleciendo la cohesión grupal y el sentido de pertenencia al equipo.

Otro aspecto crucial es la motivación social que se genera en los grupos de trabajo. En edades tempranas, particularmente en la adolescencia, los vínculos interpersonales tienen un peso considerable en el compromiso con las actividades escolares. El trabajo colaborativo canaliza esta motivación hacia objetivos académicos, convirtiendo el clima de aula en un entorno más dinámico, participativo y productivo. Además, la evaluación sistemática de procesos no solo mide resultados, sino que también permite reflexionar sobre las dinámicas internas del grupo, identificando fortalezas y aspectos a mejorar.

En síntesis, cuando se organiza de manera estructurada, el trabajo colaborativo en programación no solo mejora el rendimiento académico, sino que también constituye un espacio de aprendizaje integral donde se desarrollan competencias técnicas, comunicativas y sociales imprescindibles para el futuro.

1.3 Bases Legales

1.3.1 La Declaración Universal de los Derechos Humanos.

El artículo 26 de la declaración universal de los derechos humanos, dice lo siguiente:

1. Toda persona tiene derecho a la educación. La educación debe ser gratuita, al menos en lo concerniente a la instrucción elemental y fundamental. La instrucción elemental será obligatoria. La instrucción técnica y profesional habrá de ser generalizada; el acceso a los estudios superiores será igual para todos, en función de los méritos respectivos.
2. La educación tendrá por objeto el pleno desarrollo de la personalidad humana y el fortalecimiento del respeto a los derechos humanos y a las libertades fundamentales; favorecerá la comprensión, la tolerancia y la amistad entre todas las naciones y todos los grupos étnicos o religiosos, y promoverá el desarrollo de las actividades de las Naciones Unidas para el mantenimiento de la paz. (Unidas, 2023)

El artículo 26, de la Declaración Universal de los Derechos Humanos, respalda la educación



como una parte importante del ser humano, desde los estudios iniciales a los superiores, lo cual permite al ser humano desarrollarse en sus competencias permitiendo una sociedad educada.

1.3.2 Constitución de la República Del Ecuador

Los artículos 16 y 347 de la Constitución de la República dicen:

Art. 16.- Todas las personas, en forma individual o colectiva, tienen derecho a: 2. El acceso universal a las tecnologías de información y comunicación. (Ecuador, 2021)

El artículo mencionado anteriormente, establece el derecho universal de las personas, tanto de forma individual como colectiva, al acceso a las TIC. Este precepto garantiza que la tecnología no sea un privilegio de unos pocos, sino un recurso accesible para toda la población, lo cual es clave para disminuir la brecha digital y promover la inclusión social.

Art. 347.- Será responsabilidad del Estado: 8. Incorporar las tecnologías de la información y comunicación en el proceso educativo y propiciar el enlace de la enseñanza con las actividades productivas o sociales. (Ecuador, 2021)

El art 347, refuerza esta idea al señalar que es responsabilidad del Estado incorporar las TIC en el proceso educativo. Este mandato busca que la educación no se limite a la enseñanza tradicional, sino que utilice herramientas tecnológicas para modernizar la enseñanza, facilitar el aprendizaje y vincularlo con las actividades productivas y sociales. Esto implica que la formación académica debe preparar a los estudiantes para desenvolverse en un mundo digital y laboralmente competitivo.

Sección tercera: Comunicación e Información

Art. 17.- El Estado fomentará la pluralidad y la diversidad en la comunicación, y al efecto: 1. Garantizará la asignación, a través de métodos transparentes y en igualdad de condiciones, de las frecuencias del espectro radioeléctrico, para la gestión de estaciones de radio y televisión públicas, privadas y comunitarias, así como el acceso a bandas libres para la explotación de redes inalámbricas, y precautelará que en su utilización prevalezca el interés colectivo. (Ecuador, 2021)

En este artículo, resalta el compromiso del Estado de garantizar la pluralidad y diversidad en la comunicación, así como el acceso transparente y equitativo a los medios y redes de comunicación, incluidos el espectro radioeléctrico y las redes inalámbricas. Esto no solo fomenta la libertad de expresión, sino que asegura que la tecnología y los medios sean gestionados con un enfoque de interés colectivo, evitando monopolios o exclusiones.

Sección quinta: Educación





Art. 26.- La educación es un derecho de las personas a lo largo de su vida y un deber ineludible e inexcusable del Estado. Constituye un área prioritaria de la política pública y de la inversión estatal, garantía de la igualdad e inclusión social y condición indispensable para el buen vivir. Las personas, las familias y la sociedad tienen el derecho y la responsabilidad de participar en el proceso educativo. (Ecuador, 2021)

El art. 26, declara la educación como un derecho humano fundamental y un deber inexcusable del Estado, reconociéndola como una prioridad en la política pública. Este artículo establece que la educación es un medio para lograr igualdad, inclusión social y bienestar, así como una responsabilidad compartida entre el Estado, la familia y la sociedad.

Art. 27.- La educación se centrará en el ser humano y garantizará su desarrollo holístico, en el marco del respeto a los derechos humanos, al medioambiente sustentable y a la democracia; será participativa, obligatoria, intercultural, democrática, incluyente y diversa, de calidad y calidez; impulsará la equidad de género, la justicia, la solidaridad y la paz; estimulará el sentido crítico, el arte y la cultura física, la iniciativa individual y comunitaria, y el desarrollo de competencias y capacidades para crear y trabajar. La educación es indispensable para el conocimiento, el ejercicio de los derechos y la construcción de un país soberano, y constituye un eje estratégico para el desarrollo nacional. (Ecuador, 2021)

En el art. 27, profundiza en el enfoque que debe tener la educación: debe ser centrada en la persona, integral, inclusiva y de calidad, fomentando el pensamiento crítico, la creatividad, la equidad de género, la justicia, la solidaridad y la paz. También resalta que la educación es indispensable para el ejercicio de los derechos, el fortalecimiento de la soberanía y el desarrollo nacional, lo que implica que su alcance trasciende la formación académica para convertirse en un motor de cambio social.

Los artículos de la Constitución mencionados anteriormente se complementan para establecer un marco donde el acceso a la tecnología, la equidad en la comunicación y una educación integral y tecnológica son derechos fundamentales y, a la vez, responsabilidades estatales y sociales. Este enfoque busca formar ciudadanos críticos, competentes y comprometidos con el desarrollo sostenible y democrático del país. Respaldando la investigación ya que indican sobre la importancia de la educación en el ser humano y las diversas herramientas que nos brindan para establecer una sociedad tranquila y con conciencia social.

CAPÍTULO II: METODOLOGÍA PARA EL DESARROLLO DE LA INVESTIGACIÓN Y ESTUDIO DIAGNÓSTICO

El capítulo II titulado metodología para el desarrollo de la investigación y estudio diagnóstico, tiene como propósito sustentar teórica y metodológicamente la investigación, desarrollando conceptos clave sobre la enseñanza de la programación, el pensamiento computacional y el uso de Scratch como herramienta educativa. En este capítulo se presentan los enfoques pedagógicos y el diseño metodológico que orientan el estudio. Su contenido se fundamenta en los antecedentes investigativos analizados en el Capítulo I y proporciona el soporte conceptual y metodológico necesario para el diseño e implementación de la propuesta didáctica desarrollada en el Capítulo III, asegurando la coherencia y continuidad del proceso investigativo.

2.1 Conceptualización y operacionalización de las categorías y variables

En la presente investigación se identifican como variables centrales el aprendizaje de la programación y la motivación hacia la programación, las cuales constituyen los ejes analíticos para comprender el fenómeno educativo estudiado en los estudiantes de octavo grado de la asignatura de Computación de la Escuela Particular de Educación Básica “Pablo Palacio”.

La motivación hacia la programación se concibe como el conjunto de actitudes, intereses, percepciones y disposiciones internas que influyen en la participación activa del alumno frente al aprendizaje de contenidos programáticos. Desde una perspectiva educativa, la motivación incide directamente en el compromiso, la persistencia y la disposición del alumno para enfrentar tareas cognitivamente demandantes, como aquellas propias de la programación informática. En este sentido, una adecuada motivación favorece la implicación del estudiante en las actividades académicas y potencia su proceso de aprendizaje.

Según Oroxom (2022) las estrategias didácticas no deben ser forzadas ni obligatorias sino propositivas e intencionales lo cual permite al estudiante realizar una comparación entre el conocimiento nuevo y el predecesor, lo cual esto se apoya con técnicas eficaces para lograr un aprendizaje significativo, por este motivo las estrategias didácticas deben tener una relación muy cercana con el desarrollo de habilidades y destrezas (p. 6).

Por su parte, el aprendizaje de la programación se entiende como el proceso mediante el cual los estudiantes adquieren, comprenden y aplican conocimientos básicos relacionados con la lógica computacional, la secuenciación de instrucciones y la resolución de problemas mediante el uso de lenguajes de programación educativos. Este aprendizaje no se limita a la adquisición de conocimientos teóricos, sino que implica el desarrollo de habilidades cognitivas y



procedimentales necesarias para la construcción de soluciones digitales, lo cual resulta fundamental en el contexto de la educación básica.

Con el propósito de analizar empíricamente las variables de investigación, estas fueron operacionalizadas a partir de dimensiones e indicadores que permitieron su medición objetiva y sistemática. La motivación hacia la programación fue evaluada mediante indicadores relacionados con el interés, la participación y la disposición hacia el aprendizaje, utilizando técnicas de encuesta y cuestionarios estructurados. En cuanto al aprendizaje de la programación, se consideraron indicadores vinculados con la comprensión de conceptos básicos, la aplicación práctica y el desempeño académico, los cuales fueron medidos a través de instrumentos de evaluación específicos y observación del desempeño estudiantil.

Esta operacionalización permitió establecer una relación coherente entre las variables, los objetivos de la investigación y las técnicas de recolección de datos empleadas, garantizando así la validez metodológica del estudio y una adecuada interpretación de los resultados obtenidos.

Variables	Estrategias Didácticas	Enseñanza de programación
Conceptualización	Las estrategias didácticas permiten al educador orientar el recorrido pedagógico, estableciendo procedimientos que deben seguir los educandos para construir sus aprendizajes. (Herrera & Villafuerte, 2023)	La enseñanza de la programación a los niños es fundamental ya que aumenta la creatividad y la capacidad de razonar de mejor manera, para poder resolver problemas del ámbito educativo y de la vida diaria. (Angamarca & Andrade, 2022)
Operacionalización	Para la medición de su valoración, se elaboró una escala numérica del 1 a 3, siendo 1 la mínima calificación y 3 la máxima. De igual forma, contó con un espacio para que los jueces registraran las observaciones y consideraciones necesarias. (Peña, 2022)	Se tomó en cuenta la encuesta inicial, en el cual se evaluó nuevamente a los estudiantes con el fin de determinar el rango de mejoramiento en cuanto al razonamiento lógico-matemático.
Indicadores y dimensiones	-Intencionalidad Pedagógica (5) -Flexibilidad (5) -Contextualización (5) -Evaluabilidad (5)	-Recursos (5) -Conocimientos (5) -Habilidades (5)
Escala – Valores	Escala 1.- Totalmente en desacuerdo 2.- En desacuerdo 3.- Indiferente 4.- De acuerdo 5.- Totalmente de acuerdo	Escala 1.- Nunca 2.- Raramente 3.- Ocasionalmente 4.- Frecuentemente 5.- Muy Frecuentemente
Muestra	30 estudiantes de octavo grado	30 estudiantes de octavo grado
Técnica o herramienta	Cuestionario	Cuestionario

Tabla 3 Operacionalización

2.2 El enfoque de la investigación.

La presente investigación se desarrolló bajo un enfoque mixto, ya que combina métodos cuantitativos y cualitativos con el propósito de analizar el impacto de las estrategias didácticas basadas en Scratch en la motivación y el aprendizaje de la programación en estudiantes de octavo grado. El enfoque cuantitativo permitió medir los niveles de motivación y aprendizaje mediante instrumentos estructurados, mientras que el enfoque cualitativo facilitó la comprensión de las percepciones y comportamientos de los estudiantes durante el proceso de



implementación. Este enfoque integral permitió una interpretación más completa del fenómeno educativo estudiado.

2.3 Justificación del enfoque mixto

Proporciona una visión más holística del fenómeno educativo, explorando tanto los aspectos medibles como las dinámicas sociales y emocionales.

Permite adaptarse a diferentes contextos y necesidades, ajustando los métodos de recolección de datos según la situación.

El enfoque mixto es más adecuado para esta investigación, ya que permite integrar datos cuantitativos y cualitativos, ofreciendo una compleción más enriquecedora y profunda de las estrategias didácticas basadas en Scratch y su impacto en el aprendizaje de programación en estudiantes de octavo grado.

Como afirma Ascona & Mencia, (2023) el enfoque mixto ofrece la posibilidad de integrar datos cualitativos como cuantitativos, lo que permite abordar las diversas preguntas de la investigación de diversos ángulos lo cual alimenta y enriquece la investigación, ya que nos da apertura a bastantes posibilidades durante la investigación (p. 117).

2.4 Alcance de la investigación

La presente investigación posee un alcance explicativo, ya que se orienta a identificar y analizar las relaciones causales entre la implementación de actividades de programación con Scratch y los niveles de motivación de los estudiantes de octavo grado. Desde este enfoque, no solo se describe el fenómeno, sino que se busca comprender cómo y por qué la intervención pedagógica influye en las actitudes, el interés y la participación del estudiantado. El estudio pretende, por tanto, ofrecer evidencia empírica que permita explicar los cambios observados en la motivación académica a partir de la interacción con recursos tecnológicos y actividades prácticas de programación.

Asimismo, el alcance explicativo permite establecer conexiones entre las variables centrales del estudio, integrando datos cuantitativos y cualitativos para sustentar las interpretaciones con un enfoque integral. A través del análisis de encuestas, observaciones directas y evaluaciones de desempeño, se busca determinar los factores que contribuyen de manera significativa al fortalecimiento de la motivación estudiantil. De este modo, la investigación trasciende la mera descripción, proporcionando explicaciones fundamentadas que pueden orientar futuras intervenciones educativas y aportar a la comprensión del impacto de la programación visual



como estrategia didáctica.

2.5 Declaración y justificación del tipo de investigación

La presente investigación se sustenta en un tipo de investigación documental–bibliográfica, debido a que requirió la revisión, análisis y sistematización de fuentes teóricas y antecedentes científicos relacionados con la enseñanza de la programación, el uso educativo de Scratch y las estrategias didácticas orientadas a fortalecer la motivación y el aprendizaje en contextos escolares. Este tipo de investigación permitió fundamentar teóricamente el estudio, sustentar el marco conceptual y orientar el diseño de la propuesta didáctica, en coherencia con los objetivos planteados.

De manera complementaria, la investigación es de campo, puesto que la recolección de datos se realizó directamente en el contexto real donde se manifiesta el problema de estudio, específicamente en la asignatura de Computación de la Escuela Particular de Educación Básica “Pablo Palacio”, con la participación de estudiantes de octavo grado. Este tipo de investigación permitió obtener información empírica mediante la aplicación de técnicas como encuestas, observación directa y evaluación de proyectos, lo que facilitó el análisis de la motivación y el aprendizaje de la programación en la población investigada.

En cuanto a su temporalidad, la investigación es de carácter transversal, ya que los datos fueron recopilados en un único periodo académico, permitiendo analizar las condiciones y efectos derivados de la implementación de la estrategia didáctica sin recurrir a mediciones longitudinales. Esta decisión metodológica se ajusta a los objetivos del estudio y a las características del entorno educativo, garantizando la viabilidad del proceso investigativo y la coherencia entre el diseño metodológico, las técnicas empleadas y la población objeto de estudio.

Como afirma Ascona & Mencia, (2023) el enfoque mixto ofrece la posibilidad de integrar datos cualitativos como cuantitativos, lo que permite abordar las diversas preguntas de la investigación de diversos ángulos lo cual alimenta y enriquece la investigación, ya que nos da apertura a bastantes posibilidades durante la investigación (p. 117).

2.6 Métodos empleados y sus propósitos en el contexto de la investigación

La presente investigación se sustenta en un enfoque mixto, al integrar métodos cuantitativos y cualitativos con el propósito de analizar de manera integral la relación entre la implementación de estrategias didácticas basadas en Scratch y los niveles de motivación y aprendizaje de

programación en los estudiantes de octavo grado.

Desde el punto de vista teórico, se emplean métodos como el análisis documental y la revisión bibliográfica, los cuales permiten fundamentar conceptualmente las variables de estudio, especialmente la motivación académica, el aprendizaje de la programación y el uso de estrategias didácticas digitales.

En el ámbito empírico, se utiliza el método explicativo, orientado a identificar y analizar la relación causal entre la aplicación de la estrategia didáctica y los cambios observados en la motivación y el aprendizaje de los estudiantes. Para la recolección de información se aplican encuestas, cuestionarios y observación directa, lo que posibilita obtener datos tanto numéricos como descriptivos.

Desde la perspectiva estadística, se recurre al análisis cuantitativo de los datos obtenidos mediante instrumentos estructurados, lo que permite organizar, interpretar y comparar los resultados antes y después de la intervención, fortaleciendo la validez de los hallazgos.

En cuanto al aspecto temporal, la investigación adopta un diseño transversal, ya que la recolección de información se realiza en un periodo específico del año lectivo, sin seguimiento prolongado. Este diseño facilita una visión clara de las condiciones motivacionales y del nivel de aprendizaje durante la aplicación de la estrategia didáctica.

La integración de estos métodos contribuye al rigor científico del estudio, permitiendo comprender de manera objetiva y contextualizada el impacto de la estrategia didáctica en el proceso de enseñanza-aprendizaje de la programación.

2.7 Instrumentos de la Investigación

Para la recolección de la información, se emplearon instrumentos coherentes con los objetivos del estudio y con el enfoque metodológico previamente declarado. Estos instrumentos permitieron obtener datos relevantes sobre las variables de investigación, específicamente la motivación hacia el aprendizaje de la programación y la comprensión de los conceptos básicos de programación en los estudiantes de octavo grado de la asignatura de Computación de la Escuela Particular de Educación Básica “Pablo Palacio”.

En primer lugar, se aplicó una encuesta estructurada con escala tipo Likert, diseñada para medir el nivel de motivación de los estudiantes frente al aprendizaje de la programación. Este instrumento permitió recoger datos cuantitativos relacionados con el interés, la participación, la percepción de utilidad de la asignatura y la autoconfianza en la resolución de actividades de



programación. La aplicación de la encuesta se realizó antes y después de la intervención, lo que facilitó la comparación de los resultados y la identificación de posibles variaciones en la motivación estudiantil.

Como complemento, se utilizó una guía de observación directa, orientada a registrar comportamientos y actitudes de los estudiantes durante el desarrollo de las actividades de programación. Este instrumento permitió identificar el nivel de participación, el trabajo colaborativo, la atención en clase y la interacción con la herramienta Scratch, aportando información cualitativa que enriqueció el análisis de los resultados y fortaleció la interpretación del impacto de la estrategia didáctica implementada.

Finalmente, se empleó una rúbrica de evaluación para valorar los proyectos desarrollados por los estudiantes en Scratch. Esta herramienta permitió medir el nivel de logro alcanzado en aspectos como la creatividad, la lógica de programación, la funcionalidad y la presentación del proyecto. La rúbrica constituyó un elemento esencial para obtener una evaluación objetiva y coherente con los criterios establecidos en la investigación.

En conjunto, estos instrumentos aseguraron la triangulación de los datos, fortaleciendo la validez y confiabilidad de los resultados obtenidos, al permitir contrastar la información cuantitativa con la evidencia cualitativa derivada de la observación y el desempeño práctico de los estudiantes.

2.8 Delimitación de la Población Y Muestra

En la institución "Pablo Palacio", la delimitación de la población y la muestra es crucial para asegurar que los resultados sean representativos y útiles.

Según Toledo, (2023) considera que. “La población de una investigación está compuesta por todos los elementos (personas, objetos, organismos, historias clínicas) que participan del fenómeno que fue definido y delimitado en el análisis del problema de investigación”.

Aquí se presenta una propuesta de delimitación:

2.8.1 Delimitación de la Población

Población Total

Número de Estudiantes: 30 alumnos en total.

Nivel Educativo: Estudiantes de octavo año de educación básica.





Ubicación: Institución educativa "Pablo Palacio", Guayaquil, cantón Guayas Dirección segundo callejón 25 entre la P y N, en el suburbio suroeste de Guayaquil.

Delimitación de la Muestra

Muestra Seleccionada

Número de Estudiantes: 30 estudiantes de octavo año (aproximadamente el 10% de la población total).

2.9 Descripción de las Etapas de la Investigación

2.9.1 Etapa de estudio teórico

La etapa del estudio teórico de esta investigación se centró en la revisión y análisis de los fundamentos conceptuales relacionados con la motivación académica y el aprendizaje de la programación en estudiantes de educación básica, con el propósito de comprender los factores que influyen en el interés y la participación de los estudiantes de octavo grado. Para ello, se revisaron teorías contemporáneas sobre motivación, como la teoría de la autodeterminación y los enfoques socioeducativos, además de estudios previos que abordan la relación entre actividades tecnológicas y el compromiso estudiantil. Esta revisión permitió establecer los criterios conceptuales que orientan la interpretación de la motivación en el contexto escolar y fundamentan la pertinencia de analizar su vínculo con actividades prácticas de programación.

Asimismo, el estudio teórico incluyó la revisión de investigaciones sobre el uso de Scratch como herramienta didáctica, considerando su potencial para promover el pensamiento lógico, la creatividad y la participación activa en el aprendizaje. Se analizaron experiencias educativas que demuestran cómo la programación visual favorece el aprendizaje significativo y puede incrementar la motivación en entornos escolares. Esta etapa permitió identificar los aportes y limitaciones reportados por la literatura, lo que contribuyó a definir el enfoque explicativo del estudio, a seleccionar los instrumentos adecuados y a estructurar la intervención pedagógica. En conjunto, el análisis teórico proporcionó una base científica sólida que orienta el desarrollo metodológico y la interpretación de los resultados obtenidos.

El papel de las emociones de logro. De Wen, Wu, & Hsu, (2023), reportaron que una intervención con Scratch incrementó significativamente la motivación y el rendimiento académico de los participantes. Tal hallazgo sugiere que la motivación puede fortalecerse mediante métodos pedagógicos que activan emociones de logro y autopercebidas competencias. Asimismo, los componentes motivacionales identificados en el modelo MUSIC empoderamiento, utilidad, éxito, interés y cuidado se han mostrado como predictores

significativos del éxito en el aprendizaje de programación. Examinando la relación entre los componentes del modelo MUSIC de motivación y el rendimiento estudiantil en programación informática (Alshammari, 2023), se encontró que los estudiantes que percibieron mayor utilidad e interés mostraron mejores logros académicos. Esto permite interpretar que la baja motivación detectada en el pretest puede deberse a que los estudiantes no perciben la relevancia de la programación, ni sienten que tengan control o éxito en sus actividades.

Finalmente, los estudios de revisión sistemática también sostienen que el uso de entornos gamificados o visuales en programación favorece la motivación y la participación del estudiante. En la revisión Gamificación con Scratch o App Inventor en la educación superior: una revisión sistemática (Pérez & Martínez, 2022), se concluye que plataformas como Scratch aumentan competencias de aprendizaje, incluyendo la motivación, aunque su uso aún es limitado en etapas tempranas. Esto aporta un sustento para la intervención didáctica con Scratch en tu propuesta.

Con base en estos marcos teóricos, la baja motivación registrada en el pretest puede interpretarse como resultado de una combinación de factores: falta de percepción de relevancia de la programación, escasa oportunidad de experimentar éxito temprano, y método de enseñanza poco interactivo. Estas condiciones justifican el diseño de una estrategia didáctica orientada a la motivación, el sentido de competencia y la participación activa. En consecuencia, la intervención basada en Scratch se considera adecuada para revertir dichas deficiencias al ofrecer un entorno visual, lúdico y colaborativo que promueve la motivación intrínseca, la autonomía y el interés por aprender programación.

2.9.2 Etapa de diagnóstico inicial

El diagnóstico inicial realizado con los 30 estudiantes de octavo grado de la Escuela Pablo Palacio, permitió identificar el estado real de la motivación hacia la programación antes de la intervención pedagógica con Scratch. A través de encuestas y cuestionarios tipo Likert se evidenció que una parte significativa de los estudiantes presentaba bajos niveles de motivación, manifestados en desinterés, poca autoconfianza para resolver problemas y una percepción limitada del valor académico de la programación. Los datos obtenidos reflejaron, además, que la mitad de los estudiantes consideraba que las evaluaciones y retroalimentaciones recibidas en el área de tecnología eran poco efectivas, mientras que el otro 50% afirmaba sentirse satisfecho con dichos procesos, lo que muestra una división en las percepciones sobre el acompañamiento docente inicial.



Este diagnóstico se complementó con la observación directa, la cual permitió identificar comportamientos como la escasa participación activa, la dependencia de instrucciones directas y la baja iniciativa para explorar herramientas digitales por cuenta propia. Asimismo, algunos estudiantes manifestaron la percepción de que las actividades académicas carecían de suficiente flexibilidad, lo que limitaba su creatividad y autonomía en el desarrollo de tareas. A pesar de

estas dificultades, el diagnóstico también reveló que los estudiantes valoraban positivamente el apoyo y acompañamiento docente, considerándolo un factor motivador cuando las actividades resultaban desafiantes. En conjunto, estos hallazgos justificaron la necesidad de implementar una propuesta basada en programación visual con Scratch, orientada a incrementar la motivación, fortalecer la participación y promover un aprendizaje más dinámico y significativo.

2.9.3 Etapa de la modelación de la propuesta

La modelación de la propuesta se estructuró a partir del diagnóstico inicial, el cual evidenció bajos niveles de motivación hacia la programación en los estudiantes de octavo grado de la Escuela Pablo Palacio. Con base en estos hallazgos, se diseñó una intervención pedagógica fundamentada en el uso de Scratch, con el propósito de promover un aprendizaje más dinámico, participativo y significativo. El modelo propuesto integra actividades secuenciales que combinan exploración guiada, resolución de problemas y creación de proyectos digitales, permitiendo que los estudiantes desarrollen competencias básicas de programación mediante experiencias prácticas y contextualizadas. Esta estructura responde directamente a la necesidad de incrementar la motivación, fortalecer la autoconfianza y fomentar la participación activa observada como limitada en el diagnóstico inicial.

La propuesta se modeló bajo el enfoque explicativo y el método mixto, lo que permitió articular estrategias pedagógicas con mecanismos de evaluación cuantitativa y cualitativa. En el modelo, cada actividad fue diseñada para observar y medir cambios en la motivación mediante encuestas tipo Likert, instrumentos de observación directa y rúbricas de evaluación de proyectos. Asimismo, la intervención se organizó en fases que avanzan desde la familiarización con los bloques de programación hasta el desarrollo de proyectos creativos y funcionales, favoreciendo el aprendizaje progresivo. Esta secuenciación facilita comprender cómo las actividades de programación visual pueden generar efectos sobre la motivación, permitiendo analizar el proceso a nivel explicativo. En conjunto, la modelación de la propuesta establece un marco operativo claro, coherente y fundamentado, orientado a transformar la experiencia de aprendizaje de los estudiantes a través del uso pedagógico de Scratch.

2.9.4 Fase del diseño

El objetivo principal de esta fase fue estructurar una secuencia de actividades didácticas con Scratch que permitiera a los estudiantes familiarizarse con el entorno, comprender conceptos básicos de programación y aplicar sus conocimientos en proyectos prácticos, promoviendo la motivación, la creatividad y la colaboración.

La propuesta se organizó en cuatro sesiones secuenciales, cada una con un enfoque pedagógico

específico:

Sesión 1: Introducción a Scratch

Objetivo: Conocer qué es Scratch y comprender su importancia como herramienta educativa para el aprendizaje de programación.

- Actividades: Presentación teórica sobre Scratch, ejemplos de proyectos educativos y explicación de las ventajas del entorno visual para el desarrollo del pensamiento computacional.
- Recursos: Computadoras, proyector, ejemplos de proyectos Scratch.

Sesión 2: Familiarización con el entorno de Scratch

- Objetivo: Identificar y manejar los elementos básicos del entorno de Scratch, incluyendo personajes (sprites), escenario, bloques de código y menús principales.
- Actividades: Exploración guiada del entorno, reconocimiento de bloques de movimiento, eventos y apariencia, y realización de pequeños ejercicios de prueba para familiarizarse con la interfaz.
- Recursos: Computadoras con Scratch instalado, guía de referencia rápida y actividades de prueba paso a paso.

Sesión 3: Ejercicio práctico de movimiento de personajes

- Objetivo: Aplicar bloques básicos de programación para mover un personaje en el escenario.
- Actividades: Crear secuencias de bloques que permitan al personaje desplazarse, realizar giros y responder a eventos de teclado. Los estudiantes realizan sus propios experimentos, fomentando la autonomía y la comprensión de conceptos de secuencia y control de eventos.
- Recursos: Computadoras, guías impresas de ejercicios, proyector para mostrar ejemplos.

Sesión 4: Creación de un videojuego interactivo

- Objetivo: Integrar los conocimientos adquiridos en un proyecto creativo y motivador: un videojuego donde el jugador debe defenderse de un murciélago.

- **Actividades:** Diseñar y programar el videojuego, utilizando bloques de movimiento, control y eventos. Los estudiantes aplican la secuencia de aprendizaje completa: planificación, programación, pruebas y ajustes, promoviendo la creatividad, la resolución de problemas y la motivación.
- **Recursos:** Computadoras, Scratch, guía de pasos para la creación del videojuego, proyector y espacio para la exposición de los proyectos finales.

La secuencia de las sesiones fue diseñada siguiendo un enfoque progresivo y constructivista, en el que cada actividad se construye sobre los conocimientos adquiridos en la sesión anterior. La estructura garantiza que los estudiantes no solo aprendan a manejar Scratch, sino que también desarrollen habilidades de pensamiento computacional, creatividad y colaboración. Además, el diseño incorpora elementos motivadores y prácticos, como la creación de un videojuego final, que permite a los estudiantes experimentar la satisfacción de aplicar lo aprendido en un proyecto tangible, aumentando su interés y participación en el proceso de aprendizaje.

La fase de diseño permitió estructurar una estrategia didáctica coherente, progresiva y motivadora, basada en la familiarización con Scratch, la práctica guiada y la creación de proyectos creativos. Este diseño asegura que los estudiantes puedan integrar conocimientos teóricos y prácticos, desarrollando competencias de programación mientras se fomenta la motivación, la autonomía y la participación activa en el aula.

2.9.4.1 Aplicación Práctica en Octavo Año

2.9.4.1.2 Aprendizaje Basado en Proyectos

Mi primer proyecto: “Crea tu primer videojuego de aventuras en Scratch”.

- **Producto:** un videojuego sencillo en Scratch donde un personaje (principal) debe superar obstáculos para llegar a una meta, utilizando condicionales, bucles y variables de puntaje.
- **Pregunta guía:** ¿Cómo podemos usar la programación para dar instrucciones a un computador y crear un videojuego interactivo?
- **Fases:**
 1. **Exploración inicial:** Introducción al concepto de algoritmo y a los bloques básicos de Scratch (movimiento, eventos, apariencia).
 2. **Diseño en papel:** Storyboard del juego (personajes, escenario, reglas).

3. **Prototipo:** Construcción en Scratch del escenario inicial y movimiento del personaje principal.
4. **Desarrollo progresivo:** Implementación de obstáculos, bucles para movimientos repetitivos, condicionales (si toca objeto entonces pierde vida), y variables para puntaje o vidas.
5. **Retroalimentación:** Cada grupo prueba el juego de otro equipo y ofrece sugerencias de mejora.
6. **Publicación y presentación:** Compartir el juego realizado en Scratch y presentar el funcionamiento ante la clase.

En cada fase, el docente facilita recursos y una lista de puntos que necesita para poder emplear todo lo que necesita para poder crear su primer juego en Scratch como: una lista de requisitos mínimos, tareas sencillas que rápidamente lo pueden realizar, ejemplos de “personaje principal o scripts”, procurando siempre el equilibrio entre autonomía y guía, conforme avanza el desarrollo del proyecto

2.9.4.2 Rúbrica y Evaluación Formativa

Las rúbricas aportan criterios transparentes sobre calidad técnica y didáctica del proyecto como el uso de variables, estructuras de control, paralelismo, interacción, originalidad, claridad del propósito. Estas especifican niveles de desempeño y facilitan la coevaluación y autoevaluación. Según Torres & Perera (2010) considera que “el cuadro es una herramienta de evaluación basada en una escala cuantitativa o cualitativa relacionada con criterios predeterminados que miden las actividades de los alumnos en los aspectos evaluados de la tarea o actividad” (p. 142).

Los estudiantes realizan mini retos con tarjetas de bloques de Scratch, como: mover un sprite, usar bucles para repetir un movimiento, aplicar condicionales como “si toca color rojo entonces se detiene”. Un micro proyecto semilla, por ejemplo: que dentro del mini juego, el personaje deba esquivar un objeto que cae del cielo. El objetivo es que comprendan la relación entre algoritmos simples y su representación mediante bloques. Se puede realizarlo de las siguientes formas

1. **Prototipo con plantilla:**

- Los estudiantes reciben un proyecto base con sprites y escenarios predefinidos (personaje principal, fondo y objetos).

- La tarea consiste en completar la lógica del juego, agregando:
 - Variables de puntaje y vidas.
 - Colisiones que afecten el resultado (si toca obstáculo → pierde vida).
 - Bucles y condicionales que generen interacción continua.

Permite a los estudiantes aplicar estructuras básicas de control y variables en un entorno ya estructurado.

2. Proyecto abierto:

- Cada grupo diseña y desarrolla su propio videojuego o animación relacionado con la introducción a la programación.
- El docente orienta mediante preguntas guía:
 - ¿Qué objetivo tendrá tu juego?
 - ¿Cómo se usan bucles o condicionales para lograr la mecánica?
 - ¿Qué variables representan el progreso (vidas, niveles, tiempo)?

Al realizarlo de esta forma fomenta la autonomía, la creatividad y la transferencia de lo aprendido a un producto original.

3. Revisión y mejora:

- Los equipos comparten su proyecto en un “studio” de Scratch del curso.
- Se realiza coevaluación entre pares con base en una rúbrica común.
- Se aplica la herramienta Dr. Scratch para obtener un reporte del nivel de pensamiento computacional alcanzado.
- Los estudiantes realizan mejoras antes de la entrega final, documentando cambios y aprendizajes.

Permite reflexionar sobre el proceso, identificar errores “bugs” y reforzar la calidad del producto, además que realizar retroalimentación constante, mejorando sus conocimientos pre adquiridos, y empiezan a obtener un interés o motivación hacia la programación, también aprenden a trabajar en equipo.

RÚBRICA

Claridad del objetivo del proyecto	Excelente	El propósito del juego/animación está claramente definido y es coherente.
	Aceptable	El propósito se entiende, pero con poca claridad o coherencia.
	Insuficiente	No se identifica un propósito educativo o narrativo claro.
Corrección lógica del código	Excelente	El proyecto funciona sin errores significativos; la lógica es consistente.
	Aceptable	Presenta algunos errores menores que no afectan la experiencia.
	Insuficiente	Contiene errores graves (“bugs”) que impiden jugar o interactuar.
Uso de eventos y paralelismo	Excelente	Se emplean múltiples eventos y acciones en paralelo para enriquecer la interactividad.
	Aceptable	Se utilizan algunos eventos básicos, pero sin paralelismo.
	Insuficiente	Uso muy limitado o inexistente de eventos.
Integración de variables y listas	Excelente	Incluye puntajes, vidas u otras variables, con lógica clara y funcional.
	Aceptable	Usa solo una variable básica.
	Insuficiente	No incorpora variables ni listas.
Diseño visual y usabilidad	Excelente	Presenta sprites y escenarios originales/attractivos, con instrucciones claras y navegación sencilla.
	Aceptable	Presenta diseño funcional pero limitado o poco claro.
	Insuficiente	El diseño es confuso o poco cuidado.
Documentación del proyecto	Excelente	Incluye comentarios en el código, pantalla de instrucciones y créditos claros.
	Aceptable	Contiene instrucciones parciales o poco detalladas.
	Insuficiente	No hay documentación ni instrucciones visibles.
Trabajo en equipo y roles	Excelente	Evidencias claras de colaboración y reparto de roles (diseñador, programador, tester, documentador).
	Aceptable	Se observa participación desigual entre integrantes.
	Insuficiente	No hay evidencias de colaboración organizada.

Tabla 4 Rúbrica

2.9.5 Etapa del diagnóstico final

La modelación de la propuesta se estructuró a partir del diagnóstico inicial, el cual evidenció bajos niveles de motivación hacia la programación en los estudiantes de octavo grado de la Escuela Pablo Palacio. Con base en estos hallazgos, se diseñó una intervención pedagógica fundamentada en el uso de Scratch, con el propósito de promover un aprendizaje más dinámico, participativo y significativo. El modelo propuesto integra actividades secuenciales que combinan exploración guiada, resolución de problemas y creación de proyectos digitales, permitiendo que los estudiantes desarrollen competencias básicas de programación mediante experiencias prácticas y contextualizadas. Esta estructura responde directamente a la necesidad de incrementar la motivación, fortalecer la autoconfianza y fomentar la participación activa observada como limitada en el diagnóstico inicial.

La propuesta se modeló bajo el enfoque explicativo y el método mixto, lo que permitió articular estrategias pedagógicas con mecanismos de evaluación cuantitativa y cualitativa. En el modelo, cada actividad fue diseñada para observar y medir cambios en la motivación mediante encuestas tipo Likert, instrumentos de observación directa y rúbricas de evaluación de proyectos. Asimismo, la intervención se organizó en fases que avanzan desde la familiarización con los bloques de programación hasta el desarrollo de proyectos creativos y funcionales, favoreciendo el aprendizaje progresivo. Esta secuenciación facilita comprender cómo las actividades de programación visual pueden generar efectos sobre la motivación, permitiendo analizar el proceso a nivel explicativo. En conjunto, la modelación de la propuesta establece un marco operativo claro, coherente y fundamentado, orientado a transformar la experiencia de aprendizaje de los estudiantes a través del uso pedagógico de Scratch.

2.10 Presentación de los resultados del estudio diagnóstico

En esta etapa se dio una visión precisa de la forma en que se debe evolucionar la propuesta, basado en los datos obtenidos de los estudiantes. Utilizando un formato en Google. <https://forms.gle/tPbU76vDcMosW51a8> A continuación, los siguientes análisis de datos explorados.

Te sientes motivado por aprender programación

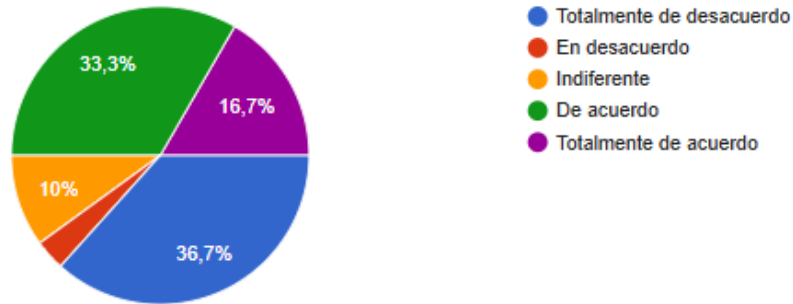


Figura 1 Te sientes motivado por aprender programación

El 50% de estudiantes indican que no se encuentran motivados al aprender programación y un 10% se muestra indiferente. Esto a punto que hay una mayoría de estudiantes que no están motivados o se sienten indiferentes del aprendizaje de programación.

¿Crees que tus profesores tienen claro el contenido a enseñar?

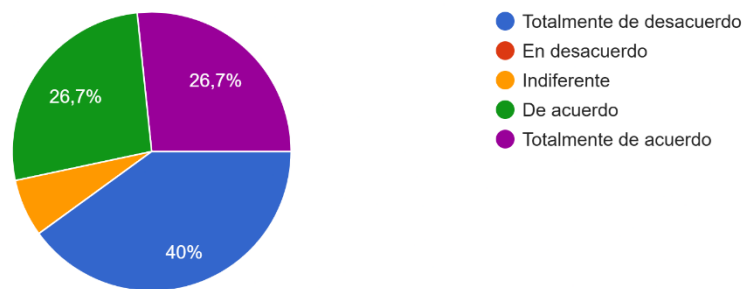


Figura 2 ¿Crees que tus profesores tienen claro el contenido a enseñar?

El 66,7 de estudiantes consideran que los docentes no tienen claro el contenido a enseñar. Esto indica que los estudiantes pueden detectar una falta de preparación por el parte del docente.

¿Te parece que las actividades de programación deben ser solamente de manera escrita?

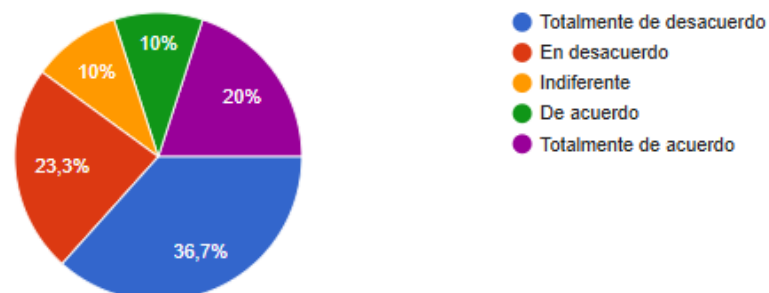


Figura 3: ¿Te parece que las actividades de programación deben ser solamente de manera escrita?

EL 60% de estudiantes consideran que las actividades de programación no deben ser solamente escrito. Esto indica una falta de innovación al momento de impartir el contenido de programación.

¿Sientes que tus profesores te apoyan cuando tienes dificultades al programar?

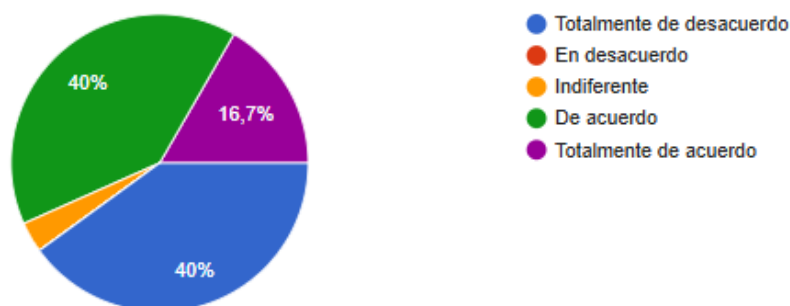


Figura 4: ¿Sientes que tus profesores te apoyan cuando tienes dificultades al programar?

Un 56,7 de estudiantes considera que los docentes lo apoyan cuando tienen dificultades al momento de programar. Esto indica un claro apoyo por parte del docente con sus estudiantes.

¿Crees que la manera en la cual estás aprendiendo te será útil en el futuro?

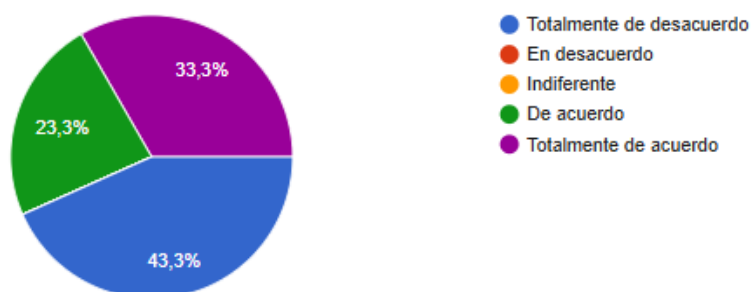


Figura 5: ¿Crees que la manera en la cual estás aprendiendo te será útil en el futuro?

El 56,6 % de los estudiantes indican que el contenido que están aprendiendo es útil para el futuro. Esto indica que los estudiantes valoran de forma correcta el contenido aprendido.

¿Puedes elegir cómo quieres trabajar tus proyectos de programación?

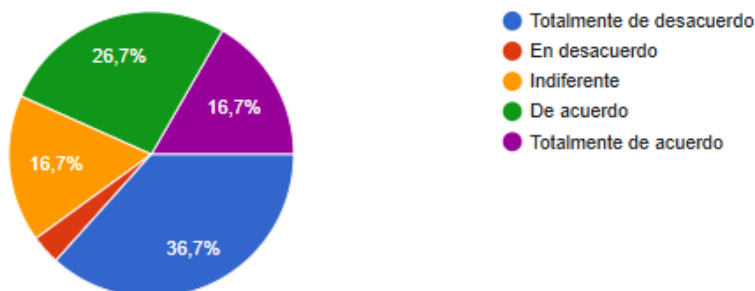


Figura 6: ¿Puedes elegir cómo quieres trabajar tus proyectos de programación?

El 53,4 % de estudiantes indican que no pueden elegir como quiere trabajar en sus proyectos de programación. Esto indica que no hay flexibilidad en la elección del modo de trabajo en los proyectos.

¿Tienes oportunidad de elegir temas o proyectos que te interesen?

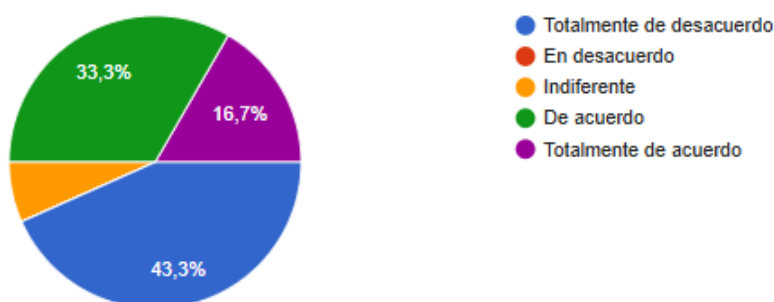


Figura 7: ¿Tienes oportunidad de elegir temas o proyectos que te interesen?

El 50% de estudiantes indican que pueden elegir sus temas o proyectos. Lo que indican flexibilidad en la elección del tema, pero no en el modo de trabajo.

¿Puedes trabajar a tu propio ritmo en tus proyectos de programación?



Figura 8: ¿Puedes trabajar a tu propio ritmo en tus proyectos de programación?

El 50% de estudiantes indican que pueden trabajar a su propio ritmo en los proyectos de programación. Esto indica la flexibilidad en el ritmo de trabajo.

¿Tienes la libertad de probar diferentes enfoques en tus proyectos de programación?

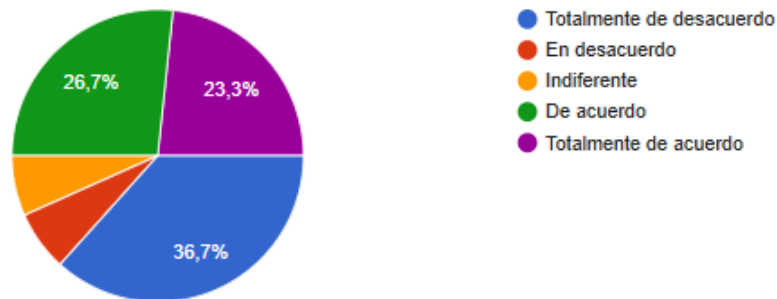


Figura 9 ¿Tienes la libertad de probar diferentes enfoques en tus proyectos de programación?

El 59% de estudiantes indican que pueden probar diferentes enfoques en tus proyectos de programación. Esto indica que hay flexibilidad al momento de dar un enfoque diferente en un proyecto.

¿Consideras que tus profesores están dispuestos a adaptar sus planes de enseñanzas a tus necesidades?

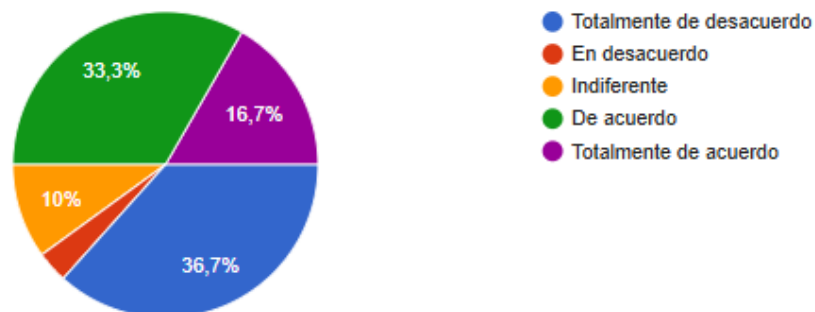


Figura 10 ¿Consideras que tus profesores están dispuestos a adaptar sus planes de enseñanzas a tus necesidades?

EL 50% de estudiantes están de acuerdo y el otro 50% están de desacuerdo. Esto indica que hay una clara división en la percepción sobre como el profesor adapta sus planes a las necesidades del estudiantado.

¿Crees que los proyectos de programación están orientados con tus intereses y hobbies?

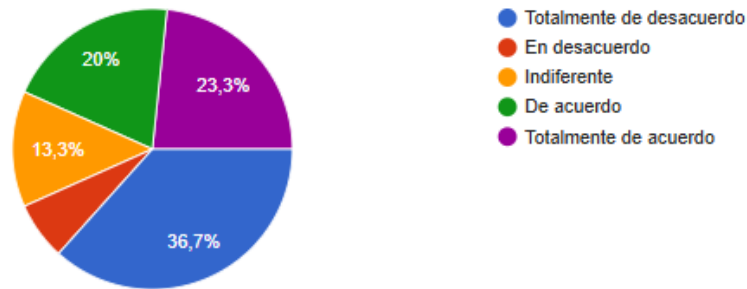


Figura 11 ¿Crees que los proyectos de programación están orientados con tus intereses y hobbies?

El 56,7 de estudiante indican que los proyectos de programación no están orientados con sus intereses y hobbies. Esto indica que los proyectos deben ser orientadas a los intereses de los estudiantes para que puedan identificarse en mejor en sus trabajos.

¿Te parece que los ejemplos utilizados en clases son relevantes en la vida diaria?

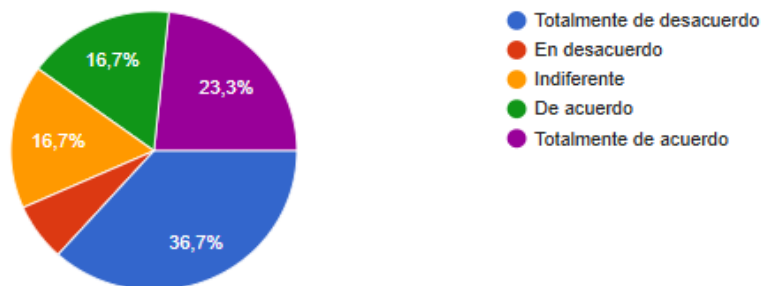


Figura 12 ¿Te parece que los ejemplos utilizados en clases son relevantes en la vida diaria?

El 60,1% de los estudiantes indican que los ejemplos utilizados en clases no son relevantes en la vida diaria. Esto indica que al momento de utilizar ejemplos se alejan de la cotidianidad del estudiante lo cual no permite que el estudiante se identifique con su proyecto.

¿Crees que la programación te ayudará a resolver problemas o situaciones que enfrentas a diario?

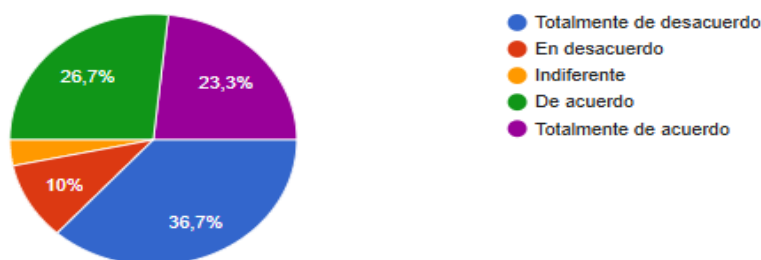


Figura 13 ¿Crees que la programación te ayudará a resolver problemas o situaciones que enfrentas a diario?

El 53,3% de estudiantes están de acuerdo con que la programación los ayudará a resolver problemas o situaciones que enfrentan a diario. Esto indica la importancia que los estudiantes tienen sobre el tema.

¿Crees que los proyectos de programación permiten aplicar tus conocimientos y habilidades en un contexto realista?

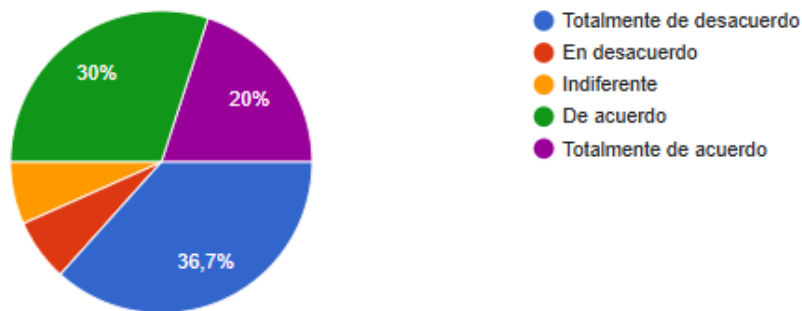


Figura 14 ¿Crees que la programación te ayudará a resolver problemas o situaciones que enfrentas a diario?

Se puede notar una clara división en el grupo de los estudiantes. Esto indica una clara división de opiniones al momento de aplicar su proyecto de manera realista.

¿Crees que las evaluaciones y retroalimentaciones que recibes en clases de programación te ayuda a mejorar tus habilidades?

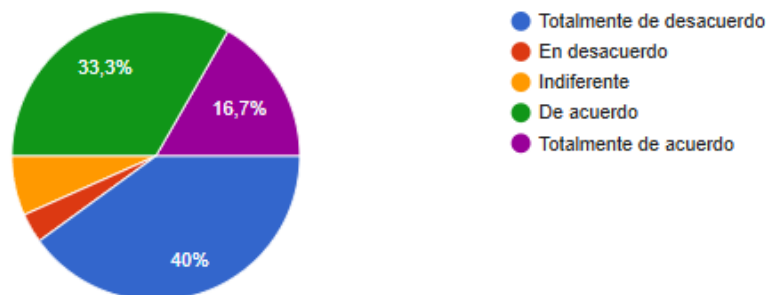


Figura 15 ¿Crees que las evaluaciones y retroalimentaciones que recibes en clases de programación te ayuda a mejorar tus habilidades?

El 50% de los estudiantes indican que están de acuerdo en que las evaluaciones y retroalimentaciones ayudan a mejorar sus habilidades mientras el otro 50% no esto de acuerdo. Esto indica que debe haber una mejora en las evaluaciones y retroalimentaciones.

¿Te parece que las evaluaciones de tus proyectos son justas y objetivas?

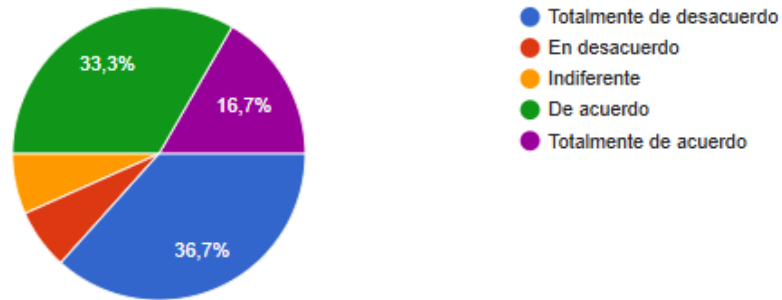


Figura 16 ¿Te parece que las evaluaciones de tus proyectos son justas y objetivas?

El 50% de los estudiantes indican que están de acuerdo que las evaluaciones son justas y objetivas mientras el otro 50% están en desacuerdo. Esto refuerza que las evaluaciones necesitan una mejor para que los estudiantes se sientan más conformes con las evaluaciones futuras.

¿Crees que las retroalimentaciones que recibes de tus profesores te ayudan a entender que debes mejorar en tus proyectos de programación?

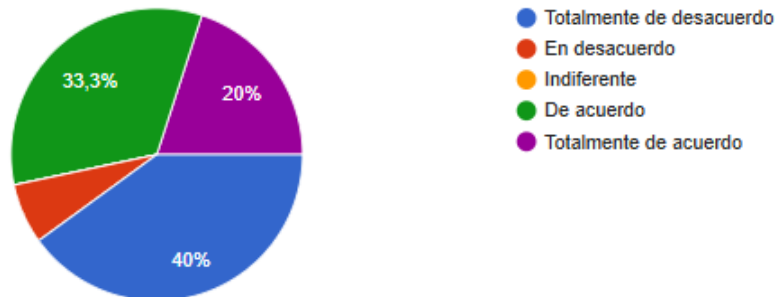


Figura 17 ¿Crees que las retroalimentaciones que recibes de tus profesores te ayudan a entender que debes mejorar en tus proyectos de programación?

El 53,3% de estudiantes indican que están de acuerdo con las retroalimentaciones de sus profesores, aunque un 46,7% indican lo contrario. Esto indica que se necesita mejorar las retroalimentaciones por parte de los docentes.

¿Te sientes motivado para mejorar tus habilidades en programación debido a las evaluaciones y retroalimentaciones que recibes?

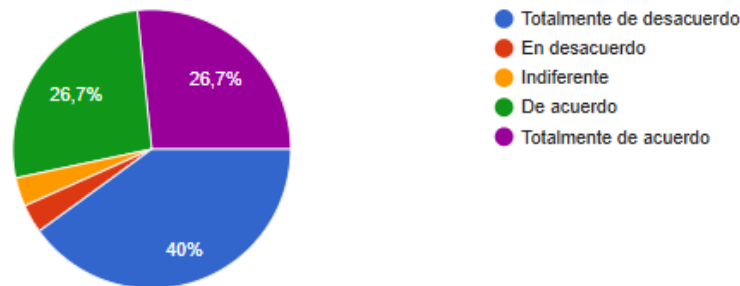


Figura 18 ¿Te sientes motivado para mejorar tus habilidades en programación debido a las evaluaciones y retroalimentaciones que recibes?

El 53,3 % de estudiantes indican que se encuentran motivado para mejorar sus habilidades debido a las evaluaciones y retroalimentaciones, aunque un 46,7% indican lo contraria. Esto indica que un gran grupo de estudiantes no se encuentran motivados y se necesita una mejoría en las retroalimentaciones.

¿Crees que las evaluaciones de tus proyectos de programación se reflejan con precisión tus habilidades y conocimientos en programación?

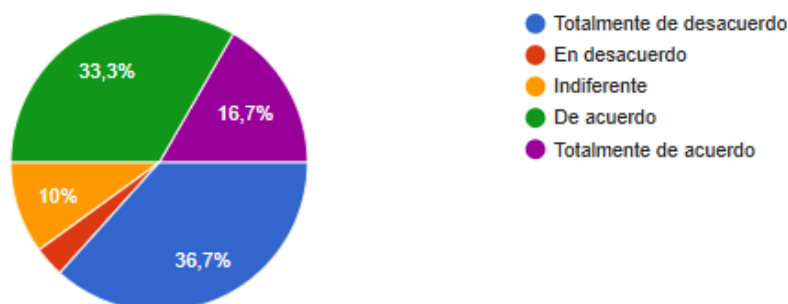


Figura 19 ¿Crees que las evaluaciones de tus proyectos de programación se reflejan con precisión tus habilidades y conocimientos en programación?

EL 50% de estudiantes indican que se encuentran de acuerdo con las evaluaciones de sus proyectos, aunque hay un 50% que indica lo contrario. Esto indica que se necesita una mejoría en las evaluaciones para que se reflejen de una manera idónea las habilidades y conocimiento en programación por parte de los estudiantes.

2.10.1 Resultados

Los resultados obtenidos en la fase diagnóstica permitieron identificar diversos factores que inciden en el proceso de enseñanza-aprendizaje de la programación en los estudiantes de octavo grado. A través de la aplicación del pretest estructurado en una escala tipo Likert, se analizaron las percepciones y actitudes de los participantes en torno a la motivación, la evaluación, la retroalimentación y el acompañamiento docente.

En primer lugar, se evidenció una baja motivación general hacia el aprendizaje de la programación. Una proporción significativa de los estudiantes manifestó falta de interés, escasa participación activa y poca disposición para explorar contenidos nuevos. Estos resultados reflejan una tendencia hacia la desmotivación intrínseca, que puede deberse a la percepción de dificultad de la materia, la ausencia de metodologías dinámicas y la falta de relación entre los contenidos y los intereses del alumnado. Este hallazgo coincide con lo expuesto en la etapa teórica, donde se sustenta que la motivación depende del sentido de competencia, autonomía y conexión social que experimentan los estudiantes durante su proceso formativo.

En cuanto a las evaluaciones y retroalimentaciones aplicadas por los docentes, los resultados muestran una división equilibrada del 50 % entre los estudiantes que consideran que estos procesos son efectivos y aquellos que perciben que no contribuyen de manera significativa a su aprendizaje. Este dato revela la necesidad de revisar las estrategias evaluativas y las formas de retroalimentación empleadas, pues si bien una parte del alumnado valora la orientación docente, otra parte siente que los mecanismos actuales no reflejan adecuadamente su progreso ni fortalecen su comprensión de los contenidos.

Por otra parte, se observó que una proporción considerable de los participantes considera que falta flexibilidad en ciertas áreas del proceso de enseñanza. Esto sugiere que las metodologías utilizadas podrían estar limitando la autonomía y la creatividad de los estudiantes, generando una experiencia de aprendizaje rígida y poco adaptada a los diferentes ritmos y estilos de aprendizaje presentes en el aula. Tal situación refuerza la necesidad de incorporar estrategias pedagógicas que promuevan aprendizajes más abiertos, exploratorios y centrados en el estudiante, favoreciendo la autoexpresión y el pensamiento crítico.

A pesar de las debilidades señaladas, los resultados también reflejan una valoración positiva del acompañamiento y apoyo de los docentes. La mayoría de los estudiantes reconoce que la intervención y orientación del profesor constituyen un elemento fundamental para su comprensión de los temas y para mantener cierto grado de motivación. Este aspecto destaca la importancia del rol docente no solo como transmisor de conocimientos, sino como facilitador

del aprendizaje y mediador emocional, especialmente en áreas percibidas como complejas.

En conjunto, los resultados del diagnóstico evidencian un escenario educativo que combina fortalezas y áreas de mejora. Mientras el acompañamiento docente se percibe positivamente, existen deficiencias en la motivación, la flexibilidad metodológica y la efectividad de las evaluaciones. Estas observaciones justifican la implementación de una estrategia didáctica innovadora basada en el uso de Scratch, orientada a incrementar la motivación, promover la participación activa y fortalecer la comprensión de los contenidos mediante experiencias de aprendizaje prácticas, creativas y colaborativas.

Conclusiones del Diagnóstico

El diagnóstico realizado permitió identificar una baja motivación general hacia el aprendizaje de la programación en los estudiantes de octavo grado. Un porcentaje significativo manifestó desinterés, indiferencia o escasa participación en las actividades, lo cual evidencia que las metodologías actuales no logran despertar el interés ni fomentar una conexión significativa con los contenidos. Esta desmotivación se relaciona con la percepción de dificultad de la asignatura, el uso limitado de estrategias innovadoras y la escasa vinculación de los contenidos con la realidad e intereses del alumnado.

Asimismo, se evidenció que una parte considerable de los estudiantes considera que las actividades de programación no deberían desarrollarse únicamente de forma escrita, lo que revela la necesidad de incorporar metodologías más prácticas, visuales e interactivas, que favorezcan el aprendizaje activo. La falta de innovación metodológica limita el desarrollo de la creatividad, la autonomía y el interés por la programación, afectando directamente la motivación intrínseca de los estudiantes.

En relación con la flexibilidad pedagógica, los resultados muestran que muchos estudiantes no pueden elegir cómo trabajar sus proyectos ni tienen total libertad para adaptar los temas a sus intereses personales. Aunque existe cierta flexibilidad en el ritmo de trabajo y en los enfoques utilizados, persisten restricciones que limitan la autonomía del alumnado. Esto sugiere la necesidad de promover estrategias centradas en el estudiante, que fomenten la toma de decisiones, la creatividad y la personalización del aprendizaje.

Respecto a las evaluaciones y retroalimentaciones, se identificó una clara división de opiniones. Mientras aproximadamente la mitad de los estudiantes considera que estos procesos contribuyen a mejorar sus habilidades, la otra mitad percibe que no reflejan adecuadamente sus conocimientos ni favorecen su aprendizaje. Esta situación evidencia la necesidad de fortalecer las prácticas evaluativas hacia enfoques más formativos, claros y orientados a la mejora

continua, que permitan a los estudiantes comprender sus avances, debilidades y oportunidades de mejora.

A pesar de las debilidades detectadas, los estudiantes valoran positivamente el acompañamiento docente, reconociendo el apoyo recibido cuando presentan dificultades en la programación. Este aspecto confirma el rol fundamental del profesor como guía, mediador y facilitador del aprendizaje, especialmente en asignaturas que requieren el desarrollo de habilidades lógicas y técnicas.

Finalmente, los resultados del diagnóstico justifican la implementación de una estrategia didáctica innovadora basada en Scratch, orientada a fortalecer la motivación, promover la participación activa, fomentar la creatividad y mejorar el aprendizaje de la programación mediante actividades prácticas, contextualizadas y significativas. Esta propuesta responde a las necesidades identificadas y busca transformar la experiencia educativa hacia un modelo más dinámico, flexible y centrado en el estudiante.

CAPÍTULO III: PRESENTACIÓN Y VALIDACIÓN DE LA PROPUESTA

El capítulo III titulado presentación y validación de la propuesta, tiene como propósito presentar y desarrollar la propuesta de la estrategia didáctica basada en cápsulas educativas para la enseñanza de la programación mediante Scratch. En este capítulo se describe la estructura, los recursos, las sesiones y el proceso de implementación y evaluación de la propuesta. Su diseño se sustenta en los antecedentes investigativos abordados en el Capítulo I y en los fundamentos teóricos y metodológicos establecidos en el Capítulo II, permitiendo evidenciar la aplicación práctica de la investigación y la validación de la propuesta en el contexto educativo estudiado.

3.1 Propuesta

La propuesta didáctica surge como respuesta directa a los resultados obtenidos en la fase diagnóstica de la investigación, los cuales evidenciaron bajos niveles de motivación y dificultades en la comprensión de conceptos básicos de programación en los estudiantes de octavo grado de la asignatura de Computación de la Escuela Particular de Educación Básica “Pablo Palacio”. Dichas dificultades se relacionan con el uso predominante de metodologías tradicionales, poco interactivas, que no favorecen la participación activa ni el aprendizaje significativo en un contexto educativo socialmente vulnerable.

En este sentido, la propuesta se orienta al diseño e implementación de una estrategia didáctica basada en Scratch, como entorno de programación visual, con el propósito de fortalecer el aprendizaje de la programación mediante actividades prácticas, lúdicas y progresivas. Scratch permite a los estudiantes interactuar con conceptos fundamentales de programación de manera visual e intuitiva, facilitando la comprensión de estructuras básicas como secuencias, eventos, condicionales y ciclos, al tiempo que promueve la creatividad y el pensamiento lógico.

La propuesta contempla el desarrollo de cuatro sesiones pedagógicas, estructuradas de forma secuencial, que combinan actividades teóricas y prácticas orientadas a la resolución de problemas y al trabajo colaborativo. Cada sesión se diseña considerando el nivel previo de los estudiantes, el acompañamiento docente y el uso del laboratorio de computación institucional, con acceso a internet y recursos tecnológicos disponibles. Esta organización busca generar un ambiente de aprendizaje activo, donde los estudiantes participen de manera constante y significativa.

Finalmente, la propuesta pretende contribuir al incremento de la motivación hacia la asignatura de Computación y al desarrollo de habilidades básicas de programación, pensamiento computacional y trabajo en equipo. Su aplicación se concibe como una alternativa pedagógica viable y contextualizada, que responde a las necesidades educativas detectadas y que puede ser

replicada o adaptada en contextos escolares similares.

3.2 Fundamentación

Diseñar una estrategia didáctica basada en cápsulas educativas con el programa Scratch responde a la necesidad de motivar a los estudiantes de octavo grado en el aprendizaje de la programación. Las cápsulas educativas, al ser breves y dinámicas capturan la atención de los alumnos, facilitando la comprensión de conceptos complejos a través de un enfoque visual e interactivo.

Scratch permite a los estudiantes crear sus propios proyectos, mejorando su creatividad y fomentando un aprendizaje activo. Esta herramienta promueve un ambiente que estimula la curiosidad y el deseo de explorar, haciendo que la programación se perciba como una actividad entretenida y accesible. Además, las cápsulas educativas pueden ser revisadas a su propio ritmo, lo que favorece la autonomía y el aprendizaje individualizado.

El trabajo colaborativo en la creación de estas cápsulas fortalece no solo las habilidades técnicas, sino también las competencias interpersonales, como la comunicación y el trabajo en equipo. En conjunto, esta estrategia busca no solo enseñar programación, sino también cultivar un entorno de aprendizaje motivador que preparemos a los alumnos para un futuro digital, desarrollando en ellos habilidades técnicas como un pensamiento crítico y creativo.

3.3 Objetivos Generales y Específicos

3.3.1 Objetivos Generales

- Fomentar el interés y la motivación de los estudiantes de octavo grado hacia la programación mediante el uso de cápsulas educativas interactivas.

3.3.2 Objetivos Específicos

- Diseñar un conjunto de cápsulas educativas que abordan conceptos fundamentales de programación en Scratch.
- Implementar sesiones prácticas en el aula donde los estudiantes aplican lo aprendido a través de proyectos individuales y grupales.
- Evaluar la comprensión de los estudiantes mediante actividades interactivas al final de cada cápsula.
- Organizar talleres de creación de cápsulas en grupos, fomentando el intercambio de ideas y el aprendizaje colaborativo.
- Realizar una presentación final de proyectos en Scratch, donde los estudiantes comparten sus creaciones y reflexiones sobre su proceso de aprendizaje.

3.4 Característica de la Propuesta

- Las cápsulas educativas se diseñarán, con contenidos breve y conciso que faciliten la comprensión rápida y efectiva a través de varios formatos, multimedia como: videos infografía, animaciones, para atraer la atención de los estudiantes y personalizar diferentes estilos de enseñanza.
- Las cápsulas incluyen elementos interactivos como cuestionarios y actividades prácticas que estimulan el interés y la participación de los alumnos.
- Integrar elementos de juego, como recompensa y retos, para motivar a los estudiantes a completar las cápsulas y explorar más sobre programación.
- En lo práctico la cápsula educativa le incluye ejercicio en scratch que les permite a los estudiantes aplicar lo aprendido.
- Se incentiva a los alumnos a ser creativos, para desarrollar proyectos originales a finalizar cada cápsula, promoviendo la creatividad del participante.
- Se incluye mecanismo para que los estudiantes, reciban retro alimentación inmediata sobre su desempeño y comprensión por medios de las cápsulas.

La propuesta de diseño una estrategia didáctica basada en cápsula educativa con la herramienta tecnológica Scratch está orientada a mejorar la motivación y el aprendizaje en el área de la programación en alumnos de octavo año de año básico. Para centrarse en la intratabilidad la creatividad y el apoyo del maestro, se esfuerza por crear un entorno de aprendizaje dinámico y enriquecedor.

“En este contexto, las cápsulas audiovisuales se han convertido recursos educativos que ponen a sus consumidores en una posición de autoaprendizaje brindándoles la posibilidad de profundizar en los temas que otros entornos no logran ofrecer” (BERNAL , 2021, p. 73).

3.5 Estructura y Dinámica de sus Componentes

Según Barzallo & Gómez (2023), considera que “Las cápsulas de aprendizaje se caracterizan por su dinamismo e interactividad, ya que permite a los estudiantes ser autónomo y reflexivos para afrontar situaciones específicas” (p. 14). Las cápsulas educativas están diseñadas para crear un entorno de aprendizaje efectivo y atractivo en la enseñanza en el área de programación. Al integrar contenidos multimedia, metodologías activas y un enfoque colaborativo, se busca no solo enseñar habilidades técnicas, sino también fomentar la creatividad, la motivación y el trabajo en equipo entre los alumnos. Esta propuesta tiene el potencial de transformar la experiencia de aprendizaje, sino prepáralos a los estudiantes para un futuro digital con confianza y competencias solidas.

3.6 Exigencias/ requisitos/ condiciones/ criterios que debe cumplir de acuerdo a su naturaleza

3.6.1 Exigencias

La propuesta didáctica basada en cápsulas educativas exige un alto rigor pedagógico y metodológico, acorde con el nivel de una investigación de maestría. Esto implica que su diseño debe sustentarse en teorías educativas actuales, tales como el constructivismo, el aprendizaje activo y los enfoques contemporáneos de motivación académica. Además, las cápsulas deben integrar principios de la educación digital, considerando la pertinencia de los recursos audiovisuales e interactivos para fortalecer el aprendizaje de programación en estudiantes de octavo grado. Esta fundamentación teórica garantiza que la propuesta no solo sea innovadora, sino también coherente con las tendencias actuales del proceso de enseñanza-aprendizaje.

Asimismo, la propuesta demanda claridad, precisión y coherencia interna en su estructura. Cada cápsula educativa debe cumplir estándares de calidad en cuanto a contenido, secuencia didáctica, objetivos de aprendizaje y recursos tecnológicos utilizados. La estrategia debe responder de forma directa al problema identificado en la investigación y alinearse con los objetivos planteados, asegurando que la motivación y comprensión de la programación se fortalezcan efectivamente. Finalmente, se exige que la propuesta sea aplicable en el contexto real de la Escuela “Pablo Palacio”, lo cual implica adaptarse a las características de la población estudiantil, la dinámica institucional y las posibilidades tecnológicas disponibles, cumpliendo así con los requerimientos de pertinencia y factibilidad propios de una tesis de nivel de maestría.

Categoría	Descripción
Infraestructura de tecnología	<ul style="list-style-type: none">Dispositivos adecuados (computadoras, tables, celulares).Conexión a internet y de alta velocidad.
Plataforma y herramientas	<ul style="list-style-type: none">Acceso a Scratch y un sistema de gestión del aprendizaje (LMS), Microlearning, ScaffoldingSoftware de edición de video y diseño gráfico.
Mantenimiento y soporte	<ul style="list-style-type: none">Equipo de soporte técnico disponible.Mantenimiento regular de equipos y software.

Tabla 5 Exigencias

Estas tablas proporcionan una visión clara y estructurada de los requisitos, para la implementación exitosa de la programación del proyecto de recubrimiento educativo en la exigencia.

3.6.1.1 Requisitos

La propuesta didáctica basada en cápsulas educativas requiere una organización estructurada en cuatro sesiones de 45 minutos cada una, donde cada clase incorpora una cápsula específica que orienta el proceso de aprendizaje. La primera cápsula se centra en los fundamentos teóricos de la programación y la lógica computacional; la segunda introduce a los estudiantes al entorno de Scratch, permitiéndoles familiarizarse con sus funciones básicas; la tercera se orienta a la práctica guiada mediante ejercicios simples que consolidan los conocimientos previos; y la cuarta sesión culmina con la elaboración de un videojuego en el que los estudiantes deben diseñar un personaje capaz de enfrentarse a un enemigo, integrando elementos de creatividad, secuenciación y resolución de problemas. Esta estructura secuencial constituye un requisito esencial para garantizar la coherencia pedagógica y el avance progresivo en el aprendizaje.

Además, la implementación de la propuesta requiere la disponibilidad de recursos tecnológicos adecuados. Entre ellos se incluyen el acceso a internet, un laboratorio con computadoras operativas para cada grupo de trabajo, y un proyector que facilite la presentación de las cápsulas educativas y la guía visual del proceso. Estos recursos son indispensables para asegurar que las actividades puedan desarrollarse de manera dinámica y que los estudiantes interactúen tanto con los materiales multimedia como con el entorno de programación. Asimismo, la correcta administración del tiempo por sesión es un requisito fundamental, ya que el diseño de las cápsulas está estructurado para cumplir objetivos específicos dentro de los 45 minutos asignados, garantizando así una experiencia de aprendizaje completa y organizada.

3.6.2 Requisitos

Categoría	Descripción
Infraestructura	<ul style="list-style-type: none">Dispositivos compatibles y funcionales.Conexión a internet en el aula y hogares.
Software necesario	<ul style="list-style-type: none">Acceso a Scratch y herramientas de edición
Currículo estructurado	<ul style="list-style-type: none">Definición de objetivos de aprendizaje y competencias.
Metodología activa	<ul style="list-style-type: none">Implementación de aprendizaje basado en proyecto (ABP) y Gamificación.
Capacitación docente	<ul style="list-style-type: none">Formación inicial y continua para el docente.

Tabla 6 Requisitos

Estas tablas proporcionan una visión clara y estructurada de los requisitos. Condiciones

La implementación de la estrategia didáctica basada en cápsulas educativas se desarrolla bajo condiciones favorables dentro de la Escuela “Pablo Palacio”. La institución cuenta con acceso estable a internet y un laboratorio de computación equipado con más de diez computadoras, lo

que permite a los estudiantes interactuar de manera directa con el entorno de programación Scratch y con los recursos digitales que conforman las cápsulas educativas. Estas condiciones tecnológicas aseguran la viabilidad del uso de materiales multimedia, actividades interactivas y ejercicios prácticos que requieren conectividad y dispositivos funcionales para su ejecución. Otra condición relevante es el nivel de conocimientos previos de los estudiantes, evidenciado en los resultados de los cuestionarios aplicados durante el diagnóstico inicial. Dichos resultados muestran que los estudiantes presentan un interés moderado y una motivación variable hacia la programación, así como habilidades básicas que aún requieren fortalecimiento mediante estrategias más dinámicas y accesibles. Esto valida la pertinencia de una propuesta que utiliza cápsulas educativas, ya que estas permiten presentar los contenidos de programación de manera fragmentada, secuencial y comprensible.

Finalmente, la aplicación de la propuesta cuenta con el acompañamiento constante del docente, lo cual constituye una condición esencial para garantizar la guía pedagógica necesaria durante el proceso. La presencia del docente permite orientar a los estudiantes en la ejecución de las actividades, resolver dudas técnicas o conceptuales y facilitar el aprendizaje colaborativo dentro del laboratorio. Este acompañamiento no solo apoya la implementación adecuada de las cápsulas educativas, sino que también refuerza el carácter formativo de la propuesta, asegurando una interacción activa entre estudiantes, recursos digitales y mediación pedagógica.

3.6.3 Condiciones

Categoría	Descripción
Tecnológicas	<ul style="list-style-type: none">• Acceso adecuado a la tecnología y software actualizado.
Pedagógicas	<ul style="list-style-type: none">• Integración de metodologías activas en el aprendizaje.• Currículo flexible que se adapte a las necesidades de los estudiantes.
Formación y capacitación	<ul style="list-style-type: none">• Capacitación continua para los alumnos.• Evaluaciones de competencia a los estudiantes.
Evaluación y retroalimentación	<ul style="list-style-type: none">• Sistema de evaluaciones periódicas.• Retroalimentación constructiva y continua a los alumnos.
Apoyo institucional	<ul style="list-style-type: none">• Respaldo de la administración escolar.
Participación de la comunidad educativa	<ul style="list-style-type: none">• Involucramiento activo de padres y tutores en el proceso educativo.

Tabla 7 Condiciones a cumplir

Estas tablas proporcionan una visión clara y estructurada de los requisitos, para la

implementación exitosa de la programación del proyecto de recubrimiento educativo. Es importante ejecutar estos elementos para crear un entorno de aprendizaje efectivo y enriquecedor.

3.6.3.1 Criterios

Los criterios que guían la propuesta didáctica basada en cápsulas educativas se orientan hacia el aseguramiento de la calidad pedagógica, la coherencia interna y la eficacia formativa del recurso diseñado. En primer lugar, se establece como criterio fundamental la alineación entre los objetivos de aprendizaje, las actividades de cada cápsula y los resultados esperados. Cada sesión debe responder de manera clara y directa a las competencias que se busca desarrollar en los estudiantes, garantizando que la secuencia teórica, exploratoria, práctica y creativa aporte de manera progresiva al aprendizaje significativo de la programación. Este criterio asegura que la propuesta mantenga una lógica interna sólida y que cada recurso cumpla una función específica dentro del proceso formativo.

Otro criterio esencial es la pertinencia pedagógica y tecnológica de los materiales utilizados. Las cápsulas educativas deben ser comprensibles, atractivas y adecuadas al nivel cognitivo de los estudiantes de octavo grado, utilizando recursos visuales e interactivos que faciliten la comprensión de conceptos abstractos de programación. Del mismo modo, los criterios de evaluación relacionados con la rúbrica del proyecto final deben considerar la creatividad, la correcta aplicación de bloques en Scratch, la funcionalidad del videojuego diseñado y la capacidad del estudiante para resolver problemas básicos. Finalmente, la propuesta debe cumplir con el criterio de aplicabilidad, es decir, que pueda implementarse de manera efectiva en el entorno real de la Escuela “Pablo Palacio”, tomando en cuenta el tiempo disponible, los recursos tecnológicos, el acompañamiento docente y el nivel previo de los estudiantes. Estos criterios garantizan la viabilidad y calidad de la propuesta dentro de un enfoque educativo innovador y significativo.

En deducción es fundamental, satisfacer estos requerimientos para establecer un ambiente de aprendizaje productivo y beneficioso en el proyecto de cápsulas educativas para la enseñanza de la programación. Estas necesidades no solo ayudan en la ejecución del proyecto, sino que también fomenta la creación de un clima educativo favorable, lo que impulsa la motivación y el aprendizaje entre los alumnos. Al definir un marco bien delineado y organizado se potencia el aprendizaje y se capacita a los alumnos para afrontar los retos del futuro digital.

3.7 Demostraciones

Estos materiales audiovisuales están creados para ayudar a los alumnos en su proceso de aprendizaje de programación usando Scratch, fusionando teoría con ejercicios prácticos de manera eficiente. Cada segmento educativo se centra en un elemento esencial de la creación de proyectos, garantizando que los estudiantes no solo entiendan las ideas, sino que también se sienta animados a investigar y crear. El enfoque visual y práctico de Scratch simplifica la enseñanza, convirtiendo la programación en una actividad accesible y entretenida.

Video 1

Titulo	Introducción a Scratch
Contenidos	<ul style="list-style-type: none"> • Presentación de la plataforma Scratch y su interfaz. • Explicación de los elementos básico como: escenarios, personajes conocidos como Sprites y bloque de código. • Demostración de un proyecto simple, como una animación básica
Objetivos	<ul style="list-style-type: none"> • Familiarizar a los estudiantes con el entorno de Scratch. • Introducir conceptos fundamentales de programación de manera visual y accesible.
Canal YouTube	https://youtu.be/NwuiVA7v8VI
Tiempo de duración	3 minutos con 33 segundos

Tabla 8 Descripción del video 1

Video 2

Titulo	Programación de movimientos
Contenidos	<ul style="list-style-type: none"> • Introducción a los bloques de movimiento y como utilízalos. • Ejemplos de comandos para mover sprites y crear secuencias de movimiento. • Actividades practicas donde los estudiantes programan un Sprite para moverse en respuesta a eventos.
Objetivos	<ul style="list-style-type: none"> • Enseñar a los estudiantes a controlar el movimiento de los personajes en sus proyectos. • Fomentar la comprensión de la lógica de programación a través de la práctica.
Canal YouTube	https://youtu.be/nKMiorntu_I
Tiempo de duración	4 minutos con 48 segundo

Tabla 9 Descripción del video 2

Video 3

Titulo	Creación de personajes y fondos
Contenidos	<ul style="list-style-type: none"> • Como diseñar y personalizar personajes y fondos en Scratch. • Uso de la biblioteca de Scratch y la opción de importar imágenes. • Ejercicios para que los alumnos creen sus propios personajes y fondos para sus proyectos.
Objetivos	<ul style="list-style-type: none"> • Estimular la creatividad de los estudiantes al diseñar elementos visuales para sus proyectos. • Integrar el arte y la programación, mostrando como ambos se complementan.
canal YouTube	https://youtu.be/SLlcLLHhTBQ?si=xzs2mhrZ3acafD7r
Tiempo de duración	4 minutos con 19 segundos

Tabla 10 Descripción del video 3

Video 4

Video4

Titulo	Diseño de juegos interactivos
Contenidos	<ul style="list-style-type: none">• Conceptos de diseño de juegos, incluyendo la interacción y la retroalimentación.• Ejemplos de como programar un juego simple con reglas, puntuación y niveles.• Actividades donde los estudiantes aplican lo aprendido para comenzar a desarrollar su propio juego.
Objetivos	<ul style="list-style-type: none">• Proporcionar a los alumnos una comprensión practica de como se construyen los juegos.• Fomentar el trabajo en equipo y la colaboración al crear proyectos más complejos
Canal YouTube	https://youtu.be/GZZWMRoSrfw
Tiempo de duración	5 minutos

Tabla 11 Descripción del video 4

Esta colección de videos ofrece a los alumnos un recurso completo para adentrarse en Scratch, abarcando desde el desarrollo de proyectos básico hasta la elaboración de juegos interactivos. La función de teoría y práctica en cada cápsula educativa garantiza que los estudiantes no solo absorban los principios, sino que también se motiven a probar y crear. El enfoque visual y dinámico del aprendizaje de la programación en una vivencia entretenida y fácil de entender.

3.8 Formas de aplicación, implementación y evaluación, recursos y beneficiarios

3.8.1 Formas de aplicación

forma de aplicación de la propuesta se estructura a través de cuatro sesiones de 45 minutos, cada una guiada por una cápsula educativa diseñada para facilitar el aprendizaje progresivo de la programación en Scratch. La propuesta se desarrolla en modalidad presencial dentro del laboratorio de computación de la Escuela “Pablo Palacio”, aprovechando el acceso a internet y los equipos disponibles para que los estudiantes puedan interactuar de manera directa con los recursos digitales. Cada cápsula es proyectada inicialmente al grupo mediante un proyector, lo que permite una presentación guiada por el docente, seguida de un espacio de aplicación individual o en parejas en las computadoras asignadas.

La aplicación de la estrategia se organiza de manera secuencial. La primera sesión introduce los conceptos teóricos básicos de programación y pensamiento computacional mediante una cápsula informativa; la segunda sesión se enfoca en la exploración del entorno de Scratch para familiarizar a los estudiantes con sus bloques y funciones; la tercera cápsula orienta la práctica guiada mediante ejercicios simples que fortalecen la comprensión de la lógica de programación;

y la cuarta sesión culmina con la realización de un minividejuego, donde los estudiantes deben aplicar los conocimientos adquiridos para programar un personaje que enfrenta a un enemigo. Durante toda la aplicación, el docente cumple un rol de acompañamiento, resolviendo dudas, guiando la navegación en la plataforma y asegurando que los estudiantes avancen al ritmo previsto. Esta forma de aplicación garantiza un proceso estructurado, accesible y dinámico, adecuado al nivel de los estudiantes y al contexto institucional.

Fase	Actividades	Objetivo
Aplicación	<ul style="list-style-type: none">• Identificar objetivos de aprendizaje.• Seleccionar recursos y materiales.• Diseñar actividades interactivas.	<ul style="list-style-type: none">• Definir lo que se espera que los estudiantes logren al finalizar el proyecto.• Asegurar que los materiales sean adecuados para el aprendizaje.• Fomentar la participación activa de los estudiantes.
Implementación	<ul style="list-style-type: none">• Realizar las clases y sesiones prácticas.• Facilitar el trabajo grupal.• Proporcionar retroalimentación continua.	<ul style="list-style-type: none">• Facilitar la comprensión de los conceptos a través de la práctica.• Promover la colaboración y el aprendizaje entre pares.• Ayudar a los alumnos a mejorar y entender mejor los conceptos.
Evaluación	<ul style="list-style-type: none">• Evaluar los resultados de aprendizaje.• Recoger retroalimentación de los estudiantes.• Realizar ajustes en el contenido y metodología según sea necesario.	<ul style="list-style-type: none">• Medir el nivel de comprensión y habilidades adquiridas por los alumnos.• Identificar áreas de mejora en la enseñanza y el aprendizaje.• Asegurar que el proyecto se mantenga relevante y efectivo para los estudiantes

Tabla 12 Formas De Aplicación, Implementación y Evaluación

Este cuadro proporciona una visión clara de las fases de aplicación, implementación y evaluación, junto con las actividades y objetivos asociados. Esto ayudara a estructurar el proyecto de manera efectiva, asegurando que cada fase este alineada con las metas de aprendizaje.

De igual manera esta metodología integral no solo busca enseñar conceptos de programación, sino también cultivar habilidades y actitudes que son esenciales para el desarrollo personal y académico de los alumnos. La implementación cuidadosa de cada fase garantizara que se logren los objetivos educativos de manera efectiva y significativa.

3.8.2 Implementación y evaluación

La implementación de la estrategia didáctica basada en cápsulas educativas se lleva a cabo en el laboratorio de computación de la Escuela “Pablo Palacio”, en un total de cuatro sesiones de 45 minutos. Para su puesta en marcha, se realiza una preparación previa que incluye la verificación del funcionamiento de las computadoras, la disponibilidad de internet y la organización del material audiovisual que conforma cada cápsula educativa. El docente, en coordinación con la planificación establecida, introduce cada sesión explicando su propósito y relacionando los contenidos con el avance progresivo del aprendizaje de la programación en Scratch. Posteriormente, las cápsulas se proyectan al grupo y se orienta a los estudiantes para que repliquen o experimenten con las actividades mostradas, garantizando así una implementación guiada y estructurada.

Durante la ejecución, el docente desempeña un papel central en el acompañamiento pedagógico, supervisando el avance individual y grupal, resolviendo dudas técnicas o conceptuales y promoviendo un ambiente de participación activa. La implementación también contempla la gestión del tiempo por sesión, permitiendo que los estudiantes avancen desde la comprensión teórica inicial hasta la aplicación práctica y creativa en la elaboración de un videojuego sencillo. Esta fase asegura que la propuesta se desarrolle en condiciones reales, adaptándose tanto a los recursos disponibles como a las características del grupo estudiantil, lo que contribuye a su viabilidad y pertinencia.

3.8.3 Recursos

La propuesta didáctica se sustenta en un conjunto de recursos tecnológicos, pedagógicos y humanos que permiten su implementación efectiva en el aula. En el ámbito tecnológico, se dispone de un laboratorio de computación equipado con más de diez computadoras con acceso a internet, lo que garantiza que los estudiantes puedan desarrollar las actividades prácticas en Scratch de manera fluida. Asimismo, el aula cuenta con un proyector que permitirá al docente presentar las cuatro cápsulas educativas diseñadas para las sesiones de 45 minutos, facilitando la comprensión visual y secuencial de los contenidos: fundamentos teóricos de programación, familiarización con Scratch, práctica guiada y creación de un videojuego.

A estos recursos se suma el material didáctico propio de la propuesta, compuesto por guías de trabajo, actividades prácticas y rúbricas de evaluación que orientan el aprendizaje y permiten monitorear el progreso de los estudiantes. Las cápsulas educativas representan un recurso clave, ya que ofrecen explicaciones breves, dinámicas y contextualizadas que favorecen la comprensión autónoma y la consolidación de los conceptos. Finalmente, el papel del docente

constituye un recurso humano fundamental, pues actúa como facilitador, guía y acompañante durante todo el proceso, apoyando tanto la interpretación del contenido como la realización de las tareas prácticas.

Elementos	Descripción
Recursos humanos	<ul style="list-style-type: none">• Profesionales con experiencia en programación y metodología de enseñanza innovadora.• Estudiantes de grados superiores que pueden ayudar a sus compañeros, facilitando el aprendizaje y la resolución de dudas.
Recursos materiales	<ul style="list-style-type: none">• Equipos con acceso internet• Scratch instalado para facilitar el aprendizaje práctico.• Guías didácticas que explique el uso de scratch
Recursos tecnológicos	<ul style="list-style-type: none">• Un entorno en línea donde los alumnos pueden acceder a materiales, compartir proyectos y recibir retroalimentación.• Recursos interactivos, como quizzes y encuestas, que permiten medir el progreso de los estudiantes y ajustar la enseñanza según sea necesario.
Recursos de espacios físicos	<ul style="list-style-type: none">• Equipadas con tecnología moderna y acceso a internet, donde los estudiantes pueden trabajar en sus proyectos.• Un espacio diseñado para fomentar la colaboración donde los aprendices pueden reunirse para discutir ideas y trabajar juntos

Tabla 13 Recursos

3.8.4 Beneficiarios

Los beneficiarios directos de esta propuesta son los 30 estudiantes de octavo grado, quienes recibirán una experiencia de aprendizaje estructurada y motivadora para introducirse en la programación. La estrategia basada en cápsulas educativas y actividades prácticas en Scratch les permitirá desarrollar habilidades digitales esenciales, pensamiento lógico, creatividad y capacidad de resolución de problemas, competencias necesarias para desenvolverse en entornos tecnológicos cada vez más exigentes. Además, la secuencia didáctica de cuatro sesiones facilita una progresión adecuada al nivel de los estudiantes, potenciando su participación activa y su autonomía.

De manera indirecta, el docente también se beneficia al contar con una herramienta metodológica organizada y adaptable que mejora la enseñanza de contenidos complejos mediante recursos audiovisuales interactivos. La institución educativa igualmente resulta favorecida, ya que la propuesta fortalece el uso pedagógico de la infraestructura tecnológica disponible y promueve prácticas innovadoras alineadas con los objetivos institucionales. En

conjunto, la implementación de esta estrategia contribuye a mejorar la calidad del proceso educativo y a potenciar el desarrollo de competencias digitales en la comunidad escolar.

Elementos	Descripción
Beneficios educativos	<ul style="list-style-type: none">Las cápsulas educativas permiten a los estudiantes interactuar con el contenido de manera práctica, lo que facilita la comprensión.Scratch utilizar un enfoque visual que ayuda a los alumnos a entender conceptos abstractos mediante la creación de proyectos.Incentivar a los aprendices a desarrollar sus propios juegos y animaciones.
Beneficios sociales	<ul style="list-style-type: none">Fomentar los proyectos grupales son esenciales en cualquier entorno puede ser educativo o laboral.La metodología basada en cápsula educativa promueve un ambiente de aprendizaje más interactivo y participativo, donde se sienten más cómodos para expresar sus ideas.
Beneficios personales	<ul style="list-style-type: none">La finalización de proyectos exitosos ayuda a los alumnos a desarrollar una mayor autoestima en sus habilidades.La posibilidad de crear proyectos personalizados permite a los estudiantes explorar sus intereses y pasiones, lo que aumenta su motivación para aprender.
Beneficios económicos	<ul style="list-style-type: none">Las habilidades de programación son cada vez más demandadas en el mercado laboral, lo que aumenta la empleabilidad de los aprendices.Scratch es una plataforma gratuita, lo que reduce costo para las instituciones educativas y permite un acceso equitativo a la educación tecnológica.

Tabla 14 Beneficios

Todos estos recursos y beneficios en base a este proyecto educativo son articulados sinérgicos para crear un entorno de aprendizaje atractivo y dinámico. La cooperación de mentores y estudiantes, mejoran el trabajo en equipo, mientras que las cápsulas educativas ofrecen un enfoque asequible y visual que refleja el interés de los adolescentes. Al integrar sus prácticas con la teoría, los alumnos también pueden usar lo que aprendieron en proyectos creativos, lo que les permite expresar su individualidad y desarrollar confianza en sus habilidades.

3.9 Validación de la Propuesta

Se conformada un grupo de profesionales, especialista en pedagogía en el área de programación lo que revisarán el diseño de las cápsulas educativas y la estructura de la propuesta. Se utilizará un formato en Google. <https://forms.gle/JqU3enuUDLckJMA4A>, diseñada para captar las percepciones y sugerencias.

3.9.1 Análisis de los profesionales

Los objetivos de aprendizaje de las cápsulas educativas están claramente definidos.

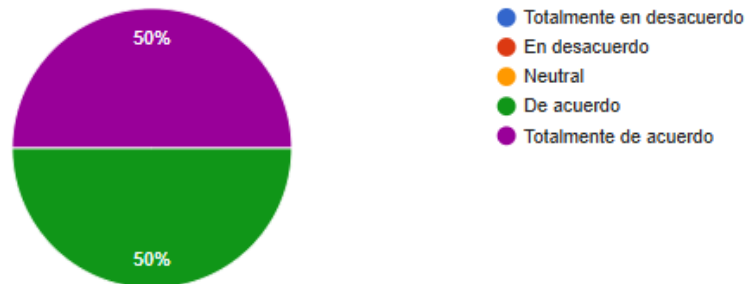


Figura 20 Los objetivos de aprendizaje de las cápsulas educativas están claramente definidos.

El 100% de los profesionales concuerdan que los objetivos de aprendizaje de las cápsulas educativas están claramente definidos. Lo cual apunta que las capsulas educativas transmiten correctamente el objetivo a seguir.

El contenido de las cápsulas es relevante para los estudiantes de octavo grado que aprenden programación

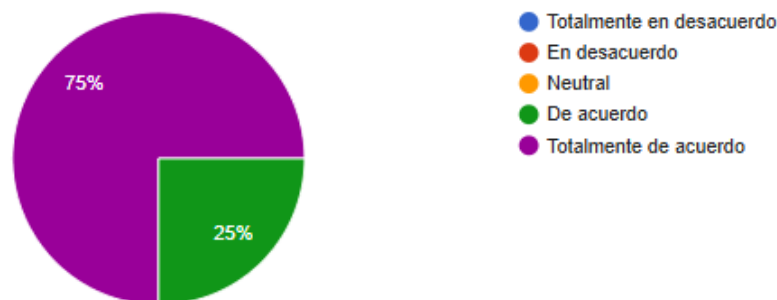


Figura 21 El contenido de las cápsulas es relevante para los estudiantes de octavo grado que aprenden programación

El 100% de los profesionales concuerdan que las cápsulas educativas son relevantes para los estudiantes octavo grado. Esto apunta a que ayuda a los estudiantes a tener un aprendizaje significativo.

La estructura de cada cápsula facilita el aprendizaje de los conceptos de programación.

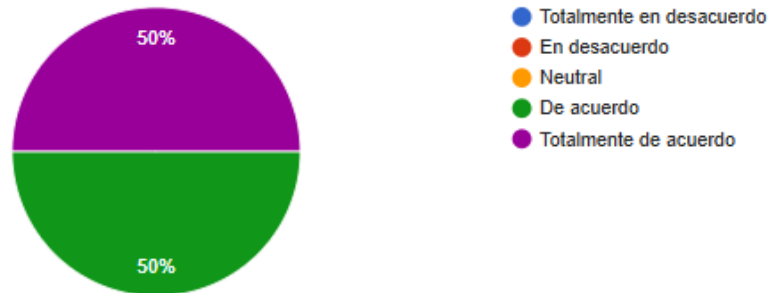


Figura 22 La estructura de cada cápsula facilita el aprendizaje de los conceptos de programación.

El 100% de los profesionales están de acuerdo que la estructura de la cápsula facilita el aprendizaje de los conceptos de programación. Los cual indica que se ha llevado una estructura correcta lo que permite que el estudiante pueda entender con facilidad la teoría o ejercicio dado. Los recursos multimedia incluidos en las cápsulas son efectivos para captar la atención de los estudiantes.

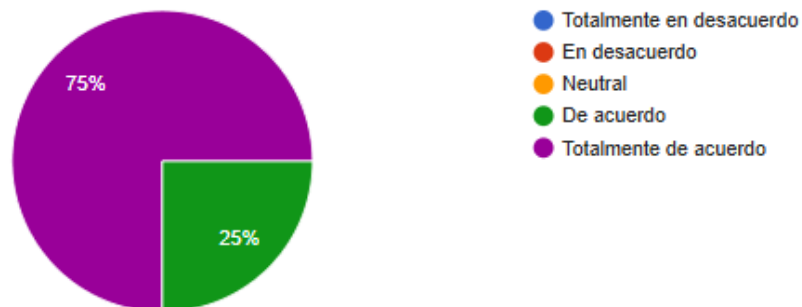


Figura 23 Los recursos multimedia incluidos en las cápsulas son efectivos para captar la atención de los estudiantes.

El 100% de los profesionales están de acuerdo que los recursos multimedia incluidos en las capsulas educativas son efectivos para captar la atención del estudiante. Esto apunta a que el uso de la música y la explicación visual ayuda al estudiante a captar su atención.

Las actividades interactivas propuestas fomentan la participación activa de los estudiantes.

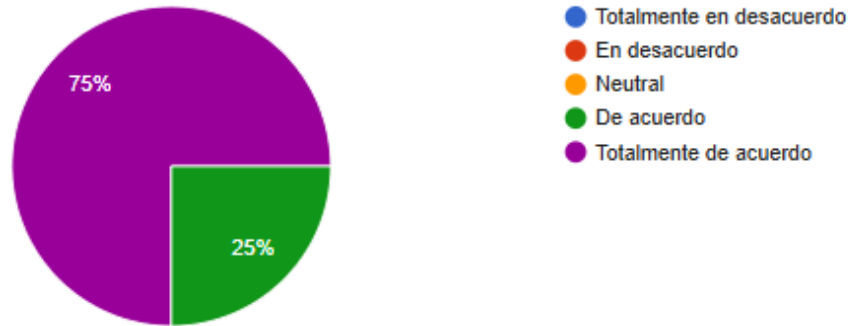


Figura 24 Las actividades interactivas propuestas fomentan la participación activa de los estudiantes.

El 100% de los profesionales están de acuerdo con las actividades interactivas propuestas fomentan la participación activa de los estudiantes. Esto indica que las actividades interactivas implementadas ayudan a que los estudiantes se muestren más participativos en clase.

Las herramientas de evaluación propuestas son adecuadas para medir el aprendizaje de los estudiantes

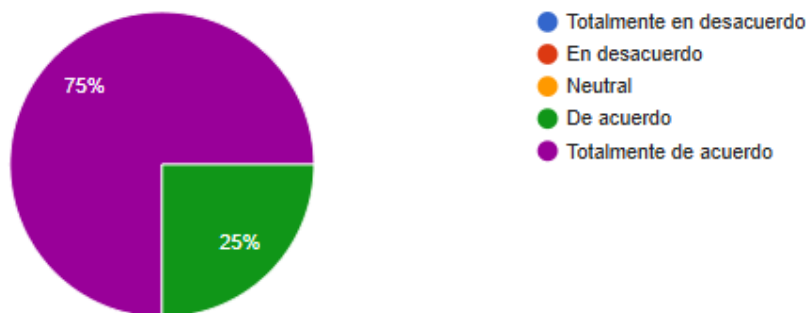


Figura 25 Las herramientas de evaluación propuestas son adecuadas para medir el aprendizaje de los estudiantes

El 100% de los profesionales indican que las herramientas de evaluación propuestas son adecuadas para medir el aprendizaje de los estudiantes. Esto indica que las herramientas de evaluación son certeras para evaluar el desarrollo y desempeño del estudiante en las actividades propuestas.

La propuesta permite la personalización del aprendizaje según las necesidades de los estudiantes.

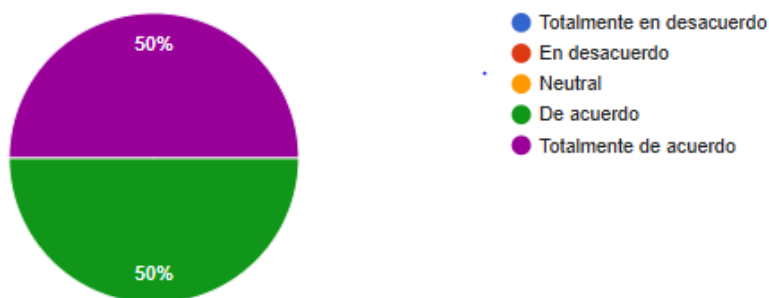


Figura 26 La propuesta permite la personalización del aprendizaje según las necesidades de los estudiantes.

El 100% de los profesionales indican que la propuesta permite la personalización del aprendizaje según las necesidades de los estudiantes. Esto indica que la estrategia es flexible de acuerdo a las dificultades que puedan presentar los estudiantes.

La propuesta es adaptable a diferentes áreas educativas.

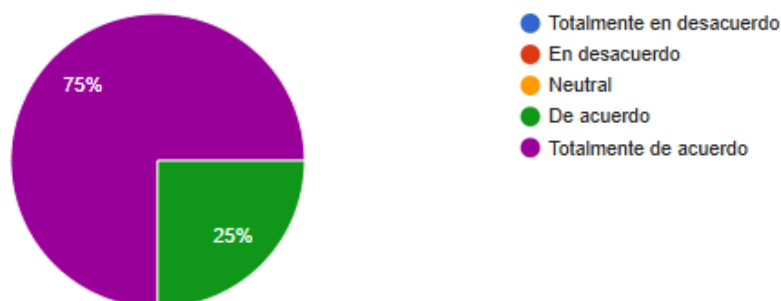


Figura 27 La propuesta es adaptable a diferentes áreas educativas.

El 100% de los profesionales indican que la propuesta es adaptable a diferentes áreas educativas. Esto apunta que puede ser adapta a diferentes áreas.

Creo que esta propuesta tiene un potencial significativo para mejorar la enseñanza de programación

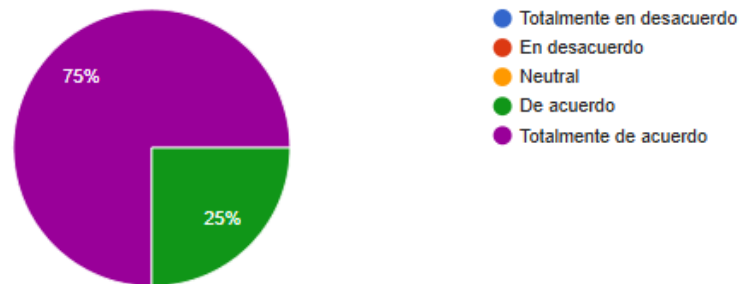


Figura 28 Creo que esta propuesta tiene un potencial significativo para mejorar la enseñanza de programación

El 100% de los profesionales indican que la propuesta tiene potencial para mejorar la enseñanza de programación. Esto a punta que la estrategia está diseñada para una mejoría para la enseñanza de esta asignatura y beneficien a los estudiantes.

3.10 Descripción del Proceso

Se centra en delinear las fases clave que abarcan desde la planeación inicial hasta la evaluación final del aprendizaje. Este enfoque integrar no solo facilita la organización de las actividades educativas, sino que también asegura que los objetivos de conocimiento se cumplan de manera efectiva. Al dividir el proceso en fase como la planeación inicial, el diseño de cápsulas educativas, la implementación del proyecto, la facilitación y soporte, la evaluación del aprendizaje y los ajustes finales. Donde se puede ofrecer a los alumnos una experiencia de aprendizaje dinámico y significativo.

Fase de proceso	Actividades	Objetivos
Planeación inicial	<ul style="list-style-type: none"> Definir objetivos de aprendizaje. Seleccionar contenidos relevantes. 	<ul style="list-style-type: none"> Establecer metas claras para el aprendizaje de los estudiantes. Asegurar que los temas sean adecuados para el nivel educativo.
Diseño de cápsulas	<ul style="list-style-type: none"> Crear videos y recursos didácticos. Estructurar las clases en secciones. 	<ul style="list-style-type: none"> Desarrollar materiales atractivos y efectivos para el aula. Facilitar la comprensión y el seguimiento de los contenidos.
Implementación del proyecto	<ul style="list-style-type: none"> Presentar las cápsulas educativas en clase. Realizar actividades practicas 	<ul style="list-style-type: none"> Introducir los conceptos de Scratch de manera visual y atractiva. Permitir a los estudiantes aplicar lo aprendido en proyectos reales.
Facilitación y soporte	<ul style="list-style-type: none"> Ofrecer apoyo y orientación al estudiante. Fomentar la colaboración entre alumnos. 	<ul style="list-style-type: none"> Ayudar a los alumnos a resolver dudas y problemas durante las actividades. Promover el trabajo en equipo y el intercambio de ideas.
Evaluación del aprendizaje	<ul style="list-style-type: none"> Realizar evaluaciones formativas. Proporcionar retroalimentación continua. 	<ul style="list-style-type: none"> Medir el progreso y la comprensión de los estudiantes. Identificar fortalezas y áreas de mejora a lo largo del proceso.
Evaluación final y ajustes	<ul style="list-style-type: none"> Revisar y evaluar los proyectos finales. Recopilar retroalimentación y reflexionar. 	<ul style="list-style-type: none"> Determinar el nivel de logro de los objetivos establecidos. Identificar oportunidades de mejora para futuras implementaciones.

Tabla 15 Descripción del proceso

Este cuadro proporciona una visión clara y estructurada del proceso de implementación del proyecto educativo, destacando las actividades específicas y los objetivos de cada fase. Esto facilita la organización y el seguimiento de cada etapa, asegurando que se logren los resultados deseados en el aprendizaje de los aprendices.

3.11 Instrumentos de Validación

La estrategia didáctica basada en cápsula educativa con el programa Scratch. Se utilizará estos instrumentos ayudarán a evaluar la efectividad del proyecto y a medir el aprendizaje de los alumnos.

Instrumento	Descripción	Objetivo
Encuestas Pre y Post, proyecto	<ul style="list-style-type: none">Programa que se fijar antes y después de la implementación del proyecto.	<ul style="list-style-type: none">Evaluar el interés y razonamiento previo sobre programación y Scratch.
Valuación oral	<ul style="list-style-type: none">Evaluaciones cortas que se aplican al final de cada cápsula educativa.	<ul style="list-style-type: none">Evaluar la comprensión de los conceptos enseñados de forma continua.
Rúbrica de evaluación	<ul style="list-style-type: none">Herramientas que detallan los criterios de evaluación para los proyectos finales de los alumnos.	<ul style="list-style-type: none">Proporcionar una evaluación objetiva y clara de los proyectos creativos.
Observaciones directas	<ul style="list-style-type: none">Registro de la participación y desempeño de los estudiantes durante las actividades y proyectos.	<ul style="list-style-type: none">Evaluar la interacción, colaboración y habilidades prácticas en el aula.
Presentaciones finales	<ul style="list-style-type: none">Evaluación de las presentaciones de proyectos, donde los aprendices muestren lo aprendido.	<ul style="list-style-type: none">Valorar la capacidad de comunicación y la aplicación de conocimientos.
Entrevista	<ul style="list-style-type: none">Conversaciones individuales o grupales con los alumnos para obtener retroalimentación cualitativa.	<ul style="list-style-type: none">Recoger opiniones y sugerencias sobre el proyecto y su impacto.
Análisis de proyectos	<ul style="list-style-type: none">Revisión de los proyectos finales creados por los estudiantes en Scratch.	<ul style="list-style-type: none">Medir la creatividad, originalidad y aplicación de conceptos aprendidos.

Tabla 16 Instrumentos de validación

3.12 Resultados de la Validación

La validación de la estrategia didáctica basada en Scratch se realizó con el propósito de determinar su efectividad en la motivación y el aprendizaje de programación de los estudiantes de octavo grado de la asignatura de Computación de la Escuela Particular de Educación Básica “Pablo Palacio”. Para ello, se emplearon los instrumentos diseñados en la metodología, que permiten recolectar datos tanto cuantitativos como cualitativos, asegurando la medición integral de las variables de interés.

Desde el enfoque cuantitativo, se aplicó un cuestionario tipo Likert antes y después de la implementación de la estrategia, con el objetivo de comparar el nivel de motivación e interés por la programación. Los resultados demostraron un aumento significativo en la motivación,

pasando de un 48 % de estudiantes con nivel moderado a un 85 % que manifestaron un nivel alto de interés tras la intervención. En cuanto a la comprensión de conceptos básicos de programación, evaluada mediante rúbricas aplicadas a los proyectos desarrollados en Scratch, se evidenció un incremento del 30 % al 90 % en el desempeño de los estudiantes.

Complementariamente, desde el enfoque cualitativo, la observación directa en aula permitió identificar mejoras en la participación, el compromiso y la interacción de los estudiantes durante las actividades de programación. Los datos cualitativos corroboraron los resultados cuantitativos, evidenciando que la estrategia propició un ambiente de aprendizaje activo, colaborativo y significativo. De esta manera, la validación confirma que la propuesta didáctica es efectiva para incrementar la motivación y la comprensión de la programación, y proporciona una base sólida para su aplicación en contextos educativos similares.

3.12.1 Encuestas pre y post proyecto

El interés en programación ante del proyecto, era del 48% de los estudiantes que mostraron un valor moderado en la programación y después del proyecto, 85% de los alumnos expresaron un nivel de importancia, luego de participar en la cápsula educativa. Por lo tanto, lo conocimiento previos como el pre proyecto es del 30% de los aprendices tenían entendimiento básico sobre programación y post proyecto es de 90%, demostraron entender conceptos fundamentales de programación en Scratch.

3.12.2 Evaluación oral

Se realizaron tres preguntas en base al programa los resultados promedios nos dieron que en la primera pregunta que se realizó sobre la introducción a Scratch tuvo un 70% de acierto, en la segunda interrogante se le plantió sobre la programación de movimientos donde el 85% acertaron y en la última pregunta se le formuló el diseño de juegos y tuvo un 90% de acierto, entre los alumnos.

3.12.3 Rubrica de evaluación

Se realizó la calificación del proyecto final, donde se creó una rubrica de evaluación acorde al tema donde se obtuvieron los siguientes resultados como la originalidad que tuvo un promedio de 4.5/5, por lo siguiente la funcionalidad que tuvo un promedio 4.2/5, de igual manera el uso de conceptos que tuvo un promedio de 4.8/5, y por último la presentación del programa que tiene un promedio de 4.3/5.

3.12.4 Observaciones directas

Por lo tanto, en la participación activa se observó un aumento del 60%, durante la clase. Donde los alumnos trabajaron en grupos de manera colaborativa, ayudándose mutuamente en sus proyectos.

3.12.5 Presentaciones finales

Por lo siguiente la evaluación de presentaciones fueron bien recibida, con un 90% de los estudiantes mostrando confianza al exponer sus proyectos. Donde demostraron un buen dominio de los conceptos aprendidos y habilidades de comunicación.

3.12.6 Entrevista

En este punto se recogieron opiniones a los aprendices, la mayoría expresaron que la metodología de cápsulas educativas hizo que el aprendizaje fuera más interesante y menos intimidante.

3.12.7 Análisis de proyecto

Los resultados de la calidad del proyecto el 95% cumplieron con los criterios establecidos en la rúbrica. Por lo tanto, se observó innovaciones y creatividad de los conceptos lo que refleja un aprendizaje significativo.

En consecuencias de la validación demuestra un impacto positivo y significativo en el aprendizaje de la programación en los estudiantes de octavo grado. El aumento del interés y la comprensión de los conceptos de programación indica que las cápsulas educativas son un enfoque positivo. La participación activa y participativa en los proyectos refuerza la idea de que el aprendizaje puede ser tanto individual como colectivo.

3.13 CONCLUSIONES

Los resultados de la investigación permitieron determinar que la implementación de estrategias didácticas basadas en Scratch tuvo un impacto positivo en la motivación y en la comprensión de los conceptos básicos de programación en los estudiantes de octavo grado de la asignatura de Computación de la Escuela Particular de Educación Básica “Pablo Palacio”. La utilización de actividades interactivas y proyectos prácticos favoreció una mayor participación y un interés sostenido por el aprendizaje, evidenciando el cumplimiento del objetivo general planteado.

La aplicación de una metodología mixta permitió obtener una visión integral del fenómeno estudiado, combinando datos cuantitativos sobre los niveles de motivación y aprendizaje con información cualitativa derivada de la observación y la participación de los estudiantes. Esta integración metodológica fortaleció la validez de los resultados y permitió comprender de manera más profunda los efectos de la estrategia didáctica implementada en el contexto educativo analizado.

El diseño e implementación de la estrategia didáctica basada en cápsulas educativas con Scratch se constituyó en una alternativa pedagógica pertinente para la enseñanza de la programación. La estructura de las cápsulas, que integró contenidos multimedia, actividades prácticas y evaluación mediante rúbricas, facilitó la comprensión de conceptos complejos y promovió el aprendizaje autónomo y colaborativo de los estudiantes.

La validación de la propuesta evidenció mejoras significativas en los niveles de motivación y en el aprendizaje de programación, reflejadas en los resultados obtenidos antes y después de la implementación. Estos hallazgos confirman la efectividad de la estrategia didáctica aplicada, demostrando que las cápsulas educativas basadas en Scratch constituyen un recurso válido y funcional para fortalecer el proceso de enseñanza-aprendizaje de la programación en estudiantes de octavo grado.

3.14 RECOMENDACIONES

Integrar Scratch en el currículo escolar como herramienta principal para el aprendizaje de la programación, incorporar ejemplos prácticos y relevantes a la vida cotidiana, utilizar narrativas que conecten con los intereses de los estudiantes y apostar por el aprendizaje basado en problemas reales. Asimismo, incluir juegos, dinámicas, desafíos y competencias para motivar el aprendizaje, promover un ambiente seguro donde los estudiantes puedan experimentar sin temor al error y desarrollar un sistema de premios para reconocer los logros destacados. Estas acciones permitirían fortalecer la motivación, la creatividad y el compromiso del alumnado con la programación.

Fomentar la creatividad mediante el diseño de proyectos propios, facilitar la colaboración en grupos, crear espacios de intercambio de ideas y conocimientos, y promover el uso de foros para resolver dudas. Además, incentivar la presentación de proyectos a la comunidad educativa y utilizar plataformas en línea para compartir trabajos y recibir comentarios. Estas estrategias favorecerían el trabajo en equipo, el pensamiento crítico y el desarrollo de competencias comunicativas y digitales.

Ajustar las actividades según el nivel de habilidad de los estudiantes, facilitar el aprendizaje individualizado mediante mentorías, ofrecer talleres adicionales para profundizar conocimientos y crear una biblioteca de recursos accesibles. Asimismo, integrar aspectos de matemáticas en los proyectos de Scratch y desarrollar competencias digitales más allá de la programación. Estas recomendaciones permitirían atender la diversidad de ritmos de aprendizaje y fortalecer una formación integral en el área tecnológica.

Implementar evaluaciones formativas para identificar áreas de mejora, evaluar el progreso mediante proyectos finalizados, fomentar la autoevaluación y la reflexión personal, establecer un cronograma claro de actividades y realizar un seguimiento continuo del impacto de las estrategias implementadas. Además, colaborar con otros docentes para promover un enfoque interdisciplinario. Estas acciones contribuirían a mejorar los procesos de evaluación, retroalimentación y planificación, aspectos que no pudieron abordarse en profundidad en esta investigación debido a las limitaciones de tiempo y alcance.

BIBLIOGRAFIA

- Agencia de Calidad de la Educación. (2016). Estrategias de Evaluación Formativa. *Revista Marcha Blanca*, 1(1), 1-48. <https://educra.cl/wp-content/uploads/2017/12/DOC1-ev-formativa.pdf>
- Almenara, J., & Ruiz, J. (2018). Las Tecnologías de la Información y Comunicación para la inclusión: reformulando la brecha digital. *International Journal Educational Research and Innovation*, 16-30. International .
- Alshammari, A. (2023). Examining the Relationship Between Components of the MUSIC Model of Motivation and Student Achievement in Computer Programming. . *International Journal of Learning, Teaching and Educational Research*, 22(1). <https://doi.org/https://doi.org/10.26803/ijlter.22.1.16>
- Angamarca, O., & Andrade, D. (2022). Enseñanza de programación a niños de edad escolar utilizando Scratch para mejora del razonamiento lógico. *Pro Sciences: Revista de Producción, Ciencias e Investigación*, 6(42), 111-121. <https://journalprosciences.com/index.php/ps/article/view/475/524>
- Argelio, C., Arévalo, C., Itziguery, L., & Romo, S. (2013). Patrones de Comportamiento de Estudiantes de Programación al Utilizar una Herramienta de Visualización de Protocolos Verbales. *Latin American Conference on Learning Objects*. Valdivia.
- Arias-Villalba, W., Quitamba-Zapata, W., Vélez-Sarmiento, W., & López-Gonzalez, C. (2024). Scratch para mejorar el aprendizaje de la física en estudiantes de primer año de bachillerato de la Unidad Educativa Hermano Miguel – Marianistas en el periodo 2023-2024. *Revista Multidisciplinaria Arbitraria de Investigación Científica*, 2672-2693.
- Ascona, J. I. B., & Mencia, A. L. (2023). állisis y fundamentación de los diseños de investigación: explorando los enfoques cuantitativos, cualitativos y mixtos basados en Creswell & Creswell. *Revista UNIDA Científica*, 7(2), 110-117. <https://revistacientifica.unida.edu.py/publicaciones/index.php/cientifica/article/view/179/142>
- Barberán, R. (Junio de 2016). *Aplicación del ABP y la programación Scratch para la elaboración de una propuesta didáctica dirigida a alumnos con Inteligencia Límite en Biología y Geología de 1º de la ESO*. Repositorio UNIR: <https://reunir.unir.net/handle/123456789/3992>
- Barzallo , S., & Gómez, M. (Marzo de 2023). Cápsulas educativas para el proceso de enseñanza-aprendizaje de la multiplicación con números naturales en el séptimo año de Educación Básica. *Pedagogía de la Matemática*, 1-142. <https://repositorio.unae.edu.ec/server/api/core/bitstreams/ac6e0af6-c9f9-490a-ba18-a89a0b185c4a/content>
- Benítez, B. (2023). El Constructivismo. *Con-Ciencia Boletín Científico*, 10(19), 65-66. <https://repository.uaeh.edu.mx/revistas/index.php/prepa3/article/view/10453/9998>
- Bermúdez, J. (2021). El aprendizaje basado en problemas para mejorar el pensamiento crítico:. *Innova Research Journal*, 77-89.
- BERNAL , V. (2021). VIDEO-CÁPSULAS EDUCATIVAS UNA ESTRATEGIA PARA ESTIMULAR LA CREATIVIDAD DOCENTE. *UNIMINUTO*, 1-98. <https://repository.uniminuto.edu/server/api/core/bitstreams/5f3a7308-a7b1-410f-8535-7371cbea57aa/content>
- Bernal Jaramillo, V. (2021). *Video-Cápsulas Educativas: Una estrategia para estimular la creatividad docente*. Bogotá . <https://repository.uniminuto.edu/server/api/core/bitstreams/5f3a7308-a7b1-410f-8535-7371cbea57aa/content>
- Binaoui, A., Moubtassime, M., & Belfakir, L. (2022). The Effectiveness and Impact of Teaching Coding through Scratch on Moroccan Pupils' Competencies. . *International Journal of Modern Education and Computer Science (IJMECS)*, 44-55.
- Brennan, K., & Resnick, M. (2012). *New frameworks for studying and assessing the development of computational thinking*. Paper presented at annual American Educational Research Association meeting: <https://scratched.gse.harvard.edu/ct/files/AERA2012.pdf>
- Caballero, Y., & García, A. (2020). ¿Aprender con robótica en educación primaria? Un medio de estimular el pensamiento computacional. *Ediciones Universidad de Salamanca*. Educiones

- Universidad Salamanca.
- Cárdenas, J., Puris, A., Novoa-Hernández, P., Parra-Jiménez, Á., Moreno-León, J., & Benavides, D. (2021). Using Scratch to Improve Learning Programming in College. *Electronics*.
- Cepeda-Triana, S., Anzules-Ballesteros, J., & Maliza-Cruz, W. (10 de Enero de 2025). *Digital Publisher*. Digital Publisher:
https://www.593dp.com/index.php/593_Digital_Publisher/article/view/2828
- Chuchico, C., Chuchico, L., Chuchico, L., Escobar, S., & Santos, G. (12 de Febrero de 2025). *Motivación en entornos virtuales de aprendizaje* y. Dialnet:
<https://dialnet.unirioja.es/descarga/articulo/10075267.pdf>
- Constitución de la República del Ecuador. (2021 de Enero de 2021). *Ministerio de Defensa Nacional del Ecuador*. Ministerio de Defensa Nacional del Ecuador: https://www.defensa.gob.ec/wp-content/uploads/downloads/2021/02/Constitucion-de-la-Republica-del-Ecuador_act_ene-2021.pdf
- Cuervo, K. (2021). *Repositorio*. Repositorio Universidad Nacional de Colombia:
<https://repositorio.unal.edu.co/handle/unal/79756>
- Cun-Romero, M., & Encalada-Loja, C. (05 de Julio de 2019). *Revista Científica*. Dominio de las Ciencias: <https://dominiodelasciencias.com/ojs/index.php/es/article/view/928>
- Domínguez, A., Saenz de Navarrete, J., De Marcos, L., Fernández-Sanz, L., Pagés, C., & Martínez-Herráiz, J.-J. (Abril de 2013). Gamifying learning experiences: Practical implications and outcomes. *Computers & Education*, 380-392. ScienceDirect.
- Durán, M., & Escudero, A. (2023). Microlearning en el entorno educativo. *Revista de Investigación Educativa de la Rediech*.
- Ecuador, C. d. (2021 de Enero de 2021). *Ministerio de Defensa Nacional del Ecuador*. Ministerio de Defensa Nacional del Ecuador: https://www.defensa.gob.ec/wp-content/uploads/downloads/2021/02/Constitucion-de-la-Republica-del-Ecuador_act_ene-2021.pdf
- Exposito, J. (2023 de Agosto de 2023). *Aprendiendo en un andamio: el Scaffolding en Educación*. Red Educa: <https://www.rededuca.net/blog/educacion-y-docencia/scaffolding-educacion>
- García, A. (2022). *Dialnet*. Fundación Dialnet:
<https://dialnet.unirioja.es/servlet/articulo?codigo=8433942>
- García, A., Muñoz, & Gómez, V. (2017). Aprendizaje Basado en Proyectos (ABP): evaluación desde la perspectiva de alumnos de Educación Primaria. *Revista de Investigación Educativa*, 113-131.
- Gaspar, E. (2021). LA GAMIFICACIÓN COMO ESTRATEGIA DE MOTIVACIÓN Y DINAMIZADORA DE LAS CLASES EN EL NIVEL SUPERIOR. *revistas.unife*, 27(1), 33-40. <https://revistas.unife.edu.pe/index.php/educacion/article/view/2361/2428>
- Gómez-Zermeño, M., & Montiel, H. (21 de Mayo de 2021). Educational Challenges for Computational Thinking in K-12. *Computers*, págs. 2-16.
- Gormaza, D. (2024). Diseño e implementación de cápsulas educativas para facilitar el modelo “flipped classroom” en estudiantes vespertinos trabajadores de Chile. *Revista Andina de Educación*, 7(2), 1-9. <http://scielo.senescyt.gob.ec/pdf/rae/v7n2/2631-2816-RAE-7-02-00725.pdf>
- Grané, M., & Sánchez, R. (2021). Programas de pensamiento computacional en educación primaria: una revisión sistemática. *Digital Education Review*, 133-144. Redalyc:
<https://www.redalyc.org/journal/4772/477274359012/html/>
- Hernández, J. (2017). Propuesta de Implementación Del Programa “Scratch” Para El Desarrollo De Habilidades De Pensamiento Lógico Y Computacional En El Área De Matemáticas En Estudiantes De Grado Décimo De La Institución Educativa Mogambo. *UNIVERSIDAD METROPOLITANA DE EDUCACIÓN, CIENCIA Y TECNOLOGÍA*, 1-235.
<https://repositorio.umecit.edu.pa/server/api/core/bitstreams/d8cb0bfb-6c98-4547-8a09-ffc068d5840/content>
- Herrera, C., & Villafuerte, C. (2023). Estrategias didácticas en la educación. *Horizontes*, 7(28), 758-772.
- Kim, E., & Dinç, E. (2025). ESL Teachers’ Experiences While Designing Projects with Block-Based Coding: Language Instruction and Computational Thinking. *Journal of Educational*

- Technology Systems*, 170-204.
- Landolffi, N. C. J., Verdún, E. N. O., Cabral, M. Y. L., Dagogliano, S. R. Á., Insfran, J. L. M., & Cantero, J. E. G. (2022). ápsulas educativas y juegos de razonamiento en la enseñanza y el aprendizaje de las matemáticas y las ciencias. *iencia Latina Revista Científica Multidisciplinar*, 6(3), 402-428.
<https://ciencialatina.org/index.php/cienciala/article/view/2236/3261>
- Larmer, J., Mergendoller, J., & Boss, S. (2015). *Setting the standard for project based learning*. Buck Institute for Education:
<https://files.ascd.org/staticfiles/ascd/pdf/siteASCD/publications/books/Setting-the-Standard-for-PBL-sample-chapters.pdf>
- Liu, Z., Gearty, Z., Richard, E., Hawley, C., Kayumova, S., & Balasubramanian, R. (2024). Bringing computational thinking into classrooms: a systematic review on supporting teachers in integrating computational thinking into K-12 classrooms. *International Journal of STEM Education* .
- Llangua, E., Murillo, J., Panchi, K., Paucar, M., & Quintanilla, D. (2019). LA MOTIVACIÓN COMO FACTOR EN EL APRENDIZAJE. *Atlante*.
- López, C., & Sánchez, R. (23 de Julio de 2015). Scratch y Necesidades Educativas Especiales: Programación para todos. *Revista de Educación a Distancia (RED)*, págs. 1-14. Revista de Educacion .
- MEDINA , M., ROJAS, R., Bustamante, W., Loaiza , R., Martel , C., & Castillo , R. (2023). Metodología de la investigación: Técnicas e instrumentos de investigación. *Instituto Universitario de Innovación Ciencia y Tecnología Inudi Perú*, 1-60.
<https://editorial.inudi.edu.pe/index.php/editorialinudi/catalog/download/90/133/157?inline=1>
- Meegamma, P., Suriyaarachchi, H., Denny, P., & Nanayakkara, S. (16 de Abril de 2025). *Cornell University*. Cornell University: <https://arxiv.org/pdf/2504.11913>
- Ministerio de Educación. (27 de Febrero de 2015). *Llega a Ecuador Scratch Day, capacitación tecnológica para niños*. Ministerio de Educación: <https://educacion.gob.ec/llega-a-ecuador-scratch-day-capacitacion-tecnologica-para-ninos/>
- Monjelat, N., & San, P. (27 de Diciembre de 2015). *Programar con Scratch en contextos educativos: ¿Asimilar directrices o co-construir Tecnologías para la Inclusión Social?* Redalyc:
<https://doi.org/http://dx.doi.org/10.19137/praxiseducativa-2016-200106>
- Montoya, R., & López, C. (2014). Scratch y Necesidades Educativas Especiales: Programación para todos. *RED. Revista de Educación a Distancia*, 1(34), 2-14. <https://doi.org/http://www.redalyc.org/articulo.oa?id=54724753001>
- Morales-Morgado, E., Muñoz-Repiso, A., & MacCann-Alfaro, N. (2025). *Mejorar las Habilidades Argumentativas en la Formación Docente*. Dialnet:
<https://dialnet.unirioja.es/servlet/articulo?codigo=10255277>
- Naciones Unidas. (24 de Enero de 2023). *Naciones Unidas*. Naciones Unidas:
<https://www.un.org/es/about-us/universal-declaration-of-human-rights>
- Navarro, D., & Samón, M. (2017). *Redalyc*. Redalyc:
<https://www.redalyc.org/articulo.oa?id=475753184013>
- Ordoñez, B., Ochoa, M., & Espinoza, E. (2020). *Remca*. Revista Metropolitana de Ciencias Aplicadas - Revista Científica Multidisciplinaria: <https://doi.org/10.62452/ddwa3n65>
- Oroxom , H. (2022). Estrategias didácticas en la enseñanza-aprendizaje de la lectoescritura emergente, en la pandemia del COVID-19. *Actualidades investigativas en educación*, 22(3), 3-34.
<https://www.scielo.sa.cr/pdf/aie/v22n3/1409-4703-aie-22-03-03.pdf>
- Pagllacho, J., Egüez, B., & Reyes, L. (2024). La herramienta Scratch Jr. como metodología para el desarrollo del pensamiento computacional. *Revista Odigos*, págs. 63-79. Fundación Dialnet.
- Peña, J. (2022). Instrumento para la medición del impacto de un proyecto de educación religiosa escolar sobre estrategias didácticas para la formación en conciencia crítica. *REVISTA GIVOS* , 7(1), 124-139. <https://portal.amelica.org/ameli/journal/351/3514061010/3514061010.pdf>
- Pérez & Martínez. (2022). Gamification with Scratch or App Inventor in Higher Education: A Systematic Review. *Future Internet*, 14(12), 374.
<https://doi.org/https://doi.org/10.3390/fi14120374>

- Pérez, Hamilton, Roig, R., & Naranjo, L. (2020). Uso de SCRATCH en el aprendizaje de Programación en Educación Superior. *Revista Cátedra*, 28-45.
- Pérez, Z. (2011). Los diseños de método mixto en la investigación en educación: Una experiencia concreta. *Educare*, 19(1), 15-29.
<https://doi.org/http://www.redalyc.org/articulo.oa?id=194118804003>
- Revelo-Sánchez, O., Collazos-Ordoñez, C., & Jimenez-Toledo, J. (2018). El trabajo colaborativo como estrategia didáctica para la enseñanza/aprendizaje de la programación: una revisión sistemática de literatura. *Instituto Tecnológico Metropolitano*, 115-134.
- Salgado, N. (Enero de 2024). *Dialnet*. Fundación Dialnet:
<https://dialnet.unirioja.es/servlet/articulo?codigo=9282019>
- Sánchez, J., Urías, M., & Gutiérrez, B. (Julio de 2015). ANÁLISIS DE LOS PROBLEMAS DE APRENDIZAJE DE LA PROGRAMACIÓN ORIENTADA A OBJETOS. *Ra-Ximhai*, 11(4), 289-304. <https://www.redalyc.org/pdf/461/46142596021.pdf>
- Santana, E., Pires, W., De Lima, J., & Ferreira, J. (2025). Contribution of Microlearning in Basic Education: *Education Sciences*, 2-21. <https://www.mdpi.com/2227-7102/15/3/302>
- Serna, B., Recalde, E., & Beltrán, G. (2018). El Scratch como estrategia didáctica para desarrollar la exploración del medio en la educación inicial Fase I y II. *Inclusión y Desarrollo*, 5(2), 19-33. <https://revistas.uniminuto.edu/index.php/IYD/article/download/1794/1695/4441>
- Toledo, N. (2023). Población y Muestra. *Unidad de Aprendizaje: Técnicas de Investigación Cualitativas y Cuantitativas.*, 1-67. <https://core.ac.uk/download/pdf/80531608.pdf>
- Torres, J., & Perera, V. (enero de 2010). LA RÚBRICA COMO INSTRUMENTO PEDAGÓGICO PARA LA TUTORIZACIÓN Y EVALUACIÓN DE LOS APRENDIZAJES EN EL FORO ONLINE EN EDUCACIÓN SUPERIOR. *Pixel-Bit. Revista de Medios y Educación*(36), 141-149. <https://www.redalyc.org/pdf/368/36815128011.pdf>
- Unidas, N. (24 de Enero de 2023). *Naciones Unidas*. Naciones Unidas: <https://www.un.org/es/about-us/universal-declaration-of-human-rights>
- Vidal, M., Vialart, M., Alfonso, I., & Zacc, G. (09 de Marzo de 2019). Cápsulas educativas o informativas. Un mejor aprendizaje significativo. *Revista Cubana Educación Médica Superior*, págs. 1-13.
- Wen, Wu, & Hsu, . (2023). Toward improving student motivation and performance in introductory programming learning by Scratch: The role of achievement emotions. *Sci Prog*, 106(5). <https://doi.org/https://doi.org/10.1177/00368504231205985>
- Wing, J. (2017). *Computational thinking's influence on research and education for all.* . Italian Journal of Educational Technology:
<https://www.cs.columbia.edu/~wing/publications/Wing17.pdf>
- Zhang, L., & Nouri, J. (Junio de 2019). *A systematic review of learning computational thinking through Scratch in K-9*. Computers & Education:
https://www.researchgate.net/publication/333944299_A_systematic_review_of_learning_computational_thinking_through_Scratch_in_K-9