



UNIVERSIDAD
BOLIVARIANA
DEL ECUADOR

TRABAJO DE TITULACIÓN



UNIVERSIDAD BOLIVARIANA DE ECUADOR

MAESTRÍA EN EDUCACIÓN CON MENCIÓN EN ENTORNOS DIGITALES

**TRABAJO DE TITULACIÓN PREVIO A LA OBTENCIÓN DEL TÍTULO DE:
MAGÍSTER EN EDUCACIÓN CON MENCIÓN EN ENTORNOS DIGITALES**

TEMA

**LA HERRAMIENTA CODING C++ PARA MEJORAR EL PROCESO DE
ENSEÑANZA-APRENDIZAJE DE LA ASIGNATURA DE PROGRAMACIÓN
ESTRUCTURADA**

Autor/es:

**ROBLES MEDINA CARLOS MARINO
SALINAS VERA EDISON FLORENCIO**

Tutor/a:

DRA. NIVELA CORNEJO MARIA ALEJANDRINA

ECUADOR

2024



La Universidad para todos



DEDICATORIA

Éste trabajo se lo dedico principalmente a Dios por ser mi motor y guía en este proceso.

A mis padres Roque Robles y Rosa Medina por darme la vida, a mi adorado hijo Carlos Mateo Robles, por ser mi inspiración, mi fuerza y mi motivación para seguir adelante, a mi amada esposa Diana Sánchez por su apoyo incondicional y siempre creer en mí, incluso en momentos que ni yo mismo lo hacía, y por luchar junto a mi lado en todo momento para cumplir nuestros sueños.

Ing. Carlos Marino Robles Medina



AGRADECIMIENTO

Principalmente agradezco a Dios, por la fuerza y valentía que me ha dado para enfrentar los retos de la vida, a su vez permitirme cumplir cada una de las metas propuestas, por sus bendiciones a lo largo del camino y por ubicar personas grandiosas en mi camino que de alguna forma han contribuido para que esta meta se cumpla.

A mi amada esposa Diana Merli Sánchez Lalangui, por ser quien me ha acompañado en todo el desarrollo del trabajo de investigación, gracias por sus palabras de aliento y motivación, buscando siempre alentarme a seguir adelante y cumplir este objetivo, sin duda su apoyo incondicional fue de gran ayuda.

A mis padres, hermanos, compañeros y a todas aquellas personas de que una u otra manera contribuyeron para que esta meta se haga realidad.

Ing. Carlos Marino Robles Medina



RESUMEN

El proceso de enseñanza-aprendizaje de programación estructurada en el segundo año de bachillerato de la Unidad Educativa Amazonas, se emplea bajo un enfoque pedagógico tradicional, carece de integración de herramientas digitales y flexibilidad para satisfacer las necesidades de aprendizaje ubicuo de los estudiantes. El objetivo consistió en generar una propuesta para el uso de la herramienta “CODING C++” en el mejoramiento del proceso de enseñanza-aprendizaje de la asignatura de programación estructurada en los estudiantes de segundo de bachillerato paralelo “A”, de la Unidad Educativa Amazonas, durante el periodo febrero a octubre de 2024. Se asume la ruta de investigación con enfoque mixto, alcance descriptivo y propositivo; se aplicaron dos instrumentos: un cuestionario y una entrevista semiestructurada, a una muestra de 32 estudiantes y tres docentes. Como resultados la mayoría de los estudiantes expresaron que sus docentes casi nunca consideran la dimensión educativa, metodológica, interactiva, motivacional y evaluativa; esto es, casi nunca: brindan una base sólida de conocimientos teóricos y la oportunidad de aplicarlos en situaciones prácticas. Se diseñó una propuesta de una guía titulada: “Guía para el uso de la Herramienta “CODING C++” en el proceso de enseñanza-aprendizaje de Programación Estructurada”; la cual contiene seis apartados, elaborada con un diseño llamativo y una estructura constructivista. Se concluyó que la guía generada enriquece la experiencia educativa, Simplifica la comprensión de los conceptos y permite a los estudiantes poner en práctica lo que han aprendido en situaciones reales, lo que es crucial para el aprendizaje de la programación.

Palabras clave: Herramienta “CODING C++”, proceso de enseñanza-aprendizaje, programación estructurada, educación.



ABSTRACT

The teaching-learning process of structured programming in the second year of high school at the Amazonas Educational Unit, is used under a traditional pedagogical approach, lacks integration of digital tools and flexibility to satisfy the ubiquitous learning needs of students. The objective was to generate a proposal for the use of the “CODING C++” tool to improve the teaching-learning process of the structured programming subject in second-year students of parallel high school “A”, of the Amazonas Educational Unit, during the period February to October 2024. The research route is assumed with a mixed approach, descriptive and propositional scope; Two instruments were applied: a questionnaire and a semi-structured interview, to a sample of 32 students and three teachers. As results, the majority of students expressed that their teachers almost never consider the educational, methodological, interactive, motivational and evaluative dimension; that is, almost never: they provide a solid base of theoretical knowledge and the opportunity to apply it in practical situations. A proposal for a guide titled: “Guide for the use of the “CODING C++” Tool in the teaching-learning process of Structured Programming” which contains six sections, made with a striking design and a constructivist structure. It was concluded that the guide generated enriches the educational experience, simplifies the understanding of concepts and allows students to put into practice what they have learned in real situations, which is crucial for learning programming.

Keywords: “CODING C++” tool, teaching-learning process, structured programming, education.



ÍNDICE GENERAL/FIGURAS/TABLAS/ANEXOS

ÍNDICE GENERAL

FICHA SENESCYT PARA EL REPOSITORIO.....	ii
COPIA INFORME DE SIMILITUD (ANTIPLAGIO).....	iv
CERTIFICACIÓN DE AUTORÍA Y CESIÓN DE DERECHOS DEL AUTOR (ES).....	v
AVAL DEL TUTOR DE LA TESIS.....	vi
DEDICATORIA.....	vii
AGRADECIMIENTO.....	viii
RESUMEN.....	ix
ABSTRACT.....	x
ÍNDICE GENERAL/FIGURAS/TABLAS/ANEXOS.....	xi
ÍNDICE GENERAL.....	xi
ÍNDICE DE TABLAS.....	xv
ÍNDICE DE FIGURAS.....	xvi
LISTADO DE ANEXOS.....	xvii
INTRODUCCIÓN.....	1
Justificación del problema.....	1
Planteamiento del problema.....	3
Precisión del tema.....	4
Objeto de la investigación.....	4
Objetivo general.....	4
Preguntas Científicas.....	4
Declaración de las variables o categorías de la investigación.....	5
Objetivos específicos de la investigación.....	5
Identificación de los métodos a emplear (teóricos, empíricos y matemáticos estadísticos).....	6
Métodos Teóricos.....	6





Métodos Empíricos.....	6
Métodos Matemáticos o Estadísticos	7
Declaración de la población y muestra.....	7
Declaración del tipo de investigación.....	7
Principales aportes.....	8
Importancia, necesidad social, novedad y actualidad científica.....	9
Descripción breve del contenido de los capítulos que integran el informe del trabajo de titulación.....	9
CAPÍTULO 1: MARCO TEÓRICO	11
1.1. Antecedentes.....	11
1.1.1. Antecedentes Internacionales	11
1.1.2. Antecedentes Nacionales.....	13
1.2. Fundamentación Teórica	14
1.2.1. Proceso de enseñanza – aprendizaje de programación estructurada	14
1.2.2. Dimensiones del Proceso de enseñanza – aprendizaje de programación estructurada	15
1.2.3. Herramienta “CODING C++”	24
1.2.4. Evolución de C++ en el tiempo.....	25
1.2.5. Dimensiones de la Herramienta “CODING C++”	26
1.3. Fundamentación tecnológica	30
1.4. Fundamentación psicológica	31
1.5. Fundamentación pedagógica	33
1.6. Fundamentación legal.....	33



CAPÍTULO 2: METODOLOGÍA PARA EL DESARROLLO DE LA INVESTIGACIÓN Y ESTUDIO DIAGNÓSTICO.....	35
2.1. Conceptualización y operacionalización de las variables.....	35
2.1.1. Variable Dependiente: Proceso de enseñanza – aprendizaje de programación estructurada.....	35
2.1.2. Variable Independiente: Herramienta “CODING C++”.	35
2.2. Enfoque de la Investigación.....	38
2.3. Alcance de la investigación	38
2.4. Declaración y justificación del tipo de investigación	38
2.5. Métodos empleados y sus propósitos en el contexto de investigación	39
2.6. Instrumentos derivados de la metodología seleccionada.....	40
2.7. Delimitación de la población y la muestra.....	40
2.8. Estrategia metodológica investigativa o proceder metodológico general seguido en el proceso de investigación de acuerdo con el alcance e intereses de la investigación	41
2.9. La descripción de la metodología de acuerdo con las tareas de investigación se constituye en un apoyo para este apartado del trabajo de titulación. Implica la descripción de las etapas seguidas en el proceso investigativo y su propósito.....	42
Etapas del estudio teórico.....	42
Etapas del diagnóstico inicial	42
Etapas de la modelación de la propuesta.....	42
Etapas del diagnóstico final o validación de la propuesta (teórica o empírica).....	42
2.10. Presentación de los resultados del estudio diagnóstico	43
2.10.1. Conclusiones del diagnóstico inicial	50
CAPÍTULO 3: PRESENTACIÓN Y VALIDACIÓN DE LA PROPUESTA.....	51



3.1. Presentación de la propuesta.....	51
3.1.1. Título de la propuesta	51
3.1.2. Presentación.....	51
3.1.3. Objetivos generales y específicos.....	52
3.1.4. Fundamentación.....	53
3.1.5. Características de la propuesta	54
3.1.6. Estructura del contenido de la guía.....	56
3.1.7. Metodología para su implementación y evaluación	57
3.1.8. Recursos.....	59
3.1.9. Beneficiarios.....	59
3.2. Validación de la propuesta	60
Conclusiones de la validación de la propuesta	70
CONCLUSIONES.....	71
RECOMENDACIONES	73
REFERENCIAS BIBLIOGRÁFICAS	74
ANEXOS	82
TRIBUNAL PROYECTO DE TITULACIÓN	144



ÍNDICE DE TABLAS

Tabla 1 Operacionalización de variables.....	36
Tabla 2. Resultados de la dimensión educativa de la Variable dependiente	43
Tabla 3. Resultados de la dimensión metodológica de la Variable dependiente.....	44
Tabla 4. Resultados de la dimensión interactiva de la Variable dependiente.....	46
Tabla 5. Resultados de la dimensión motivacional de la Variable dependiente.....	47
Tabla 6. Resultados de la dimensión Evaluativa de la Variable dependiente	49
Tabla 7. Respuestas a la interrogante 1 de la entrevista semiestructurada	60
Tabla 8. Respuestas a la interrogante 2 de la entrevista semiestructurada	62
Tabla 9. Respuestas a la interrogante 3 de la entrevista semiestructurada	64
Tabla 10. Respuestas a la interrogante 4 de la entrevista semiestructurada	66
Tabla 11. Respuestas a la interrogante 5 de la entrevista semiestructurada.....	68



ÍNDICE DE FIGURAS

Figura 1. Resultados de la dimensión educativa de la Variable dependiente.....	43
Figura 2. Resultados de la dimensión metodológica de la Variable dependiente	45
Figura 3. Resultados de la dimensión interactiva de la Variable dependiente	46
Figura 4. Resultados de la dimensión motivacional de la Variable dependiente	48
Figura 5. Resultados de la dimensión evaluativa de la Variable dependiente.....	49





LISTADO DE ANEXOS

Anexo 1. Cuestionario para estudiantes	82
Anexo 2. Entrevista Semiestructurada para docentes.....	83
Anexo 3. Validación de los Instrumentos.....	84
Anexo 4. Guía para el uso de la Herramienta “CODING C++” en el proceso de enseñanza- aprendizaje de Programación Estructurada	109





INTRODUCCIÓN

Justificación del problema

Contexto Actual

La sociedad contemporánea se encuentra inmersa en la era digital, donde las tecnologías de la información y comunicación (TIC) , son cruciales en el entramado complejo de la cotidianidad, incluyendo el educativo (Pascagaza & Estrada, 2020), por lo que es imperativo que las metodologías educativas evolucionen hacia la combinación de herramientas digitales para responder a las demandas de estudiantes nativos digitales, quienes en teoría poseen habilidades avanzadas para emplear las TIC, en la esfera personal y en la educativa.

Los procesos educativos están evolucionando, y los docentes poco a poco se han enfrentado con la ineludible necesidad de hacer lo posible para adquirir competencias digitales, que les permitirán perfeccionar las tecnologías de enseñanza–aprendizaje, la usanza de materiales y recursos novedosos, el esmero por maneras de enseñar y aprender novedosas, e innovadoras metodologías didácticas, tal como refieren Nivelá et al. (2021). En resumen, el educador debe ser capaz de adquirir nuevas habilidades pedagógicas y digitales que respondan a las solicitudes de una sociedad coetánea.

Contexto educativo

En la Unidad Educativa Amazonas, el proceso de enseñanza y aprendizaje de la programación estructurada para los estudiantes de segundo año de bachillerato se lleva a cabo de la siguiente forma:

Los recursos que el docente utiliza principalmente son una pizarra y marcadores para explicar conceptos y escribir código, conjuntamente con libros y manuales sobre programación estructurada. Estos recursos son complementados con ejemplos impresos de código para que los estudiantes los pasen a sus cuadernos y los estudien, y las prácticas en entornos reales de programación se dan en un laboratorio de computación con ordenadores básicos.

El método de enseñanza-aprendizaje más utilizado es el tradicional, porque se centran en clases magistrales dentro del aula o el laboratorio (Idrovo, 2023), donde el profesor explica los conceptos de esta unidad curricular y demuestra ejemplos en la pizarra o con el proyector. Se fomenta la intervención de los aprendices durante la clase o en el laboratorio mediante la



enunciación de interrogantes y respuestas tanto por parte del profesor como de los propios estudiantes.

En cuanto a organización, el profesor establece unidades temáticas en base a su planificación, que cubren conceptos elementales de la asignatura, así como otros aspectos, establece un calendario de clases y talleres prácticos para cubrir cada unidad temática, garantizando así una progresión lógica en el aprendizaje. Los mecanismos de interacción entre profesor y estudiantes se dan principalmente en el aula o laboratorio durante las clases y talleres prácticos.

La evaluación del avance de los aprendices se la realiza mediante exámenes escritos para evaluar el conocimiento teórico y la capacidad para emplear la base conceptual estudiada en la resolución de algoritmos de programación, y además también se evalúa a través de tareas y proyectos. El trabajo del profesor se limita a ser un facilitador del conocimiento, enfocado en transmitir información y valorar el logro estudiantil; su tarea principal es explicar conceptos, proporcionar ejemplos y suscitar la intervención de los aprendices en cada actividad.

Por su parte, el guion del aprendiz se circunscribe a recibir la información ofrecida por su profesor, transcribir ejemplos de código en sus cuadernos y participar en las actividades planteadas. La carencia de orientación sobre métodos activos de aprendizaje utilizando herramientas digitales podría reducir su capacidad para desarrollar habilidades prácticas de programación.

Al observar y analizar todos los elementos del entorno formativo de la U.E. Amazonas, se han evidenciado manifestaciones fácticas que podrían entorpecer un adecuado perfeccionamiento del proceso educativo de la asignatura de programación estructurada en el segundo año de bachillerato; lo cual puede verse afectado por componentes como una carencia en la innovación de metodologías para la instrucción y la insuficiencia de competitividades digitales en los profesores; y, limitaciones por parte del currículo. Con relación a la carente innovación del proceso de enseñanza-aprendizaje, conlleva a que el método descrito sigue un enfoque tradicional basado en clases teóricas y prácticas en aulas y el laboratorio, lo que indica según Alcívar & Lemos (2024) una falta de adopción de métodos educativos más innovadores y enfocados en la usanza de herramientas o entornos digitales.



Ahora bien, aunque los docentes del área técnica conocen de entornos de programación, y algunas clases se efectúan en el laboratorio con ordenadores básicos, se aprecia que no integran herramientas digitales en su enseñanza, lo que sugiere según Machuca et al. (2023) una posible falta de habilidades o competencias digitales para maximizar el lucro de las bondades prestadas por la tecnología para la instrucción y la lucubración.

Con relación a las limitaciones por parte del currículo, se utilizan micro-currículos diseñados a partir del FIP y el EGC, un problema que se genera se vincula con unos contenidos complejos y de nivel avanzado y generalizado, esto conlleva a las instituciones que ofrecen carreras técnicas como es el caso de la Unidad Educativa Amazonas que oferta informática, a contar con excelente infraestructura tecnológica, lo cual en la realidad no se cumple, aunque en el PEI se propone como meta que al 2025 el 50% de los docentes implementen herramientas digitales en sus currículos, será imposible sin el presupuesto necesario para el mejoramiento de espacios tecnológicos en la institución (Acción A2 del PEI). Al no contar con equipos informáticos adecuados, conexión a internet o software especializado que admitan un buen progreso de sus actividades, los docentes del área técnica continuarán implementando métodos tradicionales en la praxis de las asignaturas técnicas.

En base a los contextos descritos se aprecia que la sistemática empleada en enseñanza-aprendizaje necesita ser perfeccionada, dado que, al ser un enfoque tradicional, no ofrece a los estudiantes la flexibilidad de estudio en cuanto a horario y lugar, ni la ubicuidad que necesitan para seguir aprendiendo externamente a las aulas, añadido a esto que la institución no posee una robusta infraestructura tecnológica para la instrucción de esta asignatura, lo que requiere la intervención para instituir un procedimiento con acciones que sirva para implementar una herramienta digital que optimice el proceso educativo de la asignatura de programación estructurada con los aprendices del 2do año de bachillerato en la Unidad Educativa Amazonas.

Planteamiento del problema

El problema a resolver se centra en la mejora del proceso de enseñanza-aprendizaje de programación estructurada en el segundo año de bachillerato de la Unidad Educativa Amazonas, mismo que se utiliza bajo un enfoque pedagógico tradicional, y carece de integración de herramientas digitales y flexibilidad para contemplar las insuficiencias de lucubración ubicua de



los estudiantes, dando como resultado bajo rendimiento académico, menoscabo de la motivación y carente interés de los aprendices, aunado a algunos asuntos como la deserción.

Bajo este contexto se analiza la implementación de la herramienta digital “CODING C++” para mejorar el proceso de enseñanza-aprendizaje de programación estructurada en los estudiantes de segundo de bachillerato, y surge la siguiente pregunta:

¿Cuál sería la incidencia que tendría el uso de la herramienta digital “CODING C++” en el mejoramiento del proceso de enseñanza-aprendizaje de la asignatura de programación estructurada en los estudiantes de segundo de bachillerato de la Unidad Educativa Amazonas durante el periodo febrero a octubre de 2024?

Precisión del tema

Herramienta “CODING C++” y su incidencia en el proceso de enseñanza-aprendizaje de programación estructurada en estudiantes de segundo de bachillerato.

Objeto de la investigación

El proceso de enseñanza-aprendizaje de la asignatura de programación estructurada en el segundo año de bachillerato de la Unidad Educativa Amazonas.

Objetivo general

Generar una propuesta para el uso de la herramienta “CODING C++” en el mejoramiento del proceso de enseñanza-aprendizaje de la asignatura de programación estructurada en los estudiantes de segundo de bachillerato paralelo “A”, de la Unidad Educativa Amazonas, durante el periodo febrero a octubre de 2024.

Preguntas Científicas

FASE 1: Fundamentación

1. ¿Cuáles son los fundamentos teóricos que sustentan el uso de la herramienta “CODING C++” en el proceso de enseñanza-aprendizaje de programación estructurada?

FASE 2: Exploración

2. ¿Cuáles son las características del proceso de enseñanza-aprendizaje de la asignatura de programación estructurada en los estudiantes de segundo de bachillerato paralelo “A” de la Unidad Educativa Amazonas, durante el periodo de febrero a octubre de 2024, ¿antes de la implementación de la herramienta “CODING C++”?



FASE 3: Definición

3. ¿Cómo es una propuesta de una guía que permita el uso de la herramienta “CODING C++” en el mejoramiento del proceso de enseñanza-aprendizaje de la asignatura de programación estructurada en los estudiantes de segundo de bachillerato paralelo “A”, de la Unidad Educativa Amazonas, durante el periodo febrero a octubre de 2024?

FASE 4: Validación

4. ¿Es válida la propuesta diseñada para el uso de la herramienta “CODING C++” en el mejoramiento del proceso de enseñanza-aprendizaje de la asignatura de programación estructurada en los estudiantes de segundo de bachillerato paralelo “A”, de la Unidad Educativa Amazonas, durante el periodo febrero a octubre de 2024?

Declaración de las variables o categorías de la investigación

Variable Dependiente: El Proceso de enseñanza – aprendizaje de programación estructurada.

Variable Independiente: Herramienta “CODING C++”.

Objetivos específicos de la investigación

FASE 1: Fundamentación

1. Determinar los fundamentos teóricos que sustentan el uso de la herramienta “CODING C++” en el proceso de enseñanza-aprendizaje de programación estructurada. (Marco teórico)

FASE 2: Exploración

2. Caracterizar el proceso de enseñanza-aprendizaje de la asignatura de programación estructurada en los estudiantes de segundo de bachillerato paralelo “A” de la Unidad Educativa Amazonas, durante el periodo de febrero a octubre de 2024, antes de la implementación de la herramienta “CODING C++”. (Diagnóstico)

FASE 3: Definición

3. Diseñar una propuesta de una guía que permita el uso de la herramienta “CODING C++” en el mejoramiento del proceso de enseñanza-aprendizaje de la asignatura de programación estructurada en los estudiantes de segundo de bachillerato paralelo “A”, de la Unidad Educativa Amazonas, durante el periodo febrero a octubre de 2024. **(Propuesta)**

FASE 4: Validación





4. Validar la propuesta diseñada para el uso de la herramienta “CODING C++” en el mejoramiento del proceso de enseñanza-aprendizaje de la asignatura de programación estructurada en los estudiantes de segundo de bachillerato paralelo “A”, de la Unidad Educativa Amazonas, durante el periodo febrero a octubre de 2024. (**Validación**)

Identificación de los métodos a emplear (teóricos, empíricos y matemáticos estadísticos)

Métodos Teóricos

Análisis y síntesis: Este método se utilizará en la FASE 1: Fundamentación, específicamente al determinar los fundamentos teóricos que respaldan la utilización de la herramienta “CODING C++” en el proceso de enseñanza-aprendizaje de programación estructurada. Se analizarán diferentes teorías y se sintetizarán los conceptos relevantes para la investigación.

Inducción y Deducción: Ambos se aplicarán en la FASE 1: Fundamentación, durante el proceso de determinar los fundamentos teóricos. La inducción se utilizará para extraer conclusiones generales a partir de casos específicos, mientras que la deducción se empleará para aplicar principios generales a casos específicos.

Enfoque de Sistema: Se aplicará principalmente en la FASE 1: Fundamentación, durante la determinación de los fundamentos teóricos. Ayuda a entender cómo las diferentes partes del sistema educativo se interrelacionan.

Revisión bibliográfica y análisis de documentos: Estos métodos se utilizarían también en la FASE 1: Fundamentación, como parte de la investigación teórica. Se revisarán estudios previos, libros, artículos y otros documentos relevantes para recopilar información sobre el uso de herramientas digitales en la enseñanza de programación estructurada.

Métodos Empíricos

Encuesta. Se aplicarán tanto en la FASE 2: Exploración como en la FASE 4: Validación. En la fase 2, se utilizará para recopilar información sobre las características del proceso de enseñanza-aprendizaje de la asignatura de programación estructurada en los estudiantes de segundo de bachillerato paralelo “A” de la Unidad Educativa Amazonas, durante el periodo de febrero a octubre de 2024, antes de la implementación de la herramienta “CODING C++”. En la fase 4, se utilizará para evaluar la percepción de los docentes sobre la incidencia de la herramienta "CODING C++" en el proceso de enseñanza – aprendizaje.



Entrevista.- Se aplicará en la FASE 4: Validación, para obtener información detallada sobre la experiencia de los profesores de la asignatura de programación estructurada con la propuesta diseñada para el uso de la herramienta “CODING C++” en el mejoramiento del proceso de enseñanza-aprendizaje de la asignatura. Las entrevistas permitirán explorar percepciones, dificultades y sugerencias para mejorar tanto la propuesta como el proceso de enseñanza-aprendizaje en estudio.

Métodos Matemáticos o Estadísticos

Análisis Estadísticos. Se utilizarán en las FASES 2 y 4: para obtener información detallada sobre la experiencia de los estudiantes y profesores con la asignatura de programación estructurada con métodos tradicionales; y para la validación de la propuesta. Los análisis estadísticos proporcionarán evidencia objetiva de la incidencia de la herramienta.

Tabulación. La tabulación ayuda a ostentar los resultados organizados sistemáticamente.

Declaración de la población y muestra

La investigación se realizará en la Unidad Educativa Amazonas con los estudiantes de segundo año de bachillerato técnico, que representa una población de 99 estudiantes, divididos en 3 paralelos. Además de tres (03) docentes que administran esta asignatura.

La muestra seleccionada de manera intencional para esta investigación va a estar conformada por 32 estudiantes del segundo año de bachillerato paralelo “A”. La selección de la muestra será no probabilística por conveniencia dado que los participantes son seleccionados de manera conveniente y accesible para los investigadores, además de que los recursos como el tiempo y el dinero son limitados.

Declaración del tipo de investigación

La investigación propuesta se alinea claramente con los fundamentos de la investigación descriptiva y propositiva, es descriptiva ya que se proporciona una representación minuciosa del contexto y las operaciones imprescindibles para la culminación célebre de la herramienta digital. Específicamente porque se caracteriza el proceso de enseñanza-aprendizaje de la asignatura de programación estructurada en los estudiantes de segundo de bachillerato paralelo “A” de la Unidad Educativa Amazonas, durante el periodo de febrero a octubre de 2024, anterior a la aplicación de la herramienta “CODING C++”.



Es propositiva ya que busca aplicar el conocimiento existente para mejorar un proceso educativo práctico en colaboración con una institución educativa, teniendo influencia inmediata en la eficacia educativa en la U.E. Amazonas, además de poder servir como modelo para otras instituciones educativas que enfrenten desafíos similares en el proceso educativo de la asignatura objeto de estudio.

Paradigma interpretativo con enfoque mixto

Se usará para una perspicacia insondable y minuciosa de las experiencias, percepciones y contextos sociales relacionados con el uso de la herramienta "CODING C++" en el proceso de enseñanza-aprendizaje de la programación estructurada, y gracias al enfoque mixto es posible combinar las fortalezas de los métodos cualitativos y cuantitativos para validar y complementar mutuamente los hallazgos.

Corte transversal

Los datos se recopilan en un solo momento en el tiempo, lo que permite comparar diferentes grupos o poblaciones en un momento dado, en el caso de ésta propuesta la investigación se centra durante el periodo de febrero a octubre de 2024, tiempo en el que se debe recolectar, tabular, y analizar los datos obtenidos de los estudiantes de segundo de bachillerato paralelo "A", y validar la propuesta de la implementación de la herramienta "CODING C++" en el proceso de enseñanza-aprendizaje de programación durante un período académico específico.

Principales aportes

Desde su aplicación práctica

Implementación de la herramienta "CODING C++" en el proceso de enseñanza-aprendizaje de la asignatura de programación estructurada.

Perfeccionamiento en el cometido erudito de los aprendices de 2do de bachillerato paralelo "A" en la Unidad Educativa Amazonas.

Desde su alcance teórico

Generación de evidencia empírica que respalde el empuje de la ejecución de la herramienta "CODING C++" en el proceso de enseñanza-aprendizaje de programación estructurada.

Caracterización y estudio de prerrogativas y retos en la ejecución de herramientas digitales en el contexto educativo, específicamente en la programación.





Importancia, necesidad social, novedad y actualidad científica.

Importancia

Contribuirá al perfeccionamiento de la eficacia educativa en la Unidad Educativa Amazonas, al proporcionar una metodología innovadora y positiva para la instrucción de programación. La investigación ayudará a identificar y abordar las manifestaciones fácticas en el proceso de enseñanza-aprendizaje de programación estructurada, con el fin de optimizar el provecho erudito de los aprendices. Las derivaciones logradas podrían tener implicaciones más amplias en el ámbito educativo, sirviendo como referencia para otras instituciones educativas interesadas en mejorar la enseñanza de la programación a través del uso de herramientas digitales.

Necesidad social

En este contexto cada día mayormente informatizado, es crucial que los aprendices desplieguen pericias tecnológicas oportunas. En este caso, la utilización de la herramienta "CODING C++" consigue ayudar a desplegar pericias de programación y pensamiento lógico, las cuales cada día son más exigidas.

Novedad y actualidad científica

Esta investigación constituye uno de los primeros intentos en la Unidad Educativa Amazonas de integrar una herramienta digital específica para la enseñanza de la programación estructurada, lo que supone una novedad en el enfoque pedagógico tradicional de la asignatura.

Descripción breve del contenido de los capítulos que integran el informe del trabajo de titulación

El presente proyecto científico se desarrollará por capítulos:

Introducción. Este apartado trata sobre el problema, su planteamiento, contextualización, el análisis crítico, la formulación, las interrogantes, se delimita el objeto de investigación, la justificación y los objetivos, de igual manera las categorías, la importancia, necesidad y novedad de esta investigación y sus principales aportes al contexto científico.

Capítulo I. En éste se señalan los antecedentes investigativos, las fundamentaciones correspondientes, se aborda el marco teórico en base a las categorías fundamentales, los antecedentes históricos del problema tratado, soluciones y aportes de otros autores, el enfoque teórico-conceptual asumido para el tratamiento del problema.



Capítulo II. En éste se presenta la metodología de investigación utilizada, se determina la población y la muestra, así como la operacionalización de variables, técnicas y procedimientos utilizados para recolectar y procesar información.

Capítulo III. En éste se presenta la propuesta y su validación (teórica o empírica), así como el análisis cualitativo y cuantitativo de resultados e interpretarán de los mismos.

Conclusiones. Se plasman las conclusiones que se originen, con base en los objetivos específicos definidos para este estudio.

Recomendaciones. Se plasman las recomendaciones de operaciones a desarrollar en lo futuro, además del establecimiento de la propuesta surgida de la investigación realizada para su posterior utilización en la institución educativa escogida para el estudio.



CAPÍTULO 1: MARCO TEÓRICO

Este marco teórico suministra una fundación consistente para vislumbrar y contextualizar el análisis de la Herramienta “CODING C++” y su incidencia en el proceso de enseñanza-aprendizaje de programación estructurada en estudiantes de segundo de bachillerato, permitiendo así abordar la investigación de manera informada y fundamentada teóricamente; además de sus fundamentaciones tecnológica, psicológica, pedagógica y legal.

1.1. Antecedentes

1.1.1. Antecedentes Internacionales

En el contexto de México, Rodríguez et al. (2024) realizaron un estudio con la intención de determinar lineamientos para la generación de un software de base de datos (RSE) para el sistema informático de la Universidad privada de Chiapas y confrontar las secuelas de los grupos semestrales. El método usado fue un experimento exploratorio completo con utilización de la prueba ANOVA para determinar si había una diferencia significativa entre las diferentes estaciones. Los datos empíricos mostraron que no hubo diferencias representativas en las instrucciones de generación y adelanto de la RSE de diferentes períodos de la UNACH. Concluyeron que los semblantes de mayor trascendencia de la tecnología y la reconstrucción de tecnología son: tener funciones de almacenamiento, tener funciones de búsqueda y descarga, uso de herramientas basadas en construcción, lenguaje de programación C++ y Python para sistema operativo Windows y MySQL Server, estructura de metadatos. Con registros en CSV y XML, la integración de datos y la eficiencia de búsqueda fueron los principales beneficios.

En Perú, Gallegos (2023) realizó una indagación cuyo objetivo fue establecer las consecuencias de la aplicación de chatgpt en la lucubración del lenguaje de programación C++ entre aprendices de primaria. Según el criterio del investigador, la población y la muestra no fueron posibles, por lo que se cuasiprobaron con dos mediciones antes y después del tratamiento. El instrumento estructurado utilizado consta de cuatro dimensiones, con calificaciones medidas en una escala de nueve puntos dividida en tres categorías: baja (1 a 3), media (3 a 6) y alta (6 a 9). Los resultados generales muestran un progreso en la lucubración de aprendices del nivel más bajo al más alto. Por otro lado, el promedio de horas dedicadas al programa es de $4,05 \pm 1,04$, o 95%). El autor concluyó que la usanza de la inteligencia artificial “asistente de lenguaje natural” en



universidades podría ser una maniobra poderosa para optimizar la lucubración y las pericias en lenguajes de programación.

Otro estudio realizado en Colombia por Abril et al. (2023) tuvo como objetivo desarrollar nuevas maneras de lucubración y tecnología entre aprendices. Consistió en una investigación sobre la programación en la escuela y su incidencia en el progreso del pensamiento examinador y la capacidad para crear de los aprendices. Se utilizó el lenguaje de programación C++ como herramienta de investigación y se creó una facultad para enseñar programación básica. Se aplicó un enfoque de aprendizaje. Concluyeron que la ejecución de una sección académica asumió una incidencia positiva en el interés y la comprensión del programa de los estudiantes. Hubo un aumento notable en la contribución de los aprendices en actividades prácticas y su entusiasmo por probar un vasto universo de la programación.

La investigación desarrollada en Malasia por Sulaiman et al. (2023) refiere que aprender lenguajes de programación es especialmente difícil para aquellos aprendices que no pertenecen a informática porque el tema puede ser desconocido y complejo. Este estudio tuvo como propósito presentar un proyecto para desarrollar la educación basada en juegos en las escuelas primarias, a través de un programa que admite que aprendices instruirse en programación y resolución de problemas mientras juegan. Este proyecto siguió el modelo Game Development Life Cycle (GDLC), que incluye varias etapas. Comienza explicando el concepto básico y la dirección del juego. Luego viene el proceso de preproducción, donde se desarrolla el concepto y se prepara un plan detallado. La fase de producción implica implementar el código del software. Luego viene el episodio. Finalmente, el juego se ofrece al usuario final. El resultado mostró que todas las pruebas sobre el rendimiento del programa fueron exitosas, concluyendo que es posible confirmar que este programa se puede esgrimir como alternativa para instruirse y programar en colación con inventivas habituales.

Otro estudio antecedente realizado en Malasia por Zainol & Ismail (2021) tuvo por objetivo desarrollar un prototipo de juego llamado "Code-E" que permite a los jugadores instruirse y desdoblarse pericias de codificación mientras juegan. Los autores se basaron en la Investigación de Diseño y Desarrollo (DDR), por medio del cual crearon un juego de acción y aventuras en 3D y un juego educativo divertido que los estudiantes o jugadores pudieran usar durante sus sesiones



de grabación, con el tema de programación funcional en C++. En primer lugar, se desarrolló el concepto del juego, se diseñó un prototipo rápido para ayudar a los usuarios a comprender el juego más fácilmente, el concepto original del juego fue bien recibido y requirió poco desarrollo. Luego diseñaron y desarrollaron un prototipo de juego completo usando Roblox. Se aplicó un cuestionario con preguntas Likert de 5 puntos divididos en tres categorías. Los resultados del estudio mostraron que el prototipo del juego "Code-E" funcionó bien y fue bien recibido por los usuarios potenciales. Concluyen que se necesitan algunas mejoras adicionales para aumentar la capacidad en términos de complejidad y motivación.

El estudio realizado en Filipinas por Cu Hao et al. (2021) comunica derivaciones de experiencias con C++ Adventure, un juego digital diseñado para apoyar y motivar la instrucción y lucubración de codificación. El proyecto de investigación se apoyó en las posibilidades de la lucubración fundamentada en juegos para complementa las habilidades cognitivas del aprendizaje multimedia y el método de diseño ARCS. Para examinar la muestra, un equipo con 20 aprendices de 9° y 10° grado participó en la evaluación de estudiantes. La evaluación encontró que la mayoría estaba satisfecha con las imágenes, la usabilidad y la interactividad del juego, pero también hubo mejoras en el boceto y exposición de las lecciones. También resultó que la motivación para usar y aprender el software era diferente entre los dos grupos. Finalmente, se hicieron sugerencias futuras con respecto al proyecto e investigaciones similares.

1.1.2. Antecedentes Nacionales

El estudio realizado por Tumbaco et al. (2023) en Ecuador tuvo por objetivo desarrollar un método educativo para introducción básica del lenguaje C++ a través del sitio web Educ ALIM para estudiantes de primer semestre del Departamento de Comunicación de la Universidad Estatal Península de Santa Elena (UPSE). Participó una muestra de 60 estudiantes. Utilizaron una investigación descriptiva con técnicas como la observación y la encuesta. Realizar una encuesta de satisfacción estudiantil, con esa información diseñaron una unidad de trabajo con el Lenguaje de programación C++, y acciones esperadas de los aprendizajes, utilizando recursos de información digital como video conferencias, Moodle. Concluyen que la enseñanza del lenguaje de programación C++ es muy compleja porque hay temas que no se comprenden y algunos son



confusos, no son claros, como el uso de las variables, las funciones los struct, arreglos, y almacenamientos de base de datos en relación a la herramienta Educalim.

Otro estudio realizado en Ecuador por Proaño & Trujillo (2021) tuvo como propósito comparar tres métodos; a saber, el curso de posgrado, el curso de posgrado con C++ y Matlab, en requisitos para la utilidad del curso. Para valorar la incidencia de esta actividad se crearon grupos de investigación gestionados simultáneamente. Mientras que los aprendices integrantes del grupo experimental estuvieron entrenados usando código C++ para demostrar la estructura interna del producto, el grupo de control fue entrenado usando código creado por Matlab. Ambos grupos utilizaron evaluaciones, y concluyen que el estudio del DCA encontró que una manera eficiente de instruir espacios vectoriales es la que involucra clases básicas y código C++.

1.2. Fundamentación Teórica

1.2.1. Proceso de enseñanza – aprendizaje de programación estructurada

Este es esencial para una instrucción de los aprendices en informática, esta asignatura facilita los elementos ineludibles para alcanzar y emplear conceptos esenciales de la programación, sentando los fundamentos para el perfeccionamiento de pericias técnicas y lógicas tal como refieren Mayeta et al. (2020). Hace énfasis en resolución de problemas, esgrimiendo una dirección organizada y estructurada (Parra, 2022); a través de esta asignatura, los estudiantes aprenden a desarreglar una dificultad compleja en fragmentos más diminutos y dóciles, empleando técnicas como el diseño de algoritmos y empleando estructuras de control.

Entre los talentos más sustanciales del proceso educativo de esta asignatura está la perspicacia de concepciones elementales (Pucuji, 2022). Los estudiantes deben adquirir conocimientos sólidos sobre variables, tipos de datos, operadores y estructuras de control, como bucles y condicionales; estos conceptos forman la base sobre la cual los estudiantes podrán construir programas más complejos en el futuro.

Adicional a los conceptos teóricos, esta asignatura enfatiza la experiencia y solución de dificultades reales, donde los aprendices disfrutan la posibilidad de emplear las sapiencias alcanzadas por la consumación de programas, esgrimiendo un lenguaje de programación determinado; dicha práctica les permite experimentar de primera mano la manera de aplicación de conceptos teóricos en la generación de programas funcionales.



La enseñanza-aprendizaje de programación estructurada igualmente provoca el pensamiento lógico y analítico (Calzadilla, 2022), lo cual sucede a medida que enfrentan problemas y desafíos de programación, en los cuales deben ser capaces de analizar y comprender las instrucciones necesarias para llegar a una solución. Este aforo del pensamiento crítico es básico en el contexto de la programación y también resulta valiosa en otras disciplinas.

Para garantizar la efectividad del proceso de enseñanza-aprendizaje de la asignatura estudiada, es sustancial poseer una planta de profesores formados y actualizados en modernas tendencias y técnicas de instrucción (Zoila, 2023). Los docentes deben ser capaces de transmitir los conceptos claramente y utilizar estrategias didácticas que animen la intervención de los aprendices; esto mediante, la aplicación de recursos y herramientas tecnológicas capaces de perfeccionar la rutina de lucubración al suministrar modelos interactivos y oportunidades de práctica.

1.2.2. Dimensiones del Proceso de enseñanza – aprendizaje de programación estructurada

Se caracteriza por cinco dimensiones; a saber, la dimensión educativa, la metodológica, la interactiva, la motivacional, y la evaluativa.

Dimensión Educativa

La dimensión hace referencia a la instrucción de los aprendices en aspectos teóricos y prácticos vinculados con la asignatura. Proporcionando a los estudiantes una consistente fundamentación teórica y la posibilidad de emplear estas sapiencias en circunstancias prácticas, se les da formación dirigida a afrontar retos cotidianos y desarrollar habilidades importantes para su futuro (Pérez, 2024). Además, esta dimensión busca promover el pensamiento crítico, solución de inconvenientes y creatividad, aspectos esenciales para la evolución erudita de los aprendices. Entre sus indicadores se incluyen la adquisición y aprendizaje de conceptos fundamentales, el perfeccionamiento de capacidades técnicas y la sagacidad de los compendios de la programación. Con respecto al primer indicador: la adquisición y el aprendizaje de conceptos fundamentales, son considerados como elementos cruciales en el proceso educativo; entendiendo a los conceptos fundamentales según refieren Moreno et al. (2024) como aquellos conocimientos esenciales que fundamentan la lucubración y comprensión de otros conceptos más complejos; estos, suelen ser las raíces sobre las que emerge el discernimiento en disímiles áreas.



También se refiere según Gutiérrez et al. (2024) a la capacidad de los aprendices para obtener una comprensión sólida y significativa de estos conceptos; lo cual implica no solo memorizarlos, sino también, comprender su significado, aplicarlos en diferentes contextos y relacionarlos con otros conceptos; así al adquirir conceptos fundamentales de forma positiva, los aprendices desarrollan una base sólida que les permite avanzar en su aprendizaje de manera más efectiva.

Por su parte, la lucubración de conceptos fundamentales se refiere al proceso activo de construcción del conocimiento (Sosa, 2022), logrando mediante una instrucción adecuada, que los aprendices consigan examinar, experimentar y reflexionar sobre estos; lo cual involucra la participación del aprendiz, donde se aviva la solución de dificultades, la reflexión y aplicación práctica de lo estudiado.

La adquisición y el aprendizaje de conceptos fundamentales son fundamentales por diferentes causas; en principio, tal como refieren Suárez et al. (2024) brindan a los estudiantes fundamento consolidado para vislumbrar concepciones complejas posteriormente, así, al poseer una perspicacia recóndita de concepciones primordiales, los estudiantes pueden construir sobre ellos y desenvolver una perspicacia extensa y completa de un área determinada.

Como segundo término, según Cárdenas et al. (2022) suministran una moldura de reseña idónea para la criticidad y para solventar inconvenientes; al vislumbrar los elementos primordiales de la disciplina, los estudiantes pueden aplicar su conocimiento en diferentes situaciones, analizar y valorar indagación de modo certero, y adoptar laudos entendidos. Asimismo, fomentan la cabida de los aprendices para efectuar conexiones entre diferentes áreas de conocimiento; al comprender cómo los conceptos se relacionan entre sí, pueden desarrollar una visión más integrada y multidisciplinaria del mundo.

Dimensión metodológica

Esta dimensión se centra en dos indicadores que son las estrategias y los métodos utilizados para enseñar. En el espacio formativo, las estrategias y los métodos utilizados para enseñar circunscriben un rol esencial en el proceso de lucubración de los aprendices. Una estrategia comúnmente más aplicada, considerada tradicional, es el enfoque expositivo, en el cual, según expresan Coapaza et al. (2024), el educador presenta información muy organizada, brindando



explicaciones detalladas y ejemplos relevantes; este es principalmente ventajoso para introducir nuevos conceptos y proporcionar instrucciones elementales.

Otra maniobra positiva es la lucubración basada en dificultades (ABP), la cual es una estrategia didáctica innovadora donde se utilizan dificultades complejas y reales como elemento inicial capaz de suscitar la lucubración de concepciones y principios por los aprendices (Mera & Timana, 2024). A diferencia de la enseñanza tradicional, donde se presentan hechos y conceptos de manera directa, busca que los estudiantes edifiquen su discernimiento mediante la solución de problemas provenientes de situaciones reales.

El ABP se centra de esta manera en el aprendiz como constructor de su lucubración; estos se enfrentan a dificultades desafiantes que demandan esmero de sapiencias y pericias en un entorno genuino; a medida que trabajan en la solución de estos inconvenientes, los aprendices despliegan pericias para la criticidad, la solución de problemas, el compromiso en unidad, comunicación y toma de decisiones.

Una de las peculiaridades más apreciables del ABP es su manera para desarrollar el nivel de criticidad, tal como plantea Quispe (2021); esto es, porque al enfrentarse a problemas complejos, los estudiantes deben examinar, valorar y esquematizar pesquisa, identificar posibles soluciones y tomar decisiones fundamentadas, permitiéndoles desplegar pericias de tendencia crítica las cuales son fundamentales en su progreso erudito y particular.

El aprendizaje cooperativo es otra estrategia valiosa, en la cual según postulan Chimarro et al. (2023) los aprendices realizan las actividades en pequeños equipos para alcanzar metas habituales, fomentando la asistencia, comunicación segura y la permuta de opiniones, promoviendo una lucubración activa y característica. Esta habilidad didáctica suscita la instrucción mediante la socialización; su accionar consiste en fraccionar en grupos de pocos integrantes y diversos, en los cuales los aprendices laboran de forma conjunta para solventar tareas académicas.

Esta metodología fomenta la cooperación estudiantil, permitiéndoles ayudarse mutuamente a superar dificultades y entender conceptos complejos (Vallejo, 2023); además, promueve el compromiso particular y colectivo, ya que los aprendices se lían con tareas y proyectos grupales.



Tiene diversos beneficios; entre ellos, accede a los aprendices exponerse a disímiles perspectivas y enriquecer su comprensión sobre un tema.

Al ocuparse de manera grupal, los aprendices pueden recapacitar sobre sus incertidumbres y obtener conclusiones utilizando los métodos disponibles, también aviva el perfeccionamiento de pericias sociales, como la comunicación, el desempeño en equipo y la solución de conflictos (Padín, 2023). Además, el aprendizaje cooperativo asume una incidencia positiva en beneficio académico (Medina, 2021); al realizar actividades grupales, los aprendices logran intervenir conocimientos, habilidades y estrategias de aprendizaje, lo que facilita la ganancia de otras sapiencias; también, suscita la exaltación y el adeudo con el proceso de lucubración.

En cuanto a los métodos aplicados, la utilización de recursos multimedia es mucho más usado; estos métodos visuales y auditivos permiten atraer el esmero de los aprendices y proporcionan la perspicacia de concepciones complejas (Centeno, 2023). La usanza tecnológica también ha sublevado la manera de la instrucción, con la incorporación de herramientas como pizarras digitales, aplicaciones móviles y plataformas de aprendizaje en línea; estos materiales proveen a los aprendices acceder a una gran variedad de recursos y permiten una lucubración personificada y autónoma. Por ejemplo, al incorporar herramientas tecnológicas como "CODING C++", se brinda a los aprendices muchas posibilidades para su lucubración; esta herramienta específica permite a los alumnos instruirse y ejecutar la programación en el lenguaje C++, que es primordial para la asignatura programación estructurada.

Dimensión Interactiva

Esta dimensión se relaciona con la interrelación entre el profesor y el aprendiz durante la enseñanza-aprendizaje. Sus indicadores incluyen la comunicación efectiva, el apoyo individualizado, la retroalimentación constructiva y el fomento de la intervención del aprendiz. La comunicación efectiva según establece Neira (2024), es un elemento clave que permite establecer conexiones significativas con los demás, transmitir ideas y sentimientos, resolver conflictos y lograr objetivos comunes; de esta forma, se reconcilia con una pericia fundamental que ayuda a relacionarse de manera exitosa en todos los espacios vitales, ya sea personal, profesional o social.



Envuelve el aforo de enunciar opiniones de forma concisa, para lo cual, es trascendental considerar aspectos tales como, proporcionar vigilancia a la comunicación no verbal, la cual incluye otros elementos distintos al acto del habla que pueden influir en cómo se percibe el mensaje y pueden transmitir emociones y actitudes. En segundo lugar, es esencial escuchar activamente a los demás; esto conlleva prestar atención completa a lo dicho por las demás, comprendiendo su punto de vista y mostrando interés genuino.

La comunicación efectiva implica la capacidad de esgrimir una locución perceptible; evitar el uso de jergas o tecnicismos innecesarios y adaptar el lenguaje al nivel de comprensión del interlocutor es esencial para certificar que la recomendación sea entendida correctamente (Tascón & Montolío, 2020). Por último, involucra ser conscientes de las emociones y manejarlas de manera adecuada durante la interacción (Preciado, 2024); esto implica controlar las reacciones emocionales y expresarlas de manera constructiva, evitando respuestas impulsivas o agresivas que puedan dañar la relación con los demás.

Por su parte, el apoyo individualizado, la retroalimentación constructiva y la promoción de la participación activa del estudiante son elementos fundamentales en el proceso de enseñanza-aprendizaje, tal como refieren Cruz et al. (2023); Estas prácticas pedagógicas tienen como objetivo principal atender las necesidades individuales de cada estudiante, brindarles orientación y motivarlos a participar activamente en su propio proceso de aprendizaje.

El apoyo individualizado involucra la importancia de reconocer las fortalezas e impotencias de cada aprendiz para adecuar las estrategias de enseñanza y satisfacer sus necesidades específicas (Anchundia & Leal, 2023), esto implica brindar recursos adicionales, actividades o ejercicios complementarios, y proporcionar asistencia y orientación personalizada cuando sea necesario. Al ofrecer este apoyo individualizado, se fomenta el desarrollo del potencial de cada estudiante y se les brinda la oportunidad de alcanzar el éxito académico.

Por su parte, la retroalimentación productiva es básica para la evolución y lucubración de los aprendices (García, 2024); se trata de proporcionar comentarios específicos, descriptivos y orientados al beneficio de los objetivos de lucubración, se enfoca en resaltar los puntos fuertes del estudiante y brindar sugerencias y recomendaciones para mejorar; así, al recibir una



retroalimentación de calidad, los aprendices pueden vislumbrar con mayor énfasis sus áreas de perfeccionamiento y desplegar destrezas para tocarlas.

El fomento de la intervención del aprendiz es primordial para una lucubración duradera (Morante & Ramírez, 2024); así, al envolver a los aprendices en acciones interactivas, debates, proyectos grupales y otras formas de participación activa, se les motiva a adjudicarse un rol participativo en el logro de su lucubración; todas estas actividades suscitan la criticidad, creatividad, asistencia y emancipación; además, al provocar la cooperación, se genera un entorno de lucubración dinámica y enriquecedora, en la cual los aprendices se conciben estimados y avivados para intervenir con sus ideales y perspectivas.

Dimensión motivacional

Esta dimensión se refiere a la motivación e interés del aprendiz por aprender programación estructurada. Incluye indicadores como la generación de un entorno de aprendizaje vivificante, la conexión de los contenidos con situaciones de la vida real y la promoción de la autonomía y la autoeficacia del estudiante.

El diseño de un contexto de lucubración alentador, según refieren Ontoria et al. (2023) conlleva prometer la liberación de examinar opiniones, tomar decisiones y adjudicarse el adeudo de su lucubración; en este sentido, los discípulos deben ser animados a instituir metas, observar su progreso y recapacitar sobre sus logros y posibilidades de mejora. Asimismo, la diversidad y la inclusión son elementos notables en un ambiente de lucubración apremiante; es significativo que todos los aprendices se conciban valorados y respetados, sin importar sus diferencias culturales, lingüísticas o de habilidades.

Se hace necesario generar conformidades para que cada aprendiz se conciba como parte del equipo y logre asistir con sus representaciones individuales, donde la retroalimentación de forma edificante se hace primordial para un entorno de lucubración alentador; es decir, los aprendices deben recibir comentario específicos, y orientados a su crecimiento, de tal manera que les anime a percibir sus baluartes y posibilidades de mejora; esta debe ser oportuna y brindarles herramientas y estrategias para perfeccionar su desempeño.

La generación de un entorno de lucubración alentador es principal para un proceso educativo efectivo (Arboleda, 2024); al promover la curiosidad, intervención, emancipación, pluralidad y



la retroalimentación edificante, se crea un entorno en el que los aprendices se sienten motivados, estimados y afrontados. Un entorno de lucubración inspirador consiente a los aprendices desplegar pericias y sapiencias de forma significativa, preparándolos para enfrentar los retos del mundo y suscitando su evolución académica y particular.

Por su parte, la conexión de los contenidos con situaciones de la vida real es un enfoque pedagógico eficaz que permite a los estudiantes comprender la relevancia y aplicabilidad de lo que están aprendiendo en el contexto de su vida cotidiana (Chaverra, 2023), esto ayuda a generar un mayor interés, motivación y comprensión profunda de los conceptos y habilidades que se están enseñando; en esencia, cuando los contenidos se vinculan con situaciones de la vida real, los estudiantes pueden ver cómo lo que están aprendiendo tiene un impacto directo en su entorno y en su futuro, esto permite hacer conexiones significativas entre la teoría y la práctica, lo que facilita la retención y la transferencia de conocimientos.

Una forma efectiva de conectar los contenidos con situaciones de la vida real es a través del uso de ejemplos y casos prácticos; al presentar ejemplos concretos y relevantes, los estudiantes pueden ver cómo los conceptos y habilidades aprendidos se aplican en situaciones reales; esto les ayuda a comprender la utilidad y la importancia de lo que están aprendiendo, y les brinda la oportunidad de explorar diferentes escenarios y soluciones.

La conexión de los contenidos con situaciones de la realidad también fomenta el desarrollo de habilidades transferibles (Valeriano, 2021); al abordar retos y contrariedades de la realidad, los aprendices pueden desarrollar pericias como la criticidad, la solución de inconvenientes, la comunicación positiva y el trabajo en equipo; estas pericias son fundamentales para el éxito en todos los ámbitos de la vida, no sólo en el académico.

La promoción de la autonomía y la autoeficacia del estudiante son primordiales para su desarrollo integral y éxito académico (Ruiz, 2024). El fomento de la emancipación ofrece a los aprendices el aforo de tomar el compromiso de su lucubración y arrebatarse providencias (Tuárez & Guzmán, 2023); por otro lado, la autoeficacia está referida al dogma en las propias pericias y cabidas para alcanzar metas y desafiar.

Cuando se promueve la emancipación, se les da a los aprendices la oportunidad de explorar sus intereses, tomar decisiones y asumir un rol en su proceso de lucubración; esto implica brindarles



la libertad de elegir métodos de estudio, establecer metas personales y gestionar su tiempo de manera efectiva; al tener el control sobre su propia lucubración, los aprendices se vuelven más motivados, comprometidos y responsables de su progreso académico.

La promoción de la autoeficacia también es fundamental para el progreso del aprendiz; al animar a los aprendices a reconocer y apreciar sus pericias y destrezas, se les suministra la familiaridad necesaria para afrontar retos y resaltar dificultades (Ponce & Baltazar, 2024); esto envuelve ofrecer retroalimentación edificante y animar a instituir metas objetivas y factibles; de esta manera, conforme los aprendices experimentan éxito y ven que sus arrestos dan resultados, su sentido de autoeficacia se fortalece, lo cual tiene una incidencia positiva en su estimulación y persistencia.

Este adelanto tiene cuantiosos beneficios, como por ejemplo el desarrollo de habilidades de pensamiento crítico, toma de decisiones y resolución de problemas, que son fundamentales para su éxito en la educación y más allá; además, al tener la confianza y la autonomía para tomar decisiones, los estudiantes se vuelven más comprometidos y responsables, lo que se traduce en un mayor compromiso y rendimiento académico.

Es importante destacar que la promoción de la autonomía y la autoeficacia no significa dejar a los estudiantes abandonados a su suerte; los educadores desempeñan un papel crucial al brindar orientación, apoyo y retroalimentación constante; sin embargo, es fundamental permitir que los estudiantes tomen decisiones y asuman la responsabilidad de su propio aprendizaje, lo que les permite desarrollar habilidades de autorregulación y convertirse en aprendices independientes y autónomos.

Dimensión Evaluativa

Esta dimensión se relaciona con la evaluación del aprendizaje del estudiante en programación estructurada. Incluye indicadores como la utilización de diferentes tipos de evaluación, como pruebas, proyectos y trabajos prácticos, para medir el nivel de comprensión y aplicación de los conceptos por parte del estudiante.

El nivel de comprensión, es un aspecto clave en el proceso de aprendizaje y adquisición de conocimientos; se refiere a la capacidad de comprender y asimilar la información de manera significativa y profunda (Velasco & Graus, 2024). Comprende ir más allá de la simple



memorización de hechos y conceptos, hacia la capacidad de interpretar, analizar y relacionar la información de manera crítica y reflexiva. La comprensión se construye a través de la interacción con la información y el uso de estrategias cognitivas, esto involucra no solo leer o escuchar, sino también reflexionar, hacer preguntas, relacionar la información con conocimientos previos y buscar significado en lo que se está aprendiendo.

Un alto nivel de comprensión implica la capacidad de sintetizar información, identificar patrones y conexiones, y aplicar el conocimiento en diferentes contextos (Dávila, 2024); esto involucra utilizar el conocimiento de manera activa, resolver problemas, participar en discusiones y aplicar conceptos en diferentes contextos; al hacerlo, los estudiantes profundizan su comprensión y desarrollan habilidades de pensamiento crítico y análisis. Como expresa Zurita (2020), es importante destacar que el nivel de comprensión puede variar según el tema, la experiencia previa y las habilidades individuales de cada estudiante. Al respecto, se puede argumentar que algunos conceptos pueden ser más complejos y requerir un mayor nivel de comprensión, mientras que otros pueden ser más accesibles.

Por su parte, según refieren Oseda et al. (2020), la aplicación de los conceptos por parte del estudiante es un componente fundamental en el proceso de aprendizaje; no basta con simplemente comprender los conceptos teóricos, sino que es necesario poder aplicarlos en situaciones prácticas y reales; lo cual implica la capacidad de utilizar el conocimiento adquirido de manera activa y creativa; esto conlleva resolver problemas, tomar decisiones informadas y utilizar los conceptos y principios aprendidos para enfrentar desafíos del mundo real.

Al aplicar los conceptos en situaciones prácticas, el estudiante puede evaluar si realmente ha asimilado los conocimientos y si puede utilizarlos de manera efectiva (Puig, 2020). Además, se fomenta el pensamiento crítico y el razonamiento; al enfrentar situaciones reales que requieren la aplicación de los conceptos, el estudiante debe analizar, evaluar y tomar decisiones basadas en su comprensión de los mismos; esto promueve el desarrollo de habilidades de análisis, resolución de problemas y toma de decisiones informadas.

Los docentes desempeñan un papel concluyente en la promoción de la aplicación de los conceptos por parte de los estudiantes, ya que ellos pueden diseñar actividades y proyectos que requieran la utilización de los conceptos aprendidos, brindar retroalimentación y guiar a los



estudiantes en el proceso de aplicación (Mullo, 2023). Esto enfatiza el rol que los docentes tienen con respecto a la responsabilidad de crear un entorno de aprendizaje en el que los aprendices puedan poner en práctica lo que han aprendido y desarrollar habilidades prácticas.

El nivel de habilidad requerido para enfrentar desafíos de programación y resolver problemas en el contexto de "CODING C++" involucra la capacidad de trabajar de forma autónoma en nuevos desafíos, establecer metas propias para medir el progreso, liderar iniciativas y llevarlas a cabo con poca o ninguna supervisión, planificar procesos, evaluar su efectividad y considerar alternativas y posibles resultados.

Además, se espera que el aprendiz sea completamente autónomo, sea considerada líder y experta en su dominio, y tenga la capacidad de seleccionar proyectos que permitan crear un impacto significativo en el campo y ofrecer nuevas ideas o prácticas; este nivel de habilidad se destaca en un entorno estructurado con supervisión, procesos predefinidos y criterios establecidos para evaluar la producción.

1.2.3. Herramienta "CODING C++"

El lenguaje de programación C++ ha sido una herramienta fundamental en el mundo de la programación durante décadas. Conocido por su versatilidad y potencia, C++ ha sido utilizado en una amplia gama de aplicaciones, desde sistemas operativos hasta videojuegos de alto rendimiento (Costanzo, 2024).

Una de las principales ventajas del C++ es su capacidad para realizar programación de bajo nivel (De Parga, 2021). A diferencia de otros lenguajes de programación más modernos, según refieren Martín et al. (2021) C++ permite un control preciso sobre los recursos del sistema, como la memoria y los registros de la CPU. Esto ha hecho que C++ sea la opción preferida para el desarrollo de aplicaciones que requieren un alto rendimiento y un uso eficiente de los recursos, como los videojuegos y los sistemas embebidos.

Otro aspecto destacado del C++ es su soporte para la programación orientada a objetos, el cual permite la creación de programas más organizados y modulares, facilitando el desarrollo y el mantenimiento del código (Domínguez, 2021). La programación orientada a objetos en C++ se basa en conceptos como clases, objetos, herencia y polimorfismo, lo que brinda a los programadores una mayor flexibilidad y reutilización de código.



Además de su capacidad para la programación orientada a objetos, C++ también ofrece características que lo hacen ideal para el desarrollo de sistemas complejos (Vélez, 2023). Una de estas características es la plantilla (template), que permite la creación de código genérico que puede adaptarse a diferentes tipos de datos; esto facilita la escritura de algoritmos y estructuras de datos que pueden ser utilizados con diversos tipos de variables.

Otra característica importante de C++, según García (2021) es su capacidad de manipulación de la memoria. A través de características como los punteros y la gestión manual de la memoria, los programadores tienen un control preciso sobre cómo se asignan y liberan los recursos en el sistema. Sin embargo, esta característica también requiere una mayor responsabilidad por parte del programador, ya que los errores en la gestión de la memoria pueden llevar a problemas como fugas de memoria o accesos no autorizados.

A medida que la tecnología continúa evolucionando, C++ ha seguido adaptándose para satisfacer las necesidades de los desarrolladores. La última versión del estándar de C++, conocida como C++20, según García et al. (2020) introdujo nuevas características y mejoras en el lenguaje, como el soporte para paralelismo y programación concurrente, así como la simplificación de ciertas operaciones comunes.

1.2.4. Evolución de C++ en el tiempo

A lo largo del tiempo, C++ ha experimentado varias mejoras y actualizaciones que han contribuido a su evolución y popularidad en el mundo de la programación. Desde su creación en 1985, como una extensión del lenguaje C, C++ se ha expandido (De Parga, 2021). Una de las primeras mejoras significativas en C++ fue la introducción de características orientadas a objetos, como clases, objetos, herencia y polimorfismo. Estas características permitieron a los programadores desarrollar programas más organizados y modulares.

A lo largo de los años, se han establecido estándares para C++, como el estándar ISO/IEC, que ha introducido nuevas funciones y mejoras en el lenguaje (Papy, 2020). Por ejemplo, el estándar C++11, lanzado en 2011, introdujo características como el soporte para programación. Además, C++ ha sido utilizado en una amplia gama de aplicaciones, desde el desarrollo de videojuegos hasta sistemas operativos. Su capacidad para la programación de bajo nivel y el control preciso



de los recursos del sistema lo han convertido en una opción popular para proyectos que requieren un alto rendimiento y una gestión eficiente de los recursos.

1.2.5. Dimensiones de la Herramienta “CODING C++”

La Herramienta “CODING C++” se fundamenta en cuatro dimensiones; a saber, Características y Funcionalidades; aplicaciones y casos de uso; herramientas y entornos de desarrollo; y, comunidad y recursos de aprendizaje

Dimensión características y funcionalidades

Esta dimensión se centra en las características y funcionalidades específicas de C++. Los indicadores de esta dimensión se vinculan a los conceptos clave del lenguaje, como la programación orientada a objetos, el manejo de memoria, las plantillas y las bibliotecas estándar. Estas características hacen que C++ sea una herramienta poderosa y versátil para el desarrollo de software, que permite crear programas eficientes y robustos (Román, 2022). En el entorno de programación "CODING C++", se encuentran varios conceptos clave que son fundamentales para comprender y utilizar este lenguaje de manera efectiva; algunos son la programación orientada a objetos, el manejo de memoria, las plantillas y las bibliotecas estándar.

La programación orientada a objetos (POO) es un paradigma de programación que se basa en la idea de organizar el código en objetos que representan entidades del mundo real; los objetos encapsulan datos y funciones relacionadas, lo que permite una mayor modularidad y reutilización del código. En C++, se pueden crear clases y objetos para implementar la POO; esto proporciona una estructura más clara y permite una mejor organización del código.

El manejo de memoria es otro concepto clave en C++. A diferencia de otros lenguajes de programación de más alto nivel, en C++ según refieren Martín et al. (2021) los programadores tienen un mayor control sobre la gestión de la memoria; esto significa que el programador es responsable de asignar y liberar la memoria utilizada por las variables y objetos; es importante considerar un buen manejo de memoria para evitar fugas y garantizar un rendimiento óptimo del programa.

Las plantillas son una característica poderosa de C++ que permiten la creación de código genérico; además, de escribir algoritmos y estructuras de datos que son independientes del tipo de datos con los que se trabajarán; esto proporciona una mayor flexibilidad y reutilización del



código, ya que se pueden utilizar las mismas plantillas para diferentes tipos de datos. Las plantillas son especialmente útiles para crear contenedores y algoritmos genéricos, como listas, vectores o algoritmos de ordenamiento.

Por último, las bibliotecas estándar son un conjunto de funciones y clases predefinidas que vienen con el lenguaje C++; estas proporcionan una amplia gama de funcionalidades, como manipulación de cadenas, entrada/salida, manejo de archivos, algoritmos de ordenamiento, entre otros. Utilizar las bibliotecas estándar de C++ ahorra tiempo y esfuerzo, ya que se pueden aprovechar las funciones y clases ya implementadas en lugar de tener que escribir el código desde cero.

Dimensión aplicaciones y casos de uso

Esta dimensión está relacionada con las aplicaciones y casos de uso de C++. Se utiliza con mayor frecuencia, como el desarrollo de videojuegos, sistemas embebidos, software de alto rendimiento y aplicaciones de tiempo real. C++ es un lenguaje de programación versátil y potente que ha sido ampliamente utilizado en una variedad de aplicaciones y casos de uso. Su combinación de características de alto nivel y bajo nivel lo convierte en una opción popular para desarrolladores que buscan un lenguaje eficiente y flexible (Papy, 2020). Algunas de las aplicaciones y casos de uso más comunes de C++ son las siguientes:

-Una de las áreas en las que C++ se destaca es el desarrollo de sistemas operativos y sistemas embebidos; debido a su capacidad para acceder directamente a la memoria y su eficiencia en el uso de recursos, C++ ha sido utilizado para desarrollar sistemas operativos como Windows y Linux, así como sistemas embebidos en dispositivos como teléfonos móviles, automóviles y electrodomésticos (Vélez, 2023).

-Otro caso de uso importante de C++ es el desarrollo de juegos y gráficos; ofrece un rendimiento excepcional y un control preciso sobre los recursos del sistema, lo que lo convierte en una opción ideal para el desarrollo de juegos y aplicaciones gráficas intensivas; muchos motores de juegos populares, como Unreal Engine y Unity, utilizan C++ como lenguaje base para su desarrollo.

-Es ampliamente utilizado en el campo de la inteligencia artificial y el aprendizaje automático (De Parga, 2021); la capacidad de C++ para manejar grandes cantidades de datos de manera eficiente y su capacidad para acceder directamente a la memoria hacen que sea una elección



popular para implementar algoritmos y modelos complejos en áreas como el procesamiento de imágenes, reconocimiento de voz y análisis de datos.

-El desarrollo de aplicaciones de alto rendimiento, como sistemas de tiempo real, servidores y controladores de dispositivos, también es un caso de uso común para C++. La capacidad de C++ para controlar directamente el hardware y su bajo consumo de recursos lo hacen adecuado para aplicaciones que requieren una respuesta rápida y un uso eficiente de los recursos del sistema (Román, 2022).

Dimensión herramientas y entornos de desarrollo

Esta dimensión se enfoca en las herramientas y entornos de desarrollo disponibles para programar en C++. Entre sus indicadores se tienen los IDEs (Entornos de Desarrollo Integrados) más populares para C++, las características que ofrecen, las ventajas y desventajas de cada uno, y cómo facilitan el proceso de desarrollo en C++.

Los Entornos de Desarrollo Integrados (IDE, por sus siglas en inglés) son herramientas esenciales para los programadores de C++ que buscan agilizar y facilitar el proceso de desarrollo de software. Hay varios IDE populares disponibles para C++, cada uno con sus propias características, ventajas y desventajas (Quevedo, 2022). Algunos de los IDE más populares y cómo pueden facilitar el proceso de desarrollo en C++ son los siguientes:

1. *Visual Studio*: Desarrollado por Microsoft, Visual Studio es uno de los IDE más ampliamente utilizados para C++. Según Martín et al. (2021) ofrece una amplia gama de características, como resaltado de sintaxis, depuración integrada, finalización de código, refactorización y soporte para proyectos grandes. Una de las ventajas clave de Visual Studio es su integración con otras herramientas y lenguajes de Microsoft, lo que facilita el desarrollo de aplicaciones multiplataforma utilizando C++.

2. *Eclipse*: es otro IDE popular que ofrece soporte para C++. Además de las características básicas de un IDE, Eclipse proporciona un entorno de desarrollo modular y personalizable a través de la instalación de complementos; esto permite a los desarrolladores agregar funcionalidades adicionales según sus necesidades específicas. Eclipse también es conocido por su capacidad de depuración y perfilado avanzado, lo que facilita la identificación y solución de problemas en el código C++ (De Parga, 2021).



3. *Code::Blocks*: es un IDE de código abierto que se destaca por su simplicidad y facilidad de uso. Ofrece una interfaz intuitiva y una configuración sencilla, lo que lo hace ideal para principiantes en C++. Aunque puede carecer de algunas características avanzadas, Code:Blocks proporciona las funciones básicas necesarias para el desarrollo en C++, como resaltado de sintaxis, depuración y finalización de código.

4. *CLion*: es un IDE desarrollado por JetBrains y está diseñado específicamente para el desarrollo de software en C++ y otros lenguajes; ofrece una amplia gama de características, como refactorización inteligente, navegación rápida por el código, depuración avanzada y análisis de código estático. CLion también se destaca por su integración con CMake, un sistema de construcción popular en el desarrollo de C++. Esto facilita la gestión de proyectos complejos y la configuración del entorno de desarrollo.

Cada uno de estos IDE tiene sus propias ventajas y desventajas. Algunas ventajas comunes son la facilidad de uso, la depuración eficiente, el resaltado de sintaxis y la finalización de código. Sin embargo, los IDE más completos, como Visual Studio y CLion, pueden ser más pesados y requerir más recursos de hardware en comparación con IDE más ligeros como Code:Blocks. En general, los IDE facilitan el proceso de desarrollo en C++ al proporcionar herramientas y características que ahorran tiempo y mejoran la productividad del programador (García B. L., 2021).

Estos IDE ofrecen resaltado de sintaxis para una mejor legibilidad del código, depuración integrada para identificar y resolver errores, finalización de código para una escritura más rápida y precisa, y muchas otras características útiles para el desarrollo en C++; además, los IDE ayudan a administrar proyectos grandes y complejos, proporcionando herramientas para la organización del código, la navegación rápida y la gestión de dependencias.

Dimensión comunidad y recursos de aprendizaje

Esta dimensión se refiere a la comunidad de programadores y los recursos de aprendizaje disponibles para C++. Esta comunidad de programadores de C++ es amplia y activa, lo que proporciona un valioso recurso para aquellos que desean aprender y mejorar en el lenguaje. Algunos de los recursos de aprendizaje disponibles para C++ incluyen (De Parga, 2021):



1. *Stack Overflow*: Es una plataforma donde los programadores pueden hacer consultas a sus pares en cualquier parte del mundo. Además de servir como un foro de preguntas y respuestas, Stack Overflow ofrece una vasta cantidad de recursos y materiales educativos e informativos para los programadores, incluyendo artículos, tutoriales y herramientas de aprendizaje.
2. *Reddit*: Hay subreddits específicos para diferentes lenguajes de programación, proyectos de código abierto y temas relacionados con la informática. En estos subreddits, los usuarios pueden hacer preguntas, compartir recursos y discutir sobre diferentes temas. Reddit es un excelente sitio para conectarse con otros programadores y aprender de la comunidad.
3. *GitHub*: Es una plataforma de desarrollo colaborativo que permite a los programadores compartir y colaborar en proyectos de código abierto. Además de encontrar código fuente abierto aplicable o útil para proyectos, GitHub también ofrece una sección de consejos y recursos para resolver problemas y aprender cosas nuevas. La comunidad de GitHub es útil y productiva, y se puede aprender mucho allí.
4. *Cursos y tutoriales en línea*: Existen numerosos recursos en línea, como cursos, tutoriales y blogs, que ofrecen material educativo y guías de aprendizaje para C++. Estos recursos pueden ser útiles para principiantes y programadores más experimentados que deseen mejorar sus habilidades en el lenguaje.
5. *Comunidades y grupos de desarrollo*: Hay comunidades y grupos en línea, como CodersLink, donde los programadores pueden conectarse, compartir conocimientos y colaborar en proyectos. Estas comunidades ofrecen un espacio para la interacción y el intercambio de ideas entre programadores.

1.3. Fundamentación tecnológica

Como fundamentación tecnológica, este estudio se instituye en el modelo conectivista respaldado por Van Dijk (2019), quien ofrece una definición de conectivismo como una manera de educación que se adhiere a los frutos de las tecnologías e internet para favorecer los cambios que a nivel cognitivo están experimentando los individuos producto del contacto y manipulación de estas. Es decir, el concepto de conectividad se considera una forma de expresar los fundamentos de la naturaleza humana y de la educación moderna, que se caracteriza por un alto nivel de interacción.



El conectivismo es una teoría del aprendizaje que surge en el contexto de la era digital y la sociedad de la información, fue desarrollada por George Siemens y Stephen Downes en el año 2004, el conectivismo se basa en la idea de que el conocimiento se encuentra distribuido en redes y sistemas, y que el aprendizaje se produce a través de la conexión y la interacción con estos entornos (Prado, 2021).

Una de las principales características del conectivismo es el enfoque en las conexiones y las relaciones entre personas, ideas y recursos; en este sentido, el aprendizaje se da a través de la intervención en comunidades de práctica, la colaboración con otros individuos y la interacción con diversas fuentes de información. Enfatizando la importancia de construir una red personal de aprendizaje, donde se establecen conexiones significativas con expertos, compañeros y recursos relevantes.

Además, este estudio se puede fundamentar en la necesidad de adaptar los métodos de enseñanza a las características y preferencias de los estudiantes de la generación actual, quienes están inmersos en un entorno digital y tienen una gran afinidad con la tecnología, así el uso de la herramienta "CODING C++" permitiría aprovechar su familiaridad con la tecnología y su interés por la programación para motivar y comprometer a los contenidos de asignatura.

1.4. Fundamentación psicológica

La fundamentación psicológica de esta investigación se basa en varios aspectos. En primer lugar, se cimenta en la teoría del aprendizaje constructivista, la cual sostiene que el conocimiento se construye mediante la interacción del aprendiz con el contexto y los significados tal como refieren Moreno et al. (2024). De esta forma, el uso de la herramienta "CODING C++" proporcionaría a los estudiantes la oportunidad de participar en la creación y resolución de problemas de programación, lo que fomentaría su intervención y compromiso con el proceso de aprendizaje.

Además, el uso de la herramienta "CODING C++" promovería el desarrollo de habilidades cognitivas y metacognitivas en los estudiantes, debido a que la programación requiere destrezas de pensamiento lógico, resolución de problemas y planificación, lo que contribuiría al desarrollo de habilidades de pensamiento crítico y habilidades para el aprendizaje autónomo. La programación en C++ requiere que los estudiantes analicen y descompongan problemas



complejos en pasos más pequeños y manejables, lo cual fomenta el pensamiento lógico y la capacidad de resolver problemas de manera estructurada.

Involucra la creación de algoritmos para resolver problemas; los estudiantes deben aprender a diseñar algoritmos eficientes y optimizados, lo que desarrolla su capacidad de pensamiento algorítmico y su habilidad para planificar y organizar soluciones. Requiere atención al detalle y precisión en la escritura del código; en este sentido, los estudiantes deben aprender a identificar y corregir errores, lo que mejora su capacidad de atención y su habilidad para detectar problemas en el código.

Con respecto a la metacognición; la programación en C++ conlleva a reflexionar sobre el propio proceso de programación y evaluar la efectividad de las soluciones; así los estudiantes deben ser conscientes de sus propias habilidades y limitaciones, lo que promueve la metacognición y la capacidad de autorregulación en el aprendizaje. Permite a los estudiantes encontrar soluciones creativas a problemas y desafíos; también fomenta el pensamiento crítico al requerir que los estudiantes evalúen diferentes enfoques y tomen decisiones informadas sobre la mejor manera de abordar un problema.

Como segundo aspecto de la fundamentación psicológica sería el impacto positivo en la motivación de los estudiantes, esto debido a que la programación puede ser un tema desafiante y complejo (Preciado, 2024); no obstante, el enfoque interactivo y visualmente atractivo de la herramienta puede aumentar el interés de los estudiantes en la programación, lo que a su vez puede impactar positivamente en su motivación intrínseca hacia el aprendizaje de la asignatura. Además, el uso de la herramienta puede proporcionar a los estudiantes una sensación de logro y autorrealización al enfrentarse a los desafíos propios de la programación.

Además, se puede argumentar que el uso de la herramienta "CODING C++" fomentaría el desarrollo de habilidades socioemocionales en los estudiantes, debido a que la programación requiere habilidades de trabajo en equipo, comunicación y perseverancia, lo que contribuiría al desarrollo de habilidades sociales y emocionales que son importantes para el éxito académico y personal.



1.5. Fundamentación pedagógica

En términos pedagógicos, se puede fundamentar en la necesidad de utilizar enfoques innovadores y adaptados a las características y necesidades de los estudiantes tal como declaran Ontoria et al. (2023). Considerando que la programación estructurada es compleja por lo cual requiere un enfoque didáctico adecuado para facilitar la comprensión y el aprendizaje de los conceptos. El uso de la herramienta "CODING C++" puede proporcionar un entorno interactivo y visualmente atractivo que motive a los estudiantes y facilite su participación en el proceso de enseñanza-aprendizaje.

Las teorías pedagógicas del aprendizaje que respaldan el uso de la tecnología en la educación, como el constructivismo social, plantean la necesidad de facilitar la comunicación y el diálogo para compartir, intercambiar y discutir aprendizajes con los demás agentes educativos (Quispe P. , 2024); de esta forma, se puede fundamentar en la teoría del aprendizaje constructivista, que sostiene que el conocimiento se construye activamente a través de la interacción del estudiante con su entorno y la construcción de significados.

Otro aspecto importante de la fundamentación pedagógica es la teoría del aprendizaje basada en problemas, que enfatiza el aprendizaje a través de la resolución de problemas auténticos; el uso de la herramienta "CODING C++" proporciona a los estudiantes la oportunidad de enfrentarse a desafíos de programación y resolver problemas mediante un entorno de desarrollo integrado con características como un editor de código, capacidades de depuración, acceso a bibliotecas y recursos, y una comunidad en línea para obtener soporte adicional; estas herramientas y recursos ayudan a los estudiantes a mejorar sus habilidades de programación, abordar desafíos y resolver problemas de manera efectiva.

1.6. Fundamentación legal

En Ecuador, la promulgación de la Constitución de la República de Ecuador en (2008) ha permitido emprender un camino de grandes cambios para transformar radicalmente la Educación, respaldando la importancia de la educación, la ciencia, la tecnología y la innovación en el contexto educacional. De esta forma, el Artículo 26, establece el derecho a la educación, que garantiza el acceso, permanencia, calidad y equidad en el sistema educativo. El artículo 27, reconoce el derecho a la ciencia, la tecnología y la innovación. Y el artículo 28, establece el



derecho a la cultura y al conocimiento. Con este fundamento, la propuesta de utilizar la herramienta "CODING C++" puede contribuir al mejoramiento de la calidad educativa y al acceso a una educación de calidad en el campo de la programación estructurada, mediante el uso de tecnología y la promoción de la innovación, contribuyendo al desarrollo de habilidades y conocimientos.

También establece su fundamento legal en la Ley Orgánica de Educación Intercultural (LOEI), Esta ley establece los principios y fines de la educación en Ecuador, promoviendo la calidad, equidad, inclusión y pertinencia educativa. Además, fomenta la innovación y el uso de tecnologías en el proceso educativo (República del Ecuador, 2011, 31 de marzo). En específico, el Artículo 36, refiere a la evaluación para mejorar una calificación en educación media y bachillerato, lo que implica la posibilidad de rendir nuevas evaluaciones para mejorar las calificaciones en asignaturas específicas; y, el artículo 46, aborda el material curricular didáctico, lo que implica la posibilidad de utilizar la herramienta "CODING C++" como parte del material curricular en la asignatura de programación estructurada.



CAPÍTULO 2: METODOLOGÍA PARA EL DESARROLLO DE LA INVESTIGACIÓN Y ESTUDIO DIAGNÓSTICO

2.1. Conceptualización y operacionalización de las variables

2.1.1. *Variable Dependiente: Proceso de enseñanza – aprendizaje de programación estructurada.*

Definición conceptual

Es el que proporciona los fundamentos necesarios para comprender y aplicar los conceptos esenciales de la programación, sentando las bases para el desarrollo de habilidades técnicas y lógicas tal como refieren Mayeta et al. (2020).

Definición operacional

Operacionalmente se mide a través de cinco dimensiones tales como: la educativa, la metodológica, la interactiva, la motivacional y la evaluativa; cada una con sus respectivos indicadores como se muestran en la tabla 1.

2.1.2. *Variable Independiente: Herramienta “CODING C++”.*

Definición conceptual

Según refieren Martín et al. (2021), la herramienta CODING C++ es definida como aquella que permite el control preciso sobre los recursos del sistema, como la memoria y los registros de la CPU. Esto ha hecho que C++ sea la opción preferida para el desarrollo de aplicaciones que requieren un alto rendimiento y un uso eficiente de los recursos, como los videojuegos y los sistemas embebidos.

Definición operacional

Operacionalmente se mide a través de cuatro dimensiones tales como: características y funcionalidades, aplicaciones y casos de uso, la interactiva, herramientas y entornos de desarrollo y comunidad y recursos de aprendizaje; cada una con sus respectivos indicadores como se presenta en la tabla 1.



Tabla 1

Operacionalización de variables

Variables	Definición	Dimensión	Indicadores	Instrumentos de recolección de la información	Escalas de valoración
Variable Dependiente: Proceso de enseñanza – aprendizaje de programación estructurada	Es el que proporciona los fundamentos necesarios para comprender y aplicar los conceptos esenciales de la programación, sentando las bases para el desarrollo de habilidades técnicas y lógicas tal como refieren Mayeta et al. (2020).	Educativa	-Adquiere conceptos de programación estructurada -Desarrolla habilidades técnicas -Comprende principios de la programación estructurada	Cuestionario a estudiantes	Escala de medición: Tipo Likert
		Metodológica	-Estrategias de enseñanza -Métodos utilizados para enseñar		
		Interactiva	-Se comunica efectivamente -Ofrece apoyo individualizado -Brinda retroalimentación constructiva		
		Motivacional	-Crea ambientes de aprendizaje estimulante -Conecta los contenidos con situaciones de la vida real -Promueve la autonomía estudiantil		
		Evaluativa	-Utiliza diferentes tipos de evaluación -Mide el nivel de comprensión de los conocimientos		



Variables	Definición	Dimensión	Indicadores	Instrumentos de recolección de la información	Escalas de valoración
Variable Independiente: Herramienta “CODING C++”	Es aquella que permite el control preciso sobre los recursos del sistema, como la memoria y los registros de la CPU. Esto ha hecho que C++ sea la opción preferida para el desarrollo de aplicaciones que requieren un alto rendimiento y un uso eficiente de los recursos, como los videojuegos y los sistemas embebidos (Martín et al., 2021)	Funcionalidades	-Utiliza programación orientada a objetos -Hace un buen manejo de memoria -Utiliza plantillas y bibliotecas estándar	Entrevista semiestructurada a docentes	No hay una escala numérica, se establecen criterios de claridad, profundidad, relevancia y originalidad de las respuestas. Escala cualitativa
		Aplicaciones	-Ofrece desarrollo de juegos y aplicaciones gráficas -Maneja grandes cantidades de datos de manera eficiente -Muestra una respuesta rápida y uso eficiente de los recursos del sistema		
		Entornos de desarrollo	-Utiliza los IDEs (Entornos de Desarrollo Integrados) más populares -Los IDEs utilizados facilitan el proceso de desarrollo		
		Recursos de aprendizaje	-Utiliza los recursos de aprendizaje disponibles para C++ -Ofrece comunidades y grupos en línea		

Fuente: propia



2.2. Enfoque de la Investigación

Se asume la ruta de investigación con enfoque mixto. Se usa para una comprensión profunda y detallada de las experiencias, percepciones y contextos sociales relacionados con el uso de la herramienta "CODING C++" en el proceso de enseñanza-aprendizaje de la programación estructurada; gracias al enfoque mixto es posible combinar las fortalezas de los métodos cualitativos y cuantitativos para validar y complementar mutuamente los hallazgos (Arenas, 2021).

2.3. Alcance de la investigación

Este estudio tiene un alcance propio de la investigación descriptiva y propositiva; es descriptiva ya que se proporciona una descripción detallada y precisa del contexto y las acciones necesarias para la implementación exitosa de la herramienta digital. Específicamente porque se caracteriza el proceso de enseñanza-aprendizaje de la asignatura de programación estructurada en los estudiantes de segundo de bachillerato paralelo "A" de la Unidad Educativa Amazonas, durante el periodo de febrero a octubre de 2024, antes de la implementación de la herramienta "CODING C++".

Es propositiva ya que busca aplicar el conocimiento existente para mejorar un proceso educativo práctico en colaboración con una institución educativa, teniendo impacto directo en la calidad de la educación en la U.E. Amazonas, además de poder servir como modelo para otras instituciones educativas que enfrenten desafíos similares en el proceso de enseñanza - aprendizaje de programación.

2.4. Declaración y justificación del tipo de investigación

Se aplican varios tipos de investigación, tales como Bibliográfica, y De campo; además, de acuerdo al tiempo, es de tipo transversal. Es Bibliográfica debido a que se realiza la fundamentación teórica, se revisan estudios previos, libros, artículos y otros documentos relevantes para recopilar información sobre el uso de herramientas digitales en la enseñanza de programación estructurada. Es de campo porque los datos se recogen directamente en la Unidad Educativa Amazonas.

Es de tipo transversal porque los datos se recopilan en un solo momento en el tiempo, lo que permite comparar diferentes grupos o poblaciones en un momento dado, en el caso de ésta



propuesta la investigación se centra durante el periodo de febrero a octubre de 2024, tiempo en el que se debe recolectar, tabular, y analizar los datos obtenidos de los estudiantes de segundo de bachillerato paralelo “A”, y validar la propuesta de la implementación de la herramienta “CODING C++” en el proceso de enseñanza-aprendizaje de programación durante un período académico específico.

2.5. Métodos empleados y sus propósitos en el contexto de investigación

Los métodos empleados son de tres tipos: teóricos, empíricos y matemáticos estadísticos. Entre los Métodos Teóricos se consideraron el de Análisis y síntesis, el enfoque de Sistemas; y, la revisión bibliográfica y análisis de documentos.

Análisis y síntesis: Este método se utilizó en la FASE 1: Fundamentación, específicamente al determinar los fundamentos teóricos que sustentan el uso de la herramienta “CODING C++” en el proceso de enseñanza-aprendizaje de programación estructurada. Se analizaron diferentes teorías y se sintetizaron los conceptos relevantes para la investigación.

Enfoque de Sistema: Se aplicó principalmente en la FASE 1: Fundamentación, durante la determinación de los fundamentos teóricos. Ayudó a entender cómo las diferentes partes de un sistema interactúan entre sí en el proceso de enseñanza-aprendizaje de la programación estructurada.

Revisión bibliográfica y análisis de documentos: Estos métodos se utilizaron también en la FASE 1: Fundamentación, como parte de la investigación teórica. Se revisaron estudios previos, libros, artículos y otros documentos relevantes para recopilar información sobre el uso de herramientas digitales en la enseñanza de programación estructurada.

El Método Empírico usado fue la encuesta. Se aplicó tanto en la FASE 2: Exploración como en la FASE 4: Validación. En la fase 2, se utilizó para recopilar información sobre las características del proceso de enseñanza-aprendizaje de la asignatura de programación estructurada en los estudiantes de segundo de bachillerato paralelo “A” de la Unidad Educativa Amazonas, durante el periodo de febrero a octubre de 2024, antes de la implementación de la herramienta “CODING C++”. En la fase 4, se utilizó para evaluar la percepción de los docentes sobre la incidencia de la herramienta "CODING C++" en el proceso de enseñanza – aprendizaje.



Como Métodos Matemáticos o Estadísticos se aplicó el Análisis Estadístico; este se utilizó en las FASES 2 y 4: para obtener información detallada sobre la experiencia de los estudiantes y profesores con la asignatura de programación estructurada con métodos tradicionales; y para la validación de la propuesta. Los análisis estadísticos proporcionarán evidencia objetiva de la incidencia de la herramienta; a su vez, la tabulación ayudó a presentar los resultados de manera clara y sistemática.

2.6. Instrumentos derivados de la metodología seleccionada

Se diseñaron dos instrumentos: un cuestionario y una entrevista semiestructurada. El cuestionario (Anexo 1), compuesto por trece preguntas distribuidas entre las cinco dimensiones, se aplicó a estudiantes con el propósito de recolectar datos que permitan caracterizar el proceso de enseñanza-aprendizaje de la asignatura de programación estructurada de la Unidad Educativa Amazonas, durante el periodo de febrero a octubre de 2024, antes de la implementación de la herramienta “CODING C++”.

La entrevista semiestructurada (Anexo 2) compuesta por cinco preguntas, se aplicó a los docentes con el propósito de validar la propuesta diseñada para el uso de la herramienta “CODING C++” en el mejoramiento del proceso de enseñanza-aprendizaje de la asignatura de programación estructurada en los estudiantes de segundo de bachillerato, de la Unidad Educativa Amazonas, durante el periodo febrero a octubre de 2024. Ambos instrumentos fueron sometidos a revisión de tres expertos, con la finalidad de validar el contenido y verificar que cumplieran con el propósito definido en este estudio (Anexo 3).

2.7. Delimitación de la población y la muestra

La población de estudio son los estudiantes de segundo año de bachillerato técnico, siendo un total de 99, divididos en 3 paralelos. Además de tres (03) docentes que administran esta asignatura. La muestra fue seleccionada de manera intencional quedando conformada por 32 estudiantes del segundo año de bachillerato paralelo “A”. El muestreo fue no probabilístico, por conveniencia, ya que los participantes fueron seleccionados de manera conveniente y accesible para los investigadores, además de que los recursos como el tiempo y el dinero son limitados. Al enfocarse en un solo paralelo, se simplifica la logística de la investigación para la intervención



educativa, lo cual incluye la coordinación con los docentes y la programación de actividades, lo facilitando la recolección de datos y la evaluación de la propuesta.

2.8. Estrategia metodológica investigativa o proceder metodológico general seguido en el proceso de investigación de acuerdo con el alcance e intereses de la investigación

Para desarrollar esta investigación de enfoque mixto, se siguió la siguiente estrategia metodológica:

1. Se realizó la definición del problema de investigación, luego de identificar el problema específico relacionado con el perfeccionamiento del proceso de enseñanza-aprendizaje de programación estructurada en el segundo año de bachillerato de la Unidad Educativa Amazonas. El cual es problema complejo y multifacético, lo que justifica el uso de un enfoque mixto para obtener una comprensión más completa.
2. Se formularon los objetivos generales y específicos que guiaron la investigación, se consideraron objetivos cuantitativos; por ejemplo, caracterizar el proceso de enseñanza-aprendizaje de la asignatura de programación estructurada en los estudiantes de segundo de bachillerato paralelo “A” de la Unidad Educativa Amazonas, durante el periodo de febrero a octubre de 2024, antes de la implementación de la herramienta “CODING C++”; como cualitativos; por ejemplo, validar la propuesta diseñada para el uso de la herramienta “CODING C++” en el mejoramiento del proceso de enseñanza-aprendizaje de la asignatura de programación estructurada en los estudiantes de segundo de bachillerato paralelo “A”, de la Unidad Educativa Amazonas, durante el periodo febrero a octubre de 2024.
3. Se seleccionaron los métodos; por ejemplo, se aplicaron, el cuestionario para recopilar datos cuantitativos y la entrevista semiestructurada para obtener información cualitativa. La combinación de ambos te permitió enriquecer los hallazgos.
4. Se diseñó la investigación que integró ambos enfoques, adoptando el diseño secuencial, donde el enfoque cuantitativo se utilizó primero y el cualitativo después.
5. Recolección de Datos, aplicando los instrumentos luego de ser validados por expertos.
6. Se analizaron los datos, iniciando con los cuantitativos, mediante técnicas estadísticas; y los datos cualitativos mediante análisis de contenido.



7. Interpretación de resultados mediante la integración de los hallazgos de ambos enfoques para ofrecer una interpretación más rica y completa del problema de investigación.

8. Se redactaron las conclusiones basadas en la integración de los datos y se ofrecieron recomendaciones prácticas que surgieron de los resultados de la investigación.

2.9. La descripción de la metodología de acuerdo con las tareas de investigación se constituye en un apoyo para este apartado del trabajo de titulación. Implica la descripción de las etapas seguidas en el proceso investigativo y su propósito

Etapas del estudio teórico

Esta etapa corresponde a la Fundamentación teórica del estudio, cuyo objetivo consistió en determinar los fundamentos teóricos que sustentan el uso de la herramienta “CODING C++” en el proceso de enseñanza-aprendizaje de programación estructurada. (Marco teórico)

Etapas del diagnóstico inicial

Esta etapa corresponde a la Exploración, su objetivo fue caracterizar el proceso de enseñanza-aprendizaje de la asignatura de programación estructurada en los estudiantes de segundo de bachillerato paralelo “A” de la Unidad Educativa Amazonas, durante el periodo de febrero a octubre de 2024, antes de la implementación de la herramienta “CODING C++”. (Diagnóstico)

Etapas de la modelación de la propuesta

Esta etapa corresponde a la Definición de la propuesta, su objetivo fue diseñar la propuesta de una guía que permita el uso de la herramienta “CODING C++” en el mejoramiento del proceso de enseñanza-aprendizaje de la asignatura de programación estructurada en los estudiantes de segundo de bachillerato paralelo “A”, de la Unidad Educativa Amazonas, durante el periodo febrero a octubre de 2024. **(Propuesta)**

Etapas del diagnóstico final o validación de la propuesta (teórica o empírica)

Esta etapa corresponde a la Validación, su objetivo fue validar la propuesta diseñada para el uso de la herramienta “CODING C++” en el mejoramiento del proceso de enseñanza-aprendizaje de la asignatura de programación estructurada en los estudiantes de segundo de bachillerato paralelo “A”, de la Unidad Educativa Amazonas, durante el periodo febrero a octubre de 2024. **(Validación)**



2.10. Presentación de los resultados del estudio diagnóstico

Se presenta los resultados del diagnóstico inicial que consistió en caracterizar el proceso de enseñanza-aprendizaje de la asignatura de programación estructurada de la Unidad Educativa Amazonas, durante el periodo de febrero a octubre de 2024, antes de la implementación de la herramienta “CODING C++”. Para la variable dependiente: Proceso de enseñanza – aprendizaje de programación estructurada, se presentan las tablas y figuras a continuación:

Tabla 2.

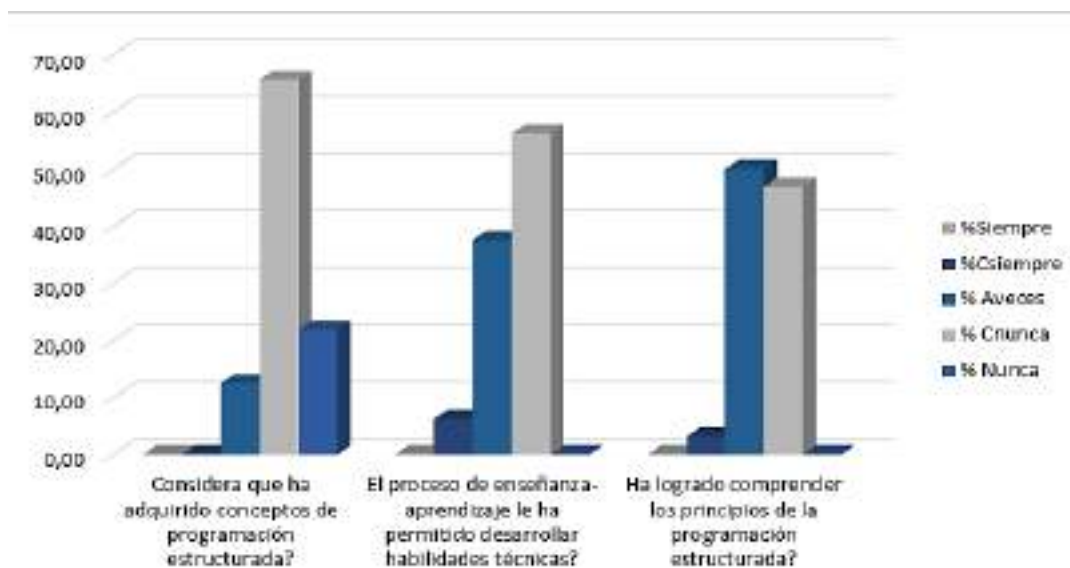
Resultados de la dimensión educativa de la Variable dependiente

Ítem	VID1. Educativa	Alternativas				
		% S	%CS	%AV	%CN	%N
1	¿Considera que ha adquirido conceptos de programación estructurada?	0,00	0,00	12,50	65,63	21,88
2	¿El proceso de enseñanza-aprendizaje le ha permitido desarrollar habilidades técnicas?	0,00	6,25	37,50	56,25	0,00
3	¿Ha logrado comprender los principios de la programación estructurada?	0,00	3,13	50,00	46,88	0,00
Media		0,00	3,13	33,33	56,25	7,29

Nota. Elaboración propia a partir de los datos del cuestionario aplicado a estudiantes

Figura 1.

Resultados de la dimensión educativa de la Variable dependiente



Nota. Elaboración propia a partir de los datos del cuestionario aplicado a estudiantes



Como se evidencia en la tabla 2 y figura 1, la mayoría de los aprendices ubicaron su respuesta entre las alternativas casi nunca y algunas veces para los ítems de esta dimensión, destacando que el 65,63% de ellos considera que casi nunca ha adquirido conceptos de programación estructurada; de igual forma, el 56,25% expresó que el proceso de enseñanza-aprendizaje casi nunca le ha permitido desarrollar habilidades técnicas; y el 50% sólo algunas veces ha logrado comprender los principios de la programación estructurada. La media calculada para esta dimensión fue del 56,25% para la alternativa casi nunca, lo que implica que la mayoría de los estudiantes expresaron que sus docentes casi nunca consideran la dimensión educativa; esto es según Pérez (2024) casi nunca brindan una base sólida de conocimientos teóricos y la oportunidad de aplicarlos en situaciones prácticas, o les prepara para enfrentar desafíos de su cotidianidad y desarrollar habilidades relevantes para su futuro.

Tabla 3.

Resultados de la dimensión metodológica de la Variable dependiente

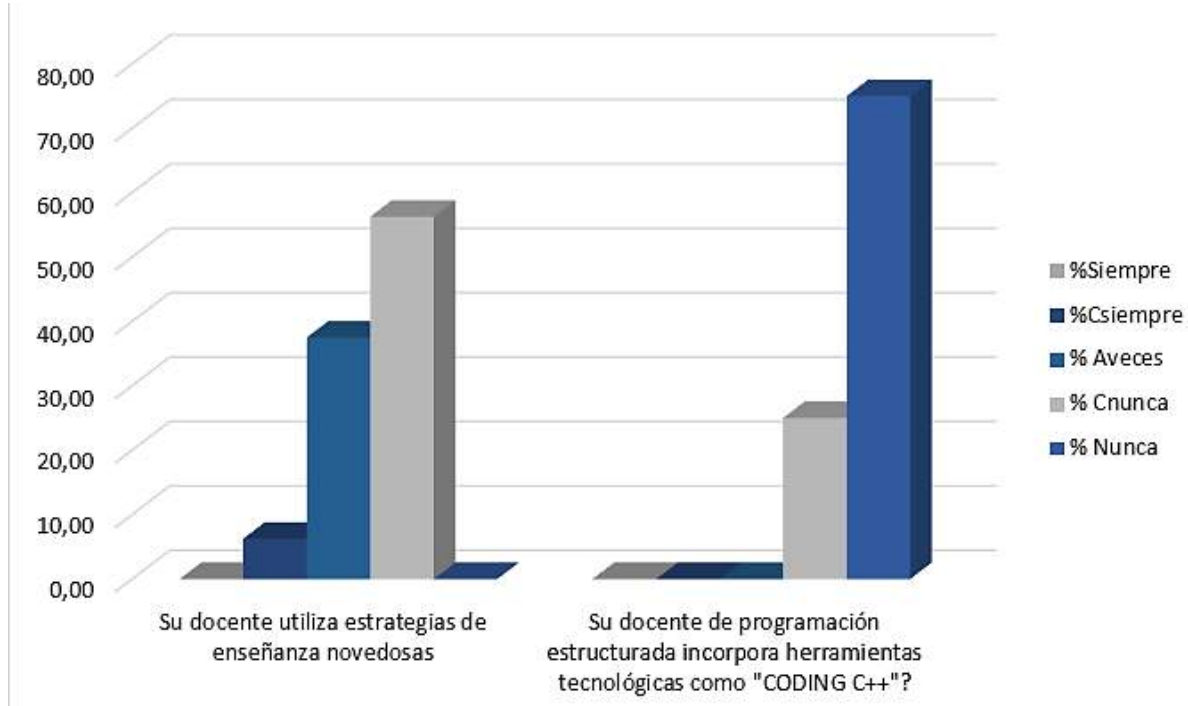
Ítem	VID2. Metodológica	Alternativas				
		% S	%CS	%AV	%CN	%N
4	¿Su docente utiliza estrategias de enseñanza novedosas?	0,00	6,25	37,50	56,25	0,00
5	¿Su docente de programación estructurada incorpora herramientas tecnológicas como "CODING C++"?	0,00	0,00	0,00	25,00	75,00
Media		0,00	3,13	18,75	40,63	37,50

Nota. Elaboración propia a partir de los datos del cuestionario aplicado a estudiantes



Figura 2.

Resultados de la dimensión metodológica de la Variable dependiente



Nota. Elaboración propia a partir de los datos del cuestionario aplicado a estudiantes

Según se evidencia en la tabla 3 y figura 2, la mayoría de los aprendices ubicaron su respuesta entre las alternativas casi nunca y nunca para los ítems de esta dimensión metodológica, destacando que el 56,25% de ellos considera que casi nunca su docente utiliza estrategias de enseñanza novedosa; y el 75% expresó que su docente de programación estructurada nunca incorpora herramientas tecnológicas como "CODING C++". La media calculada para esta dimensión fue del 40,63% para la alternativa casi nunca, lo que implica que la mayoría de los estudiantes expresaron que sus docentes casi nunca consideran la dimensión metodológica; esto es según Coapaza et al. (2024), estos docentes se apoyan en una de las estrategias más comunes, consideradas tradicionales, como es el enfoque expositivo, en el cual, el educador presenta información de manera estructurada; sin considerar aspectos relevantes como la introducción de recursos y estrategias novedosas que contemplen el uso de la tecnología.



Tabla 4.

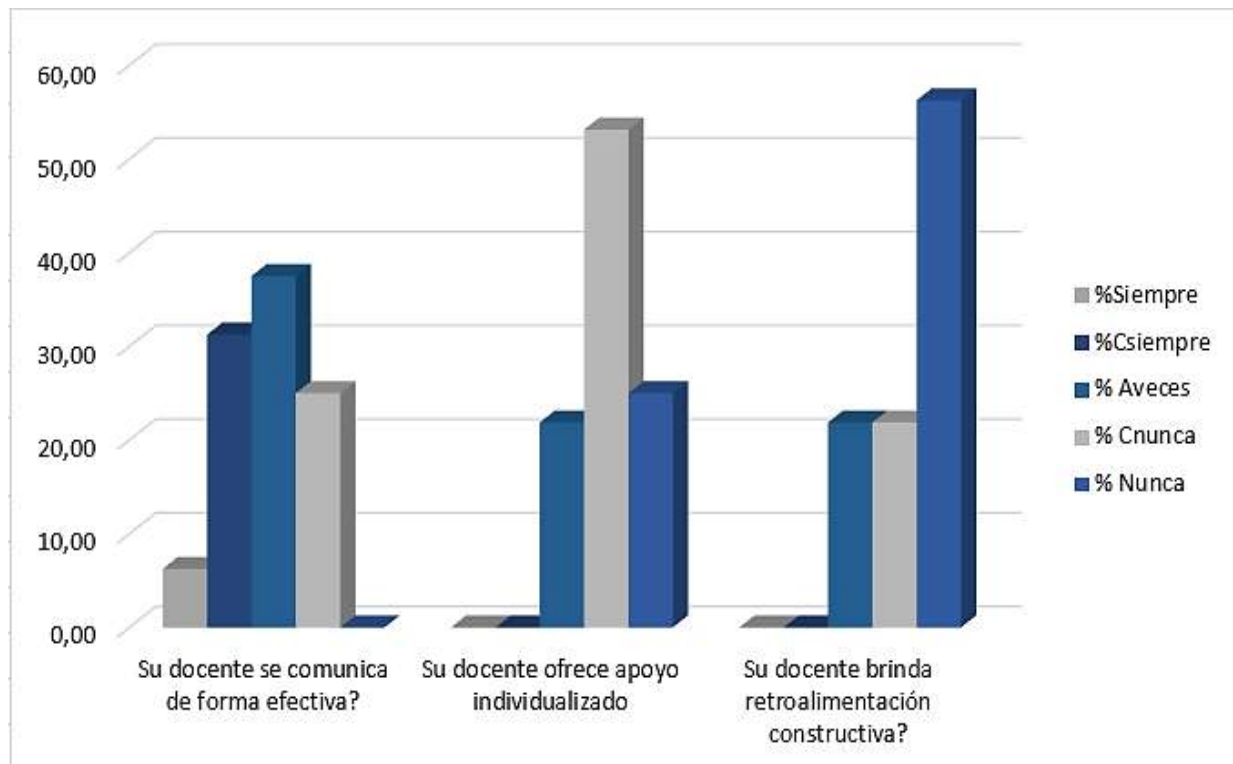
Resultados de la dimensión interactiva de la Variable dependiente

Ítem	V1D3. Interactiva	Alternativas				
		% S	%CS	%AV	%CN	%N
6	¿Su docente se comunica de forma efectiva?	6,25	31,25	37,50	25,00	0,00
7	¿Su docente ofrece apoyo individualizado?	0,00	0,00	21,88	53,13	25,00
8	¿Su docente brinda retroalimentación constructiva?	0,00	0,00	21,88	21,88	56,25
Media		1,56	7,81	20,31	25,00	20,31

Nota. Elaboración propia a partir de los datos del cuestionario aplicado a estudiantes

Figura 3.

Resultados de la dimensión interactiva de la Variable dependiente



Nota. Elaboración propia a partir de los datos del cuestionario aplicado a estudiantes

Tal como se muestra en la tabla 4 y figura 3, la mayoría de los aprendices ubicaron su respuesta entre las alternativas algunas veces, casi nunca y nunca para los ítems de esta dimensión interactiva, destacando que el 37,50% de ellos considera que algunas veces su docente se



comunica de forma efectiva; el 53,13% expresó que su docente de programación estructurada nunca ofrece apoyo individualizado; y, un 56,25% opinó que su docente nunca brinda retroalimentación constructiva. La media calculada para esta dimensión fue del 25,00% para la alternativa casi nunca, lo que implica que la mayoría de los estudiantes expresaron que sus docentes casi nunca consideran la dimensión interactiva; esto es, casi nunca ocurre una interacción adecuada entre el docente y los estudiantes durante el proceso de enseñanza-aprendizaje de programación estructurada.

Tabla 5.

Resultados de la dimensión motivacional de la Variable dependiente

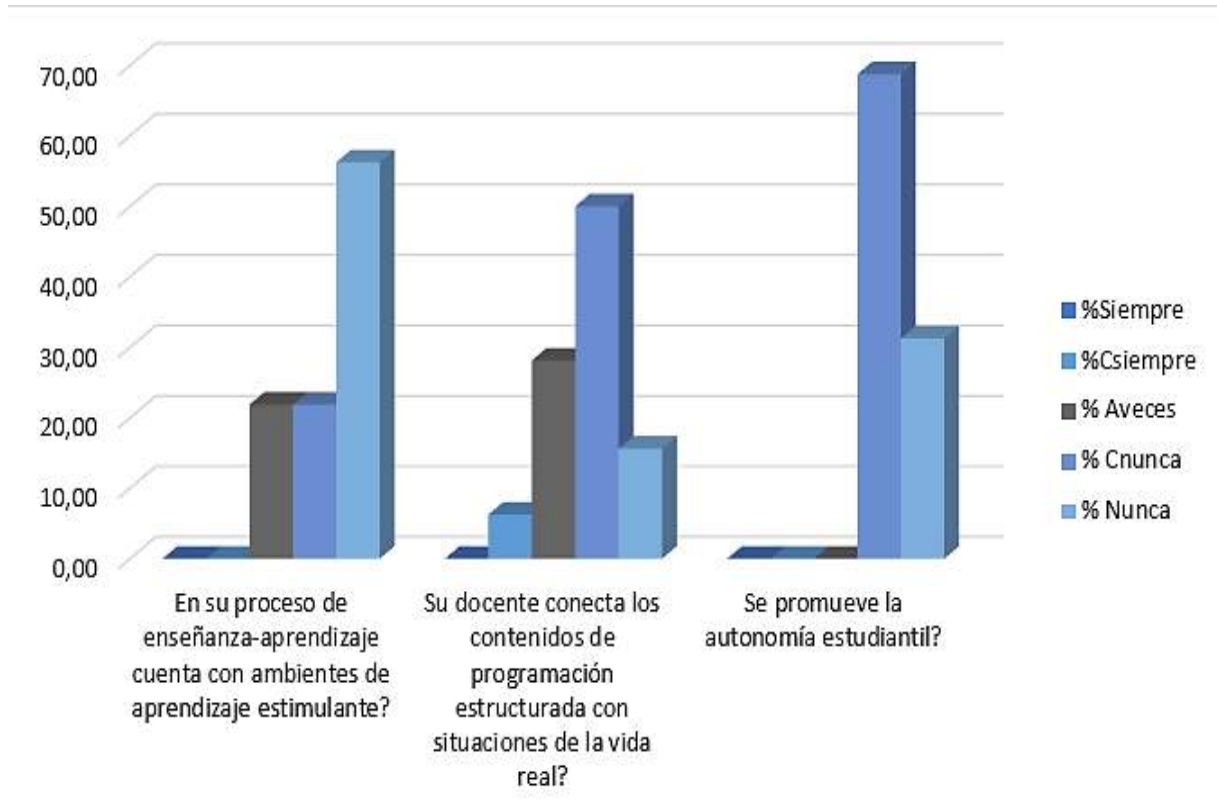
Ítem	VID4. Motivacional	Alternativas				
		% S	%CS	%AV	%CN	%N
9	¿En su proceso de enseñanza-aprendizaje cuenta con ambientes de aprendizaje estimulante?	0,00	0,00	21,88	21,88	56,25
10	¿Su docente conecta los contenidos de programación estructurada con situaciones de la vida real?	0,00	6,25	28,13	50,00	15,63
11	¿Se promueve la autonomía estudiantil?	0,00	0,00	0,00	68,75	31,25
Media		0,00	2,08	16,67	46,88	34,38

Nota. Elaboración propia a partir de los datos del cuestionario aplicado a estudiantes



Figura 4.

Resultados de la dimensión motivacional de la Variable dependiente



Nota. Elaboración propia a partir de los datos del cuestionario aplicado a estudiantes

Como se evidencia en la tabla 5 y figura 4, la mayoría de los aprendices ubicaron su respuesta entre las alternativas casi nunca y nunca para los ítems de esta dimensión motivacional, destacando que el 56,25% expresó que en su proceso de enseñanza-aprendizaje nunca cuenta con ambientes de aprendizaje estimulantes; el 50,00% manifestó que su docente casi nunca conecta los contenidos de programación estructurada con situaciones de la vida real; y el 68,75% de ellos considera que casi nunca su docente promueve la autonomía estudiantil. La media calculada para esta dimensión fue del 46,88% para la alternativa casi nunca, lo que implica que la mayoría de los estudiantes expresaron que sus docentes casi nunca consideran la dimensión motivacional; esto es, casi nunca el docente considera la motivación y el interés del estudiante en aprender programación estructurada, sin considerar indicadores como la creación de un ambiente de



aprendizaje estimulante, la conexión de los contenidos con situaciones de la vida real y la promoción de la autonomía y la autoeficacia del estudiante.

Tabla 6.

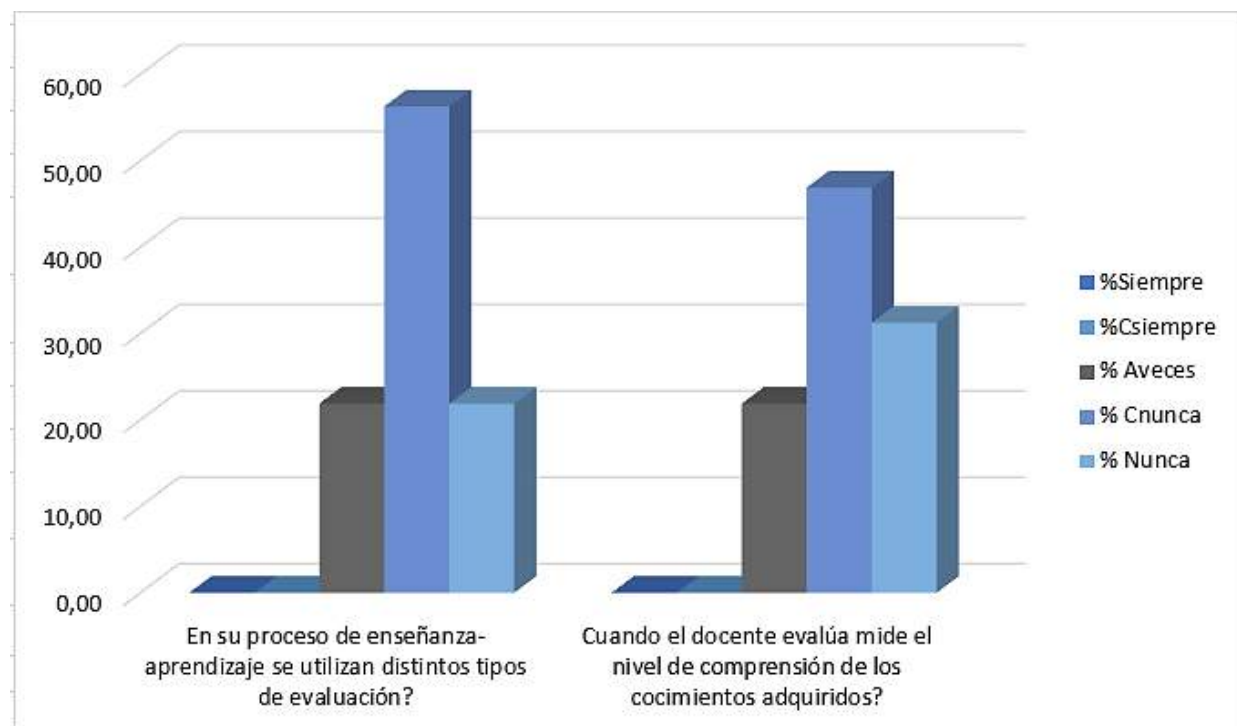
Resultados de la dimensión Evaluativa de la Variable dependiente

Ítem	VID4. Evaluativa	Alternativas				
		% S	%CS	%AV	%CN	%N
12	¿En su proceso de enseñanza-aprendizaje se utilizan distintos tipos de evaluación?	0,00	0,00	21,88	56,25	21,88
13	¿Cuándo el docente evalúa mide el nivel de comprensión de los conocimientos adquiridos?	0,00	0,00	21,88	46,88	31,25
Media		0,00	0,00	21,88	51,56	26,56

Nota. Elaboración propia a partir de los datos del cuestionario aplicado a estudiantes

Figura 5.

Resultados de la dimensión evaluativa de la Variable dependiente



Nota. Elaboración propia a partir de los datos del cuestionario aplicado a estudiantes

Según se evidencia en la tabla 6 y figura 5, la mayoría de los aprendices ubicaron su respuesta entre la alternativa casi nunca para los ítems de esta dimensión evaluativa, destacando que el



56,25% expresó que en su proceso de enseñanza-aprendizaje casi nunca se utilizan distintos tipos de evaluación; y el 46,88% manifestó que cuando el docente evalúa, casi nunca mide el nivel de comprensión de los conocimientos adquiridos. La media calculada para esta dimensión fue del 51,56% para la alternativa casi nunca, lo que implica que la mayoría de los estudiantes expresaron que sus docentes casi nunca consideran la dimensión evaluativa; esto es, casi nunca el docente consideran evaluar el nivel de comprensión; el cual según Velasco & Graus (2024) constituye un aspecto clave en el proceso de aprendizaje y adquisición de conocimientos, ya que se considera la capacidad de comprender y asimilar la información de manera significativa y profunda.

2.10.1. Conclusiones del diagnóstico inicial

La mayoría de los estudiantes expresaron que sus docentes casi nunca consideran la dimensión educativa, la metodológica, la interactiva, la motivacional y la evaluativa; esto es casi nunca:

-Brindan una base sólida de conocimientos teóricos y la oportunidad de aplicarlos en situaciones prácticas, o les prepara para enfrentar desafíos de su cotidianidad y desarrollar habilidades relevantes para su futuro.

-Estos docentes se apoyan en una de las estrategias más comunes, métodos pedagógicos tradicionales, como es el enfoque expositivo, en el cual, el educador presenta información de manera estructurada; sin considerar aspectos relevantes como la introducción de recursos y estrategias novedosas que contemplen el uso de la tecnología.

-Ocurre una interacción adecuada entre el docente y los estudiantes durante el proceso de enseñanza-aprendizaje de programación estructurada.

-Se considera la motivación y el interés del estudiante en aprender programación estructurada, sin considerar indicadores como la creación de un ambiente de aprendizaje estimulante, la conexión de los contenidos con situaciones de la vida real y la promoción de la autonomía y la autoeficacia del estudiante.

-Evalúa el nivel de comprensión; el cual constituye un aspecto clave en el proceso de aprendizaje y adquisición de conocimientos, ya que se considera la capacidad de comprender y asimilar la información de manera significativa y profunda.



CAPÍTULO 3: PRESENTACIÓN Y VALIDACIÓN DE LA PROPUESTA

3.1. Presentación de la propuesta

Este apartado cumple con el objetivo de diseñar una propuesta de una guía que permita el uso de la herramienta “CODING C++” en el mejoramiento del proceso de enseñanza-aprendizaje de la asignatura de programación estructurada en los estudiantes de segundo de bachillerato paralelo “A”, de la Unidad Educativa Amazonas, durante el periodo febrero a octubre de 2024.

La guía contiene seis apartados o capítulos distribuidos en 35 páginas con un diseño llamativo y una estructura constructivista; para su uso se ha sugerido la siguiente estrategia:

1. Se presentan unas preguntas iniciales
2. Se solicita a los estudiantes que intenten darles respuesta a estas preguntas de manera individual
3. Se le pide posteriormente que discuta su respuesta con su grupo de trabajo
4. Luego, un miembro de cada equipo comparte la respuesta con todo el salón
5. El docente da retroalimentación sobre las respuestas
6. Cada grupo entrega un ensayo después de consultar el material presentado en la guía de estudio

3.1.1. Título de la propuesta

Guía para el uso de la Herramienta “CODING C++” en el proceso de enseñanza-aprendizaje de Programación Estructurada (anexo 4).

3.1.2. Presentación

La enseñanza de la programación estructurada es un aspecto fundamental en la formación de futuros profesionales en el ámbito de la informática y la tecnología. En este contexto, el uso de herramientas como “CODING C++” se vuelve esencial para facilitar el aprendizaje de este lenguaje de programación. La creación de una guía para el uso de esta herramienta no solo es importante, sino que también es necesaria para optimizar el proceso de enseñanza-aprendizaje.

La programación es una habilidad esencial en el mundo actual, y el uso de herramientas interactivas puede aumentar la motivación y el aprendizaje efectivo. Esta guía proporciona un marco para implementar “CODING C++” en el aula, con el objetivo de mejorar el proceso de enseñanza-aprendizaje en la asignatura de programación estructurada, fomentando un ambiente de aprendizaje activo y colaborativo.



La programación estructurada implica un enfoque analítico para resolver problemas, al utilizar “CODING C++”, los estudiantes se enfrentan a desafíos que requieren que piensen críticamente y desarrollen soluciones efectivas. Por lo que una guía que incluye ejemplos prácticos y ejercicios específicos puede estimular este tipo de pensamiento, permitiendo a los estudiantes aplicar sus conocimientos de manera creativa y efectiva, esto es fundamental, ya que el pensamiento crítico es una habilidad esencial en cualquier ámbito profesional.

La guía también promueve la autonomía en el aprendizaje. Al proporcionar recursos y estrategias claras, los estudiantes pueden explorar y experimentar con el lenguaje C++ a su propio ritmo. Esto es especialmente importante en la programación, donde la práctica y la experimentación son fundamentales para el dominio del lenguaje. Esta guía empodera a los estudiantes para tomar el control de su aprendizaje, y les permitirá desarrollar confianza en sus habilidades y una mayor disposición para enfrentar desafíos futuros.

Esta guía didáctica representa una respuesta a las necesidades educativas actuales, donde la tecnología juega un papel crucial en el aprendizaje. Al mejorar el proceso de enseñanza-aprendizaje de la programación estructurada, se busca no solo formar estudiantes competentes en el área, sino también prepararlos para enfrentar los desafíos del futuro en un mundo cada vez más digitalizado.

Con el uso de esta guía se facilita la evaluación continua del progreso de los estudiantes; lo cual se logra a través de ejercicios prácticos y proyectos, los docentes pueden monitorear el desarrollo de habilidades y conocimientos, lo que permite realizar ajustes en la enseñanza según sea necesario, esta retroalimentación continua es concluyente para optimizar el rendimiento académico y aseverar que los escolares consigan los objetivos de aprendizaje señalados.

3.1.3. Objetivos generales y específicos

General

Mejorar el proceso de enseñanza-aprendizaje de la asignatura de programación estructurada en estudiantes de segundo de bachillerato mediante el uso de la herramienta “CODING C++”.

Específicos

1. Explorar las funcionalidades de la herramienta “CODING C++”, mediante el uso de programación orientada a objetos y el uso de plantillas y bibliotecas.





2. Develar las aplicaciones, en especial el desarrollo de juegos y aplicaciones gráficas.
3. Utilizar los IDEs (Entornos de Desarrollo Integrados) más populares para facilitar el proceso de desarrollo.

3.1.4. *Fundamentación*

El uso de herramientas digitales en la enseñanza de la programación ha demostrado ser efectivo para aumentar la motivación y el interés de los estudiantes; la gamificación y las plataformas interactivas permiten a los estudiantes experimentar con la programación de manera práctica y divertida, lo que puede resultar en un aprendizaje más significativo; además, estas herramientas ofrecen un entorno de aprendizaje flexible que se adapta a diferentes estilos y ritmos de aprendizaje, promoviendo así la inclusión educativa.

Desde el punto de vista tecnológico, la guía para el uso de "CODING C++" representa una innovación significativa en la enseñanza de la programación estructurada, debido a que esta herramienta permite a los estudiantes interactuar con el código de manera práctica y visual, facilitando la comprensión de conceptos complejos de esta asignatura. La interactividad que ofrece la plataforma fomenta un aprendizaje activo, donde los estudiantes pueden experimentar y aplicar lo aprendido en tiempo real.

Se fundamenta además en la necesidad de adaptar la educación a las nuevas tecnologías y metodologías de enseñanza; y es que en un mundo cada vez más digitalizado, es esencial que los estudiantes desarrollen competencias tecnológicas que les permitan no solo entender la programación, sino también utilizar herramientas modernas que son relevantes en el mercado laboral. "CODING C++" proporciona un entorno que simula situaciones reales de programación, preparando a los estudiantes para futuros desafíos.

Permite la personalización del aprendizaje, ya que los estudiantes pueden avanzar a su propio ritmo y recibir retroalimentación sobre su desempeño; lo cual es esencialmente significativo en la programación, donde los estudiantes pueden tener diferentes niveles de habilidad y comprensión. La capacidad de adaptar el contenido y las actividades a las necesidades individuales de cada estudiante es un aspecto fundamental de la guía.



3.1.5. Características de la propuesta

La propuesta para mejorar el proceso de enseñanza-aprendizaje de la asignatura de programación estructurada mediante el uso de la herramienta "CODING C++" posee varias características que aseveran su efectividad y relevancia, tales como claridad y concisión, relevancia, innovación, estrategias novedosas y beneficios tangibles.

- Claridad y Concisión: La guía es clara y directa en su contenido, evita las ambigüedades conceptuales; en este sentido, los objetivos y beneficios del uso de "CODING C++" se presentan de manera comprensible para todos los usuarios. Una de las características más destacadas de la guía "CODING C++" es su claridad y concisión. En un entorno educativo donde la sobrecarga de información puede llevar a la confusión, esta guía se asegura de presentar los objetivos y beneficios de manera directa y comprensible. Al evitar ambigüedades conceptuales, se permite que los estudiantes, independientemente de su nivel de familiaridad con la programación, comprendan fácilmente el valor y la utilidad de la herramienta. La claridad en la comunicación es esencial para que los estudiantes puedan seguir el material sin perderse en complejidades innecesarias.

- Relevancia: La relevancia de la guía se manifiesta en su atención a las necesidades específicas de los estudiantes de segundo de bachillerato, estos estudiantes se encuentran en una etapa crucial de su educación, donde la comprensión de la programación estructurada puede sentar las bases para futuras carreras en tecnología y ciencias de la computación. La guía muestra cómo "CODING C++" puede mejorar su comprensión de conceptos complejos y facilitar su capacidad de abstracción y resolución de problemas, habilidades esenciales en el mundo moderno.

- Innovación: Se destaca por la innovación que representa el uso de "CODING C++" en comparación con otros recursos característicos de los métodos tradicionales de enseñanza, así, la guía incluye elementos como la interactividad, la gamificación y la personalización del aprendizaje.

- Estrategias novedosas: La guía detalla las estrategias y actividades que se llevarán a cabo para integrar "CODING C++" en el currículo de programación estructurada, esto incluye todos los recursos necesarios; además de su enfoque innovador, la guía también detalla estrategias y actividades específicas para integrar la herramienta en el currículo de programación estructurada.



Estas estrategias son fundamentales para garantizar que la implementación de "CODING C++" sea efectiva y sostenible en el tiempo.

Se identifican los recursos necesarios para implementar "CODING C++" en el aula, incluyendo software, materiales de referencia y plataformas de colaboración en línea, al proporcionar una lista clara de lo que se necesita, se facilita a los educadores la tarea de preparar sus clases y asegurar que todos los estudiantes tengan acceso a las herramientas necesarias para aprender de manera efectiva.

Las actividades diseñadas para complementar la enseñanza también son parte integral de la guía. Estas actividades están diseñadas para ser interactivas y colaborativas, fomentando el trabajo en equipo y la resolución conjunta de problemas, esto no solo enriquece la experiencia de aprendizaje, sino que también prepara a los estudiantes para trabajar en entornos laborales donde la colaboración es fundamental.

- Beneficios Tangibles: Se resaltan los beneficios tangibles que se obtendrán, tanto para los estudiantes como para los docentes, como el aumento en la motivación, la mejora en el rendimiento académico y el desarrollo de habilidades prácticas. Uno de los beneficios más significativos de utilizar esta guía de herramientas educativas modernas es el aumento en la motivación de los estudiantes. Los estudiantes motivados son más propensos a explorar, preguntar y, en última instancia, aprender de manera más efectiva, la motivación intrínseca se ve impulsada cuando los alumnos pueden ver la relevancia de lo que están aprendiendo y cómo se aplica en el mundo real. Así, las herramientas que estimulan esta motivación no solo benefician a los estudiantes, sino que también facilitan la labor de los docentes, quienes encuentran en sus alumnos un mayor compromiso y disposición para aprender.

Otro beneficio tangible que merece mención es el desarrollo de habilidades prácticas en los estudiantes. En un mundo laboral cada vez más competitivo, la capacidad de aplicar conocimientos teóricos en situaciones reales es fundamental. La guía incorpora proyectos prácticos, simulaciones y ejercicios que permiten a los estudiantes desarrollar habilidades técnicas y blandas, como la resolución de problemas, el trabajo en equipo y la comunicación efectiva. Estas habilidades prácticas no solo son valiosas para el ámbito académico, sino que también preparan a los estudiantes para el mercado laboral.



3.1.6. Estructura del contenido de la guía

Contenido a estudiar

1. Introducción a la Programación Estructurada

- Definición y principios de la programación estructurada.
- Importancia de la programación estructurada en el desarrollo de software.
- Comparación con otros paradigmas de programación.

2. Descripción de la Herramienta “CODING C++”

- Tipos de datos y variables.
- Operadores y expresiones.
- Estructuras de control (condicionales y bucles).
- Características y funcionalidades de la herramienta.
- Descripción de la Herramienta: Explicar qué es “CODING C++” y sus características principales, como la interfaz amigable y las funcionalidades interactivas.
- Acceso y Configuración: Instrucciones sobre cómo acceder a la herramienta y configurarla para su uso en el aula.
- Requisitos técnicos para su uso (hardware y software).
- Guía de instalación y configuración.
- Objetivos que se desean alcanzar con el uso de la herramienta “CODING C++”.
- Cómo la herramienta contribuye a mejorar el proceso de enseñanza-aprendizaje.

3. Estructura de un Programa en C++

- Componentes básicos de un programa en C++: directivas, funciones, variables.
- Sintaxis y estructura general de un programa.
- Ejemplos de programas simples.

4. Manejo de Errores y Depuración

- Estrategias para identificar y corregir errores en el código.
- Uso de herramientas de depuración en “CODING C++”.

5. Ejercicios Prácticos

- Propuestas de ejercicios que los estudiantes pueden realizar utilizando la herramienta.
- Proyectos sencillos que integren los conceptos aprendidos.





6. Reflexiones y Recomendaciones

- Reflexiones sobre la importancia de la programación estructurada.
- Consejos para los docentes sobre cómo integrar la herramienta en sus clases.

3.1.7. Metodología para su implementación y evaluación

Estrategias Didácticas:

- Resolución de Problemas: se utiliza la herramienta para presentar problemas que los estudiantes deben resolver, promoviendo el pensamiento crítico. La utilización de herramientas educativas para presentar problemas a los estudiantes ofrece un enfoque interactivo y dinámico que estimula el pensamiento crítico. Estas herramientas pueden incluir software educativo, plataformas de simulación y ejercicios prácticos que plantean desafíos reales o hipotéticos. Al interactuar con estas herramientas, los estudiantes tienen la oportunidad de explorar diferentes métodos para resolver problemas, experimentar con soluciones y reflexionar sobre su proceso de pensamiento. Por ejemplo, en el ámbito de la programación, una herramienta que presenta a los estudiantes un problema de codificación les obliga a analizar el enunciado, identificar los requisitos y desarrollar un algoritmo que resuelva la situación planteada. La resolución de problemas también puede ser un catalizador para el aprendizaje colaborativo. Cuando los estudiantes trabajan juntos para abordar un problema, comparten ideas, discuten enfoques y desafían las opiniones de sus compañeros. Este intercambio de perspectivas no solo enriquece su comprensión del problema, sino que también les enseña a valorar el trabajo en equipo ya desarrollar habilidades interpersonales. La colaboración en la resolución de problemas crea un entorno de aprendizaje donde los estudiantes se sienten apoyados y motivados para explorar nuevas ideas.

- Actividades Sugeridas:

- Ejercicios Interactivos para que los estudiantes puedan resolver, permitiendo la práctica inmediata. - Desafíos Semanales, se proponen retos semanales que incentivan la competencia y el aprendizaje colaborativo.

Los ejercicios interactivos son una herramienta invaluable en el ámbito educativo que permite a los estudiantes practicar de manera inmediata y aplicar sus conocimientos de manera efectiva. La práctica inmediata, el desarrollo de habilidades prácticas y el aumento en la motivación y el compromiso son solo algunos de los beneficios que estos ejercicios ofrecen. A medida que la



educación sigue evolucionando, es esencial que se integren métodos que promuevan la interacción y el aprendizaje activo, preparando a los estudiantes para enfrentar los desafíos del futuro con confianza y competencia.

Evaluación del Aprendizaje

- Establecer criterios claros para evaluar el desempeño de los estudiantes en la guía. Además, - Implementar un sistema de retroalimentación donde los estudiantes puedan recibir comentarios sobre su progreso y áreas de mejora, estos elementos no solo proporcionan claridad y equidad en la evaluación, sino que también fomentan un aprendizaje activo y reflexivo. Al centrarse en la mejora continua y en el desarrollo de habilidades críticas, se prepara a los estudiantes para enfrentar los desafíos del futuro con confianza y competencia. La educación no debe ser solo un proceso de transmisión de conocimientos, sino una experiencia enriquecedora que promueva el crecimiento integral de cada alumno.

Capacitación Docente

- Formación en el Uso de la Herramienta: Proporcionar capacitación a los docentes sobre cómo utilizar “CODING C++” de manera efectiva en sus clases.

- Compartir Buenas Prácticas: Crear un espacio para que los docentes compartan experiencias y estrategias exitosas en el uso de la herramienta.

Monitoreo y Ajustes

- Realizar un seguimiento del uso de “CODING C++” y su impacto en el aprendizaje de los estudiantes.

- Permitir la realización de cambios en la guía, la metodología y las actividades basadas en la retroalimentación de los estudiantes y docentes.

- Evaluar cómo el uso de “CODING C++” ha mejorado la comprensión de la programación estructurada entre los estudiantes.

Perspectivas Futuras

Considerar la posibilidad de integrar otras herramientas tecnológicas que complementen el aprendizaje de la programación.



3.1.8. Recursos

-Recursos humanos

Es fundamental contar con un personal docente capacitado en el uso de "CODING C++".

-Recursos materiales

Los recursos materiales son esenciales para la implementación efectiva de la guía propuesta. Esto incluye:

Equipos Informáticos tales como Computadoras o dispositivos móviles que cumplan con los requisitos técnicos para ejecutar "CODING C++".

Acceso a una red de internet estable y rápida para facilitar el uso de la herramienta y la realización de actividades en línea.

-Recursos didácticos

Guía de Estudio propuesta donde se explican los conceptos básicos de programación estructurada y cómo utilizar "CODING C++" y otros temas relacionados.

3.1.9. Beneficiarios

Estudiantes de segundo año de bachillerato técnico, y docentes que administran la asignatura de programación estructurada en la Unidad Educativa Amazonas.



3.2. Validación de la propuesta

1. ¿La propuesta presentada cumple con las funcionalidades tales como usar programación orientada a objetos; hacer un buen manejo de memoria; y utilizar plantillas y bibliotecas estándar?

Las respuestas dadas por los entrevistados a la interrogante 1 se muestran en la tabla 7

Tabla 7.

Respuestas a la interrogante 1 de la entrevista semiestructurada aplicada a docentes

Docentes	Respuestas
D1	La guía presentada como propuesta cumple con las funcionalidades requeridas, ya que hace referencia a la programación orientada a objetos (POO), la cual permite una mejor organización del código mediante la encapsulación de datos y métodos, esto no solo facilita el manejo de memoria, sino que también permite la reutilización de código a través de clases y objetos; además, el uso de plantillas y bibliotecas estándar en C++ optimiza el desarrollo, ya que proporciona herramientas predefinidas que pueden ser utilizadas para resolver problemas comunes de manera eficiente.
D2	Considero que la guía es adecuada, ya que integra la programación orientada a objetos, lo que es fundamental para proyectos que requieren una estructura modular, esto permite un buen manejo de memoria, ya que los objetos pueden ser creados y destruidos según sea necesario, asimismo, el uso de plantillas y bibliotecas estándar en C++ es esencial, ya que no solo mejora la eficiencia del código, sino que también permite a los estudiantes familiarizarse con herramientas que son ampliamente utilizadas en la industria.
D3	La guía diseñada cumple con las expectativas al incorporar la programación orientada a objetos, lo que facilita la creación de aplicaciones más complejas y escalables. Es importante destacar que el manejo de memoria se optimiza al utilizar objetos, ya que se pueden gestionar de manera más efectiva. Además, el uso de plantillas y bibliotecas estándar en C++ es un gran acierto, ya que permite a los estudiantes aprender a utilizar recursos que simplifican el desarrollo y mejoran la calidad del código, lo cual es crucial en la formación de futuros programadores.

Nota. Elaboración propia según respuestas de la entrevista

En la tabla 7, las respuestas ofrecen una evaluación positiva de la guía propuesta, destacando aspectos clave de la programación orientada a objetos, el manejo de memoria y el uso de plantillas y bibliotecas estándar. Todas las respuestas enfatizan la importancia de la programación



orientada a objetos, lo cual es fundamental en el desarrollo de software moderno; la POO permite una mejor organización del código y facilita la creación de aplicaciones complejas, lo que es un punto fuerte en la propuesta. Se menciona que la POO optimiza el manejo de memoria al permitir la creación y destrucción de objetos según sea necesario. Esto es crucial, ya que un buen manejo de memoria es esencial para evitar fugas y mejorar el rendimiento de las aplicaciones. La referencia al uso de plantillas y bibliotecas estándar en C++ es un aspecto positivo, ya que estas herramientas permiten a los desarrolladores resolver problemas comunes de manera eficiente y mejorar la calidad del código.

La importancia de que la guía presentada cumpla con las funcionalidades como el uso de programación orientada a objetos (POO), un buen manejo de memoria y la utilización de plantillas y bibliotecas estándar radica en varios aspectos clave que impactan tanto en el proceso de enseñanza-aprendizaje como en el desarrollo de software. La programación orientada a objetos permite una mejor organización del código al agrupar datos y funciones en objetos, esto no solo facilita la comprensión y el mantenimiento del código, sino que también promueve la modularidad, lo que permite a los equipos trabajar en diferentes partes del proyecto simultáneamente sin interferencias. Esta estructura es fundamental en proyectos grandes, donde la complejidad puede aumentar rápidamente.

Por otra parte, un buen manejo de memoria es crucial en el desarrollo de software, ya que ayuda a prevenir problemas como fugas de memoria y optimiza el rendimiento de las aplicaciones. La POO permite gestionar la memoria de manera más efectiva, ya que los objetos pueden ser creados y destruidos según sea necesario, lo que contribuye a un uso más eficiente de los recursos del sistema. El uso de plantillas y bibliotecas estándar en C++ es esencial porque facilita la reutilización de código. Esto no solo ahorra tiempo de desarrollo, sino que también mejora la calidad del software al permitir que los desarrolladores utilicen soluciones probadas y optimizadas para problemas comunes. La capacidad de reutilizar código es especialmente valiosa en entornos de desarrollo ágil, donde los cambios son frecuentes y se requiere adaptabilidad.

Integrar estas funcionalidades en la guía también prepara a los estudiantes para el mundo laboral. Familiarizarse con la POO y las bibliotecas estándar les proporciona habilidades que son altamente valoradas en la industria del software. Esto no solo aumenta su empleabilidad, sino



que también les permite adaptarse más rápidamente a las herramientas y tecnologías que encontrarán en sus futuras carreras.

2. ¿Cómo observa el manejo de aplicaciones en la propuesta de guía presentada; con respecto a ofrecer desarrollo de juegos y aplicaciones gráficas; manejo de grandes cantidades de datos de manera eficiente; y la evidencia de una respuesta rápida y uso eficiente de los recursos del sistema?

Las respuestas dadas por los entrevistados a la interrogante 2 se muestran en la tabla 8

Tabla 8.

Respuestas a la interrogante 2 de la entrevista semiestructurada aplicada a docentes

Docentes	Respuestas
D1	En la propuesta de guía presentada, observo que el manejo de aplicaciones está bien fundamentado, especialmente en el desarrollo de juegos y aplicaciones gráficas, la implementación de la programación orientada a objetos permite crear estructuras más organizadas y escalables, lo que es esencial para manejar la complejidad de los gráficos y la lógica de los juegos. Además, el enfoque en el manejo eficiente de grandes cantidades de datos es crucial, ya que muchas aplicaciones gráficas requieren procesar información en tiempo real, esto, combinado con una respuesta rápida y un uso eficiente de los recursos del sistema, asegura que las aplicaciones funcionen de manera fluida y sin interrupciones.
D2	La guía presentada aborda de manera efectiva el manejo de aplicaciones, especialmente en el contexto del desarrollo de juegos y aplicaciones gráficas, la programación orientada a objetos facilita la creación de componentes reutilizables, lo que es fundamental para optimizar el rendimiento en aplicaciones que manejan grandes volúmenes de datos. Además, la propuesta enfatiza la importancia de una respuesta rápida del sistema, lo que es vital para mantener la experiencia del usuario en juegos interactivos, esto se traduce en un uso eficiente de los recursos, permitiendo que las aplicaciones funcionen sin problemas incluso en dispositivos con limitaciones de hardware.
D3	Al analizar la propuesta de guía, considero que el manejo de aplicaciones está bien diseñado para ofrecer un desarrollo efectivo de juegos y aplicaciones gráficas. La inclusión de la programación orientada a objetos no solo mejora la organización del código, sino que también permite un manejo más eficiente de grandes cantidades de datos, lo cual es esencial en este tipo de aplicaciones. Además, la guía destaca la necesidad de una respuesta rápida y un uso eficiente de los recursos del sistema, lo que garantiza que las aplicaciones sean responsivas y ofrezcan una experiencia de usuario satisfactoria.

Nota. Elaboración propia según respuestas de la entrevista



En la tabla 8, las respuestas ofrecen algunas semejanzas y diferencias entre las que se destacan las siguientes: Todas las respuestas mencionan que la programación orientada a objetos es concluyente para crear estructuras organizadas y escalables, este enfoque es destacado como fundamental para manejar la complejidad de los gráficos y la lógica de los juegos. Las tres respuestas enfatizan la importancia de un manejo eficiente de grandes cantidades de datos. Se menciona que este aspecto es esencial para el rendimiento de aplicaciones gráficas que requieren procesar información en tiempo real. Cada respuesta subraya la necesidad de una respuesta rápida del sistema, este aspecto es vital para asegurar que las aplicaciones funcionen de manera fluida y proporcionen una experiencia de usuario satisfactoria, especialmente en juegos interactivos.

Entre las diferencias se encuentran que la primera respuesta se centra más en el fundamento del manejo de aplicaciones y cómo la POO ayuda a manejar la complejidad. La segunda respuesta se enfoca en la creación de componentes reutilizables y cómo estas prácticas optimizan el rendimiento. La tercera respuesta destaca el diseño de la guía en su conjunto, lo que implica una evaluación más amplia de su efectividad.

Adicionalmente, la segunda respuesta hace énfasis en el hecho de que las aplicaciones pueden funcionar sin problemas incluso en dispositivos con limitaciones de hardware, lo que no se menciona explícitamente en las otras respuestas, esto sugiere una consideración adicional sobre la accesibilidad y el rendimiento en diferentes plataformas. La tercera respuesta concluye que el manejo de aplicaciones está bien diseñado para el desarrollo efectivo de juegos y aplicaciones gráficas, lo que implica una evaluación positiva más explícita sobre la guía; las otras respuestas se centran más en los elementos individuales que componen la guía.

Estas respuestas implican que la guía presentada está muy bien elaborada; es decir, que no solo se centra en aspectos técnicos, sino que también considera la experiencia del usuario, la eficiencia en el manejo de datos, la rapidez de respuesta y la adaptabilidad.



3. Con referencia a los entornos de desarrollo de la guía, ¿Cuál es su opinión sobre la forma como se utilizan los IDEs (Entornos de Desarrollo Integrados) más populares para facilitar el proceso de desarrollo?

Las respuestas dadas por los entrevistados a la interrogante 3 se muestran en la tabla 9

Tabla 9.

Respuestas a la interrogante 3 de la entrevista semiestructurada aplicada a docentes

Docentes	Respuestas
D1	La guía para el uso de la herramienta "CODING C++" hace un excelente trabajo al integrar los IDE más populares en el proceso de enseñanza-aprendizaje. Estas herramientas permiten a los estudiantes enfocarse en la lógica de programación sin distraerse con la configuración técnica. La funcionalidad de autocompletado y la depuración visual son características que no solo facilitan el aprendizaje, sino que también ayudan a los alumnos a identificar y corregir errores de manera más eficiente.
D2	La forma en que se utilizan los IDE en la guía es muy beneficiosa para los estudiantes, ofrecen una interfaz amigable que simplifica el proceso de programación. La capacidad de compilar y ejecutar el código con un solo clic, junto con la gestión de proyectos, permite a los alumnos centrarse en aprender los conceptos fundamentales de la programación estructurada; además, la integración de herramientas de documentación y recursos en línea dentro de estos entornos fomenta un aprendizaje más autónomo y efectivo.
D3	A mi parecer, la guía aprovecha de manera efectiva los IDE más populares para facilitar el aprendizaje de C++; no solo proporcionan herramientas de edición y depuración, sino que también permiten a los estudiantes experimentar con diferentes configuraciones y bibliotecas. Esto es crucial para desarrollar habilidades prácticas en programación. La interfaz gráfica y las funcionalidades de arrastrar y soltar hacen que el proceso de desarrollo sea más accesible, lo que puede motivar a los estudiantes a explorar más allá de los conceptos básicos.

Nota. Elaboración propia según respuestas de la entrevista

En la tabla 9, las respuestas ofrecen algunas semejanzas y diferencias entre las que se destacan: las tres respuestas coinciden en que la guía para "CODING C++" mejora significativamente el aprendizaje de los estudiantes al utilizar los IDE más populares; todas destacan que estas herramientas permiten a los alumnos centrarse en la lógica de programación y el desarrollo de habilidades prácticas.



Las respuestas mencionan características específicas de los IDE, como la depuración visual, el autocompletado y la interfaz amigable, dichas características son vistas como elementos que facilitan el proceso de aprendizaje, permitiendo a los estudiantes identificar y corregir errores de manera más eficiente. Enfatizan que el uso de los IDE hace que el desarrollo de programas sea más accesible y, por lo tanto, puede motivar a los estudiantes a explorar más allá de los conceptos básicos, esto sugiere que los entornos de desarrollo integrados no solo son herramientas técnicas, sino que también influyen en la actitud de los estudiantes hacia el aprendizaje.

Con relación a las diferencias, la segunda respuesta se centra más en la interfaz y amigable en la capacidad de compilar y ejecutar el código con un solo clic, enfatizando la simplicidad del proceso de programación; mientras que la primera y la tercera respuesta mencionan la lógica de programación y la experimentación, la segunda se detiene más en la usabilidad inmediata de los IDE.

La segunda respuesta menciona específicamente la integración de herramientas de documentación y recursos en línea, resaltando cómo esto fomenta un aprendizaje más autónomo. En contraste, las otras respuestas no hacen énfasis en este aspecto, centrándose más en las características técnicas de los IDE. Por otra parte, la tercera respuesta se distingue al hablar sobre la posibilidad de experimentar con diferentes configuraciones y bibliotecas, resaltando la importancia de estas experiencias prácticas en el desarrollo de habilidades, esto no se menciona de manera explícita en las otras respuestas, que se enfocan más en las características de los IDE y su impacto en el aprendizaje.

Los IDE tienen gran importancia ya que proporcionan la facilidad del aprendizaje en un entorno unificado donde los estudiantes pueden escribir, compilar y depurar su código de manera eficiente, esto permite que los alumnos se concentren en aprender los conceptos de programación sin distraerse con la configuración técnica del entorno. La funcionalidad de autocompletado y la depuración visual son características que ayudan a los estudiantes a identificar y corregir errores de forma más rápida y efectiva.

La mayoría de los IDE ofrecen interfaces intuitivas que simplifican el proceso de programación. Esto es especialmente beneficioso para los principiantes, ya que les permite familiarizarse con el lenguaje C++ y sus estructuras sin sentirse abrumados por la complejidad de las herramientas.



La capacidad de compilar y ejecutar el código con un solo clic también contribuye a una experiencia de aprendizaje más fluida. Los IDE suelen integrar herramientas de documentación y recursos en línea, lo que fomenta un aprendizaje más autónomo.

Los estudiantes pueden acceder a guías, tutoriales y foros de discusión directamente desde el entorno de desarrollo, lo que les permite resolver dudas y mejorar su comprensión de los temas tratados. Al permitir a los estudiantes experimentar con diferentes configuraciones y bibliotecas, los IDE contribuyen al desarrollo de habilidades prácticas en programación. Esto es crucial para que los alumnos no solo comprendan la teoría, sino que también puedan aplicar lo aprendido en proyectos reales.

4. ¿Cómo la guía diseñada, utiliza los recursos de aprendizaje disponibles para C++; y ofrece comunidades y grupos en línea?

Las respuestas dadas por los entrevistados a la interrogante 4 se muestran en la tabla 10

Tabla 10.

Respuestas a la interrogante 4 de la entrevista semiestructurada aplicada a docentes

Docentes	Respuestas
D1	La guía diseñada para C++ hace un uso excelente de los recursos de aprendizaje disponibles, integra tutoriales en vídeo, documentación interactiva y ejercicios prácticos que facilitan la comprensión de los conceptos. Además, la inclusión de comunidades y grupos en línea permite a los estudiantes interactuar con otros programas.
D2	Opino que la guía para C++ está muy bien estructurada en cuanto al uso de recursos de aprendizaje, proporciona enlaces a plataformas en línea, foros de discusión y grupos en redes sociales donde los estudiantes pueden conectarse con otros y obtener retroalimentación sobre sus proyectos, esta interacción en comunidades en línea no solo fomenta el aprendizaje colaborativo, sino que también ofrece oportunidades para que los estudiantes se mantengan actualizados con las últimas tendencias y prácticas en programación.
D3	Utiliza eficazmente los recursos de aprendizaje disponibles, así, la combinación de materiales escritos, videos y ejercicios interactivos permite a los estudiantes aprender a su propio ritmo; además, la promoción de comunidades y grupos en línea es una gran ventaja, ya que ofrece un espacio donde los alumnos pueden hacer preguntas, compartir sus proyectos y recibir apoyo de otros programadores.

Nota. Elaboración propia según respuestas de la entrevista



En la tabla 10, las respuestas ofrecen algunas semejanzas y diferencias entre las que se destacan la coincidencia en que la guía para C++ hace un uso efectivo de los recursos de aprendizaje disponibles; todas mencionan la integración de tutoriales en video, documentación interactiva y ejercicios prácticos como elementos clave que facilitan la comprensión de los conceptos. Enfatizan la importancia de las comunidades y grupos en línea, se señala que estas plataformas permiten a los estudiantes interactuar con otros programadores, lo que fomenta el aprendizaje colaborativo y ofrece apoyo adicional a los alumnos. Las respuestas resaltan que la interacción en comunidades en línea no solo ayuda a los estudiantes a resolver dudas, sino que también les permite mantenerse actualizados con las últimas tendencias y prácticas en programación, esto subraya la importancia de la colaboración en el proceso educativo.

Con relación a las diferencias; la segunda respuesta se centra en la estructura de la guía, señalando que está bien organizada en cuanto al uso de recursos de aprendizaje, mientras que las otras respuestas mencionan la efectividad de los recursos, no abordan específicamente la estructura de la guía. La primera respuesta menciona específicamente la combinación de tutoriales en video, documentación interactiva y ejercicios prácticos, mientras que la segunda y tercera respuestas son más generales al referirse a recursos de aprendizaje.

La tercera respuesta también resalta que los materiales permiten a los estudiantes aprender a su propio ritmo, un aspecto que no se menciona en las otras respuestas. Por otra parte, la segunda respuesta destaca específicamente la obtención de retroalimentación sobre proyectos a través de foros de discusión y redes sociales, aunque las otras respuestas mencionan la interacción, no enfatizan tanto este aspecto de recibir retroalimentación, lo que puede ser crucial para el aprendizaje práctico.

La importancia de guía diseñada para C++ es reconocida en estas respuestas; al respecto, la integración de recursos de aprendizaje como tutoriales en video, documentación interactiva y ejercicios prácticos permite a los estudiantes comprender mejor los conceptos de programación, estos recursos son esenciales para que los alumnos puedan aprender a su propio ritmo y reforzar su comprensión a través de la práctica.

La inclusión de comunidades y grupos en línea ofrece a los estudiantes la oportunidad de interactuar con otros programadores, dicha interacción no solo fomenta el aprendizaje



colaborativo, sino que también permite a los estudiantes compartir experiencias, resolver dudas y recibir retroalimentación sobre sus proyectos, lo cual es fundamental para el desarrollo de habilidades prácticas y para mantenerse actualizados con las tendencias en programación.

Al proporcionar enlaces a plataformas en línea y foros de discusión, la guía amplía el acceso a una variedad de recursos educativos, lo cual hace posible a los estudiantes explorar diferentes enfoques y técnicas, enriqueciendo su aprendizaje y ayudándoles a adaptarse a diversas situaciones en el campo de la programación. La promoción de comunidades en línea también apoya el aprendizaje autónomo, ya que los estudiantes pueden buscar ayuda y orientación cuando lo necesiten, desarrollando una mentalidad proactiva hacia el aprendizaje.

5. ¿Considera válida la propuesta de guía presentada?

Las respuestas dadas por los entrevistados a la interrogante 5 se muestran en la tabla 11.

Tabla 11.

Respuestas a la interrogante 5 de la entrevista semiestructurada aplicada a docentes

Docentes	Respuestas
D1	Considero que la propuesta de guía presentada es válida y muy bien estructura, proporciona un enfoque claro y accesible para el aprendizaje de C++, integrando recursos que facilitan la comprensión de los conceptos fundamentales; además, la guía incluye ejemplos prácticos y ejercicios que permiten a los estudiantes aplicar lo aprendido, lo que es esencial para consolidar sus conocimientos, la claridad en la presentación de la información y la organización.
D2	La propuesta de guía es completamente válida, la forma en que se presentan los contenidos es lógica y progresiva, lo que ayuda a los estudiantes a construir su conocimiento de manera efectiva; además, la inclusión de recursos interactivos y ejercicios prácticos permite a los alumnos experimentar con el lenguaje C++ de una manera dinámica. Esto no solo hace que el aprendizaje sea más atractivo, sino que también fomenta una comprensión más profunda de los conceptos, lo cual es crucial en la programación estructurada.
D3	Opino que la propuesta de guía es válida y pertinente para el aprendizaje de C++, la guía está diseñada de manera que los estudiantes puedan seguir un camino claro desde los conceptos básicos hasta temas más avanzados; la utilización de recursos de aprendizaje variados, como tutoriales y ejercicios prácticos, enriquece la experiencia educativa.

Nota. Elaboración propia según respuestas de la entrevista



En la tabla 11, las respuestas ofrecen algunas semejanzas y diferencias entre las que se destacan que las tres respuestas reconocen la validez y efectividad de la guía propuesta para el aprendizaje de C++. Sin embargo, cada respuesta aporta un enfoque específico, ya sea en la dinámica de aprendizaje, la claridad y organización de la información, o el enriquecimiento de la experiencia educativa, estas diferencias y similitudes enriquecen el análisis sobre la propuesta de guía y su impacto en la educación en programación estructurada.

Con relación a las semejanzas, las respuestas coinciden en considerar la propuesta de guía como válida y adecuada para el aprendizaje de C++, esto insta un consenso sobre la efectividad de la guía en el proceso educativo. Todas las respuestas destacan la importancia de la estructura clara y accesible de la guía; se menciona que la presentación de los contenidos es lógica y bien organizada, lo que facilita la comprensión y el seguimiento por parte de los estudiantes. Enfatizan la inclusión de recursos prácticos, como ejemplos, ejercicios y recursos interactivos, lo cual es visto como esencial para permitir a los estudiantes aplicar lo que han aprendido y consolidar sus conocimientos de manera efectiva. Las respuestas también mencionan que la guía permite a los estudiantes avanzar desde conceptos básicos hasta temas más avanzados, lo que es fundamental para un aprendizaje efectivo en programación.

Entre sus diferencias se encontró que la segunda respuesta se centra más en cómo la inclusión de recursos interactivos y ejercicios prácticos hace que el aprendizaje sea dinámico y atractivo, mientras que las otras respuestas no enfatizan tanto este aspecto. La primera respuesta hace hincapié en la claridad y la organización de la presentación de la información, mientras que la segunda destaca la lógica y progresión en la exposición de los contenidos; la tercera respuesta menciona ambos aspectos, pero no profundiza en uno en particular. La tercera respuesta menciona específicamente cómo la utilización de recursos variados enriquece la experiencia educativa, lo que no se aborda de manera tan explícita en las otras respuestas, esto sugiere un enfoque más holístico hacia el aprendizaje.



Conclusiones de la validación de la propuesta

La propuesta de guía para el uso de la herramienta "CODING C++" en el proceso de enseñanza-aprendizaje de programación estructurada ha sido evaluada positivamente y validada por diversos expertos en el campo, quienes recomiendan su utilización. A continuación, se presentan las conclusiones clave.

La guía está diseñada de manera que los estudiantes puedan seguir un camino claro desde los conceptos básicos hasta temas más avanzados, esta estructura lógica y progresiva facilita la construcción del conocimiento de manera efectiva, permitiendo a los estudiantes avanzar a su propio ritmo y consolidar sus habilidades de programación.

La inclusión de ejemplos prácticos, ejercicios interactivos y recursos variados, como tutoriales en video y documentación, enriquece la experiencia educativa, estos elementos no solo facilitan la comprensión de los conceptos fundamentales, sino que también permiten a los estudiantes aplicar lo aprendido en situaciones prácticas, lo cual es esencial para el aprendizaje de la programación.

Promueve la interacción en comunidades y grupos en línea, lo que fomenta el aprendizaje colaborativo, esta interacción no solo ayuda a los estudiantes a resolver dudas y recibir retroalimentación sobre sus proyectos, sino que también les permite mantenerse actualizados con las últimas tendencias y prácticas en programación.

La forma en que se presentan los contenidos hace que el aprendizaje sea más atractivo y accesible para los estudiantes, la claridad en la presentación de la información y la organización de los contenidos son aspectos que benefician el proceso de enseñanza-aprendizaje, motivando a los estudiantes a explorar más allá de los conceptos básicos.



CONCLUSIONES

En esta investigación se generó una propuesta para el uso de la herramienta “CODING C++” en el mejoramiento del proceso de enseñanza-aprendizaje de la asignatura de programación estructurada en los estudiantes de segundo de bachillerato paralelo “A”, de la Unidad Educativa Amazonas, durante el periodo febrero a octubre de 2024, la cual consistió en una guía para ser usada por docentes y estudiantes. Al respecto se concluye:

1. Se determinaron los fundamentos teóricos de fundamento al uso de la herramienta “CODING C++” en el proceso de enseñanza-aprendizaje de programación estructurada, donde se definieron las dimensiones que conforman las variables de estudio.
2. Al caracterizar el proceso de enseñanza-aprendizaje de la asignatura de programación estructurada en los estudiantes de segundo de bachillerato paralelo “A” de la Unidad Educativa Amazonas, durante el periodo de febrero a octubre de 2024, antes de la implementación de la herramienta “CODING C++”, se concluyó que una colectividad de aprendices expresaron que sus docentes casi nunca consideran la dimensión educativa, la metodológica, la interactiva, la motivacional y la evaluativa; esto es, casi nunca: brindan una estructura consistente de sapiencias teóricas y la oportunidad de aplicarlos en situaciones prácticas, o les prepara para enfrentar desafíos de su cotidianidad y desarrollar habilidades relevantes para su futuro. Estos docentes se apoyan en una de las estrategias estimadas tradicionales, como es el enfoque expositivo; se considera la estimulación y beneficio del aprendiz para formarse en programación estructurada, sin reflexionar sobre la generación de entornos de lucubración alentadores, la conexión de los contenidos con circunstancias reales y el fomento de la emancipación y autoeficacia del estudiante.
3. Se diseñó una propuesta de una guía que permita el uso de la herramienta “CODING C++” en el mejoramiento del proceso de enseñanza-aprendizaje de la asignatura de programación estructurada en los estudiantes de segundo de bachillerato paralelo “A”, de la Unidad Educativa Amazonas, durante el periodo febrero a octubre de 2024, titulada: “Guía para el uso de la Herramienta “CODING C++” en el proceso de enseñanza-aprendizaje de Programación Estructurada”; la guía contiene seis apartados o capítulos distribuidos en 35 páginas con un diseño llamativo y una estructura constructivista.



4. Al validar la propuesta diseñada para el uso de la herramienta “CODING C++” en el mejoramiento del proceso de enseñanza-aprendizaje de la asignatura de programación estructurada en los estudiantes de segundo de bachillerato paralelo “A”, de la Unidad Educativa Amazonas, durante el periodo febrero a octubre de 2024, se concluye que ha sido evaluada positivamente y validada por diversos expertos en el campo, quienes recomiendan su utilización. A continuación, se presentan las conclusiones clave.

La guía estuvo esbozada de forma que los aprendices logren continuar una calzada despejada desde nociones primordiales hasta tramas evolucionadas, esta estructura lógica y progresiva suministra la cimentación de la sapiencia de modo efectivo, permitiendo a los aprendices progresar a su ritmo y consolidar sus habilidades de programación. La inserción de ejemplos versados, ejercicios interactivos y recursos variados, como tutoriales en video y documentación, enriquece la experiencia educativa, estos elementos además de proporcionar perspicacia de concepciones primordiales, también consienten poder utilizar lo estudiado en condiciones prácticas.



RECOMENDACIONES

En atención a los resultados y limitaciones de esta investigación, se ostentan recomendaciones para futuras investigaciones las cuales pueden ahondar en aspectos concretos coherentes con la utilización de “CODING C++” en el progreso del proceso de enseñanza-aprendizaje de programación estructurada.

1. Implementar la Guía para el uso de la Herramienta “CODING C++” en el proceso de enseñanza-aprendizaje de Programación Estructurada, propuesta en esta investigación.
2. Animar la participación de docentes, mediante el cuestionamiento, y la deliberación acerca de la propuesta presentada.
3. Explorar la incorporación de otras tecnologías complementarias, como los IDEs, simuladores o plataformas de codificación en línea, estas herramientas pueden ofrecer a los estudiantes experiencias adicionales y prácticas que enriquecen su aprendizaje y les permiten experimentar con disímiles talentos de programación.
4. Continuar realizando estudios que muestren la incidencia del uso de la guía propuesta en la lucubración de aprendices de la asignatura objeto de investigación.



REFERENCIAS BIBLIOGRÁFICAS

- Abril, C. A., Martínez, J. S., & Sánchez, D. S. (2023). *Implementación de unidad didáctica de programación Básica en C++ para estudiantes de sexto de Colegio Santo Tomás de Aquino, Bogotá*. Colegio Santo Tomás de Aquino.
- Alcívar, H. S., & Lemos, D. K. (2024). Innovación Educativa para la solución de problemas en el Proceso de Enseñanza – Aprendizaje en la Unidad Educativa Fiscal “Eloy Alfaro”, ciudad Esmeraldas, Ecuador. *Revista Social Fronteriza*, 4(3).
[https://doi.org/https://doi.org/10.59814/resofro.2024.4\(3\)259](https://doi.org/https://doi.org/10.59814/resofro.2024.4(3)259)
- Anchundia, K. A., & Leal, D. A. (2023). La gestión administrativa para la enseñanza a estudiantes con capacidades específicas. *MQRInvestigar*, 7(4), 2493–2514.
<https://doi.org/https://doi.org/10.56048/MQR20225.7.4.2023.2493-2514>
- Arboleda, M. M. (2024). Desarrollo del Pensamiento Lógico-Matemático y su relación con las Prácticas Pedagógicas. *Ciencia Latina Revista Científica Multidisciplinar*, 8(1), 4556-4565. https://doi.org/https://doi.org/10.37811/cl_rcm.v8i1.9794
- Arenas, A. C. (2021). *Métodos mixtos de investigación*. Magisterio.
- Calzadilla, J. C. (2022). Sistemas de tareas docentes para la enseñanza y aprendizaje de la Programación Orientada a Objetos. *Universidad y Sociedad*, 14(S6), 480-491.
- Cárdenas, J. A., Rodríguez, C. G., Pérez, J. A., & Valencia, X. H. (2022). Desarrollo del pensamiento crítico: Metodología para fomentar el aprendizaje en ingeniería. *Revista de ciencias sociales*, 28(4), 512-530.
<https://doi.org/https://dialnet.unirioja.es/servlet/articulo?codigo=8703859>
- Centeno, J. F. (2023). *Adaptación de Google Slides para el aprendizaje de Biología Celular, con estudiantes de segundo semestre de la Carrera de Pedagogía de las Ciencias Experimentales Química y Biología (Bachelor's thesis)*. Riobamba.
- Chaverra, A. F. (2023). Filosofía para la comprensión: Modelo EPC para la enseñanza de la filosofía en Colombia. *Simbiosis Educativa*, 2(1), 22-34.
<https://doi.org/https://doi.org/10.60085/se.v2n1a2>



- Chimarro, N. R., Merino, A. C., Moreno, M. M., Romero, J. A., & Alarcón, E. E. (2023). Actividades Cooperativas y Colaborativas para fortalecer el aprendizaje. *Ciencia Latina Revista Científica Multidisciplinar*, 7(3), 3688-3707. https://doi.org/https://doi.org/10.37811/cl_rcm.v7i3.6434
- Coapaza, M. Y., Cariapaza, G. J., Díaz, Y. D., & Condori, W. W. (2024). *Aprendizaje Activo y Participativo en el Aula*. Editorial Idicap Pacífico. <https://doi.org/https://doi.org/10.53595/eip.015.2024>
- Constitución de la República de Ecuador. (2008). Registro Oficial 449 de 20-oct-2008. *Ultima modificación: 30-ene-2012*.
- Costanzo, M. (2024). *Estudio de viabilidad de SYCL como modelo de programación unificado para sistemas heterogéneos basados en GPUs en bioinformática (Doctoral dissertation)*. Universidad Nacional de La Plata.
- Cruz, C. E., Tamani, E. R., & Peralta, S. R. (2023). Pedagogía de la venerable María Bordoní: fomentando el aprendizaje significativo a través de la pedagogía de las pequeñas cosas. *Ciencia Latina Revista Científica Multidisciplinar*, 7(3), 6649-6672. https://doi.org/https://doi.org/10.37811/cl_rcm.v7i3.6660
- Cu Hao, L., Buena Delantar, M., & Tan, G. J. (2021). C++ Adventure: A Mixed Methods Pilot Study on Digital Game-Based Learning of Coding and Effect on Motivation. *Proceedings of the 2021 4th International Conference on Data Storage and Data Engineering (DSDE '21)*. New York, NY, USA: Association for Computing Machinery. <https://doi.org/https://doi.org/10.1145/3456146.3456162>
- Dávila, J. D. (2024). Valor agregado de la investigación en la educación de pregrado: un análisis de sus beneficios. *Revista Guatemalteca de Educación Superior*, 7(1), 1-17. <https://doi.org/https://doi.org/10.46954/revistages.v7i1.122>
- De Parga, C. J. (2021). *UML. Arquitectura de aplicaciones en Java, C++ y Python*. Ra-Ma Editorial.
- Domínguez, J. P. (2021). *Plantilla parallel_for heterogénea implementada en INTEL ONEAPI*. Universidad de Málaga.



- Gallegos, N. A. (2023). Chatgpt in the learning of programming languages by university students. *Revista Amazonía Digital*, 2(2), 1-6. <https://doi.org/https://doi.org/10.55873/rad.v2i2.250>
- García, A. G., Escolano, S. O., Canales, J. M., & Rodríguez, J. G. (2020). *Programación de GPUs: Usando Compute Unified Device Architecture (CUDA)*. Ra-Ma Editorial.
- García, B. L. (2021). *Diseño de software para el uso de estructuras dinámicas con lenguaje de programación C++ (Tesis de maestría)*. Universidad Nacional del Callao.
- García, G. (2024). La evaluación como herramienta para mejorar los aprendizajes: la retroalimentación y la evaluación auténtica. *Revista Latinoamericana Ogmios*, 4(9), 17–32. <https://doi.org/https://doi.org/10.53595/rlo.v4.i9.091>
- Gutiérrez, G. F., Borja, Y. A., Zapata, V. H., Rengel, V. I., & Granda, R. L. (2024). Realidades Extendidas: El futuro de la enseñanza y el aprendizaje interactivo. *Revista Científica Y Tecnológica VICTEC*, 5(8), 126–152. <https://doi.org/https://doi.org/10.61395/victec.v5i8.157>
- Idrovo, F. X. (2023). La formación de competencias profesionales en la educación superior mediante la aplicación de estrategias didácticas y las TIC: Una realidad necesaria en los Institutos tecnológicos de la provincia del Cañar. *Ciencia Latina Revista Científica Multidisciplinar*, 6(6), 11807-11818. https://doi.org/https://doi.org/10.37811/cl_rcm.v6i6.4229
- Machuca, S. A., Robalino, J. L., Benavides, C. F., & Palma, D. P. (2023). Estudio sobre el uso efectivo de las TICs en la educación y el desarrollo de competencias digitales de los docentes. *Revista Conrado*, 19(S2), 355–361. <https://doi.org/https://conrado.ucf.edu.cu/index.php/conrado/article/view/3271>
- Martín, M. V., Urquía, U. M., & Rubio, R. G. (2021). *Lenguajes de programación*. Editorial UNED.
- Mayeta, M. C., Batista, M., Rodríguez, R., & Fonden, J. C. (2020). La enseñanza de la programación estructurada y orientada a objeto en la modalidad semipresencial. *Referencia Pedagógica*, 8(1), 60-70. https://doi.org/http://scielo.sld.cu/scielo.php?script=sci_arttext&pid=S2308-30422020000100060&lng=es&tlng=es



- Medina, S. M. (2021). El aprendizaje cooperativo y sus implicancias en el proceso educativo del siglo XXI. *INNOVA Research Journal*, 6(2), 62–76. <https://doi.org/https://doi.org/10.33890/innova.v6.n2.2021.1663>
- Mera, G. J., & Timana, W. (2024). *Estrategia didáctica mediada por el software GeoGebra con enfoque de aprendizaje basado en problemas para el fortalecimiento de la comprensión del concepto de Ecuación Lineal Entera de primer grado en estudiantes de séptimo grado*. Universidad de Cartagena.
- Morante, H., & Ramírez, M. A. (2024). Comprensión Lectora y Aprendizaje Significativo en los Estudiantes de Educación Básica Superior. *Estudios y Perspectivas Revista Científica y Académica*, 4(2), 168–193. <https://doi.org/https://doi.org/10.61384/r.c.a.v4i2.207>
- Moreno, J. A., Mena, A. A., & Zerpa, L. I. (2024). Modelos de aprendizaje en la transición hacia la complejidad como un desafío a la simplicidad. *Sophia, Colección de Filosofía de la Educación*(36), 69-112. <https://doi.org/https://doi.org/10.17163/soph.n36.2024.02>
- Mullo, E. D. (2023). *Los simuladores virtuales en el proceso de enseñanza aprendizaje (Master's thesis)*. Universidad Técnica de Ambato.
- Neira, J. B. (2024). *Eficacia de un dispositivo grupal para el entrenamiento de habilidades sociales (Master's thesis)*. Universidad del Azuay.
- Nivela, M. A., Echeverría, S. V., & Santos, M. M. (2021). Educación superior con nuevas tecnologías de información y comunicación en tiempo de pandemia. *Horizontes Revista de Investigación en Ciencias de la Educación*, 5(19), 813-825. <https://doi.org/https://doi.org/10.33996/revistahorizontes.v5i19.239>
- Ontoria, A., Gómez, J. P., & de Luque, Á. (2023). *Aprendizaje centrado en el alumno: Metodología para una escuela abierta (Vol. 176)*. Narcea Ediciones.
- Oседа, D., Mendivel, R. K., & Angoma, M. (2020). Estrategias didácticas para el desarrollo de competencias y pensamiento complejo en estudiantes universitarios. *Sophia, colección de Filosofía de la Educación*(29), 235-259. <https://doi.org/https://doi.org/10.17163/soph.n29.2020.08>
- Padín, R. R. (2023). *Aprendizaje cooperativo a través de las TIC*. Aula Magna Proyecto clave McGraw Hill.



- Papy, F. (2020). *Bibliotecas digitales: interoperabilidad y usos*. ISTE Group.
- Parra, M. J. (2022). Aplicación de las TIC, b-Learning y Pensamiento Computacional para el Fortalecimiento de las Competencias Matemáticas. *Revista Tecnológica-Educativa Docentes 2.0*, 14(2), 29-41. <https://doi.org/https://doi.org/10.37843/rted.v14i2.312>
- Pascagaza, E. F., & Estrada, L. C. (2020). Modernización de la educación virtual y su incidencia en el contexto de las Tecnologías de la Información y la Comunicación (TIC). *Revista Academia y virtualidad*, 13(2), 103-116. <https://doi.org/https://dialnet.unirioja.es/servlet/articulo?codigo=7643870>
- Pérez, O. P. (2024). *Programación didáctica para el módulo de Base de Datos. Situación de aprendizaje: Realización de consultas simples (Tesis de Maestría)*. Universidad de La Laguna.
- Ponce, E. A., & Baltazar, S. (2024). Preliminares para la elaboración del estado de la cuestión sobre la educación en el más simple de sus fines y la mentalidad para alcanzar el éxito. *Revista Metropolitana de Ciencias Aplicadas*, 7(Suplemento 1), 93-101. <https://doi.org/https://doi.org/10.62452/zma24t66>
- Prado, A. (2021). Conectivismo y diseño instruccional: ecología de aprendizaje para la universidad del siglo XXI en México. *Márgenes Revista de Educación de la Universidad de Málaga*, 2(1), 4-20. <https://doi.org/https://doi.org/10.24310/mgnmar.v2i1.9349>
- Preciado, C. R. (2024). La inteligencia emocional en el rol del docente tutor universitario. *Simbiosis*, 4(8), 105–127. <https://doi.org/https://doi.org/10.59993/simbiosis.V.4i8.54>
- Proaño, P. M., & Trujillo, S. F. (2021). Aprendizaje de espacios vectoriales mediante la operación interna definida en C++. *Revista InGenio: La revista de la Facultad de Ciencias de la Ingeniería de la UTEQ*, 4(2), 28-36. <https://doi.org/https://dialnet.unirioja.es/servlet/articulo?codigo=8375688>
- Pucuji, K. M. (2022). *Estrategia tecnológica para el aprendizaje adaptativo en la enseñanza de programación estructurada (Master's thesis)*. Universidad Técnica de Cotopaxi (UTC).
- Puig, N. S. (2020). *Evaluar y aprender: un único proceso*. Ediciones Octaedro.



- Quevedo, V. (2022). *Guía para la definición y adaptación de algoritmos para su implementación en FPGAs mediante High Level Synthesis (Doctoral dissertation)*. Universitat Politècnica de València.
- Quispe, E. (2021). El aprendizaje basado en problemas y su influencia en el desarrollo del pensamiento crítico en la educación peruana: Array. *Maestro Y Sociedad*, 18(2), 541-550. <https://doi.org/https://maestroysociedad.uo.edu.cu/index.php/MyS/article/view/5357>
- Quispe, P. (2024). *Trabajo colegiado para la mejora de la práctica pedagógica de una institución educativa pública del Callao*. Universidad San Ignacio de Loyola.
- República del Ecuador. (2011, 31 de marzo). Ley Orgánica de Educación Intercultural. *Segundo Suplemento del Registro Oficial No. 417*.
- Rodríguez, V., Canchola, S. L., & Garzón, R. (2024). Pautas para la construcción de un Repositorio de Software Educativo en la Universidad Autónoma de Chiapas. *EDMETIC*, 13(1). <https://doi.org/https://doi.org/10.21071/edmetic.v13i1.15911>
- Román, R. E. (2022). *Sistema de monitoreo y control de irrigación usando Internet de las Cosas (IoT) (Bachelor's thesis)*. Universidad Técnica del Norte.
- Ruiz, G. F. (2024). Metodología de investigación científica para el estudio de variables de aprendizaje en estudiantes. *Revista Multidisciplinaria Voces de América y el Caribe*, 1(1), 380-406. <https://doi.org/https://doi.org/10.5281/zenodo.11254574>
- Sosa, H. T. (2022). Aprendizaje cognoscitivo impulsor de la autorregulación en la construcción del conocimiento. *Revista de Ciencias Sociales (Ve)*(5), 172-183. <https://doi.org/https://www.redalyc.org/journal/280/28071845014/28071845014.pdf>
- Suárez, M. O., Hernández, C. A., Peñafiel, E. J., & Villena, C. A. (2024). Utilización de juegos de razonamiento lógico para potenciar competencias matemáticas en estudiantes de bachillerato. *MQRInvestigar*, 8(2), 2931–2950. <https://doi.org/https://doi.org/10.56048/MQR20225.8.2.2024.2931-2950>
- Sulaiman, M. S., Jamaludin, M. H., & Derasit, Z. (2023). Code cody: una plataforma de aprendizaje basada en juegos para la enseñanza de la programación. *Journal of ICT in Education*, 10(1), 79–91. <https://doi.org/https://doi.org/10.37134/jictie.vol10.1.7.2023>



- Tascón, M., & Montolío, E. (2020). *El derecho a entender: la comunicación clara, la mejor defensa de la ciudadanía*. Los libros de la Catarata.
- Tuárez, L. A., & Guzmán, A. C. (2023). Importancia de la estrategia educativa para fortalecer el desarrollo de habilidades de trabajo autónomo en los estudiantes. *Revista Cognosis*, 8(EE1), 191–206. <https://doi.org/https://doi.org/10.33936/cognosis.v6i0.3335>
- Tumbaco, A. R., Miranda, E. J., Veintimilla, H. C., & Reyes, J. M. (2023). Modelo interactivo Educalm para fortalecer las competencias en la asignatura de fundamentos de programación de los estudiantes del primer semestre de la universidad UPSE. *Ciencia Latina Revista Científica Multidisciplinar*, 7(2), 11078-11103. https://doi.org/https://doi.org/10.37811/cl_rcm.v7i2.6189
- Valeriano, G. A. (2021). Experiencias de aprendizaje basado en proyectos para resolver diversos desafíos de la vida. *Revista Latinoamericana Ogmios*, 1(2), 170-177. <https://doi.org/https://doi.org/10.53595/rlo.v1.i2.016>
- Vallejo, V. A. (2023). *Estrategia didáctica, basada en el aprendizaje colaborativo para potenciar el rendimiento académico de los estudiantes de décimo año de EGB en la asignatura de Estudios Sociales (Bachelor's thesis)*. Riobamba.
- Van Dijck, J. (2019). *La cultura de la conectividad: una historia crítica de las redes sociales*. Siglo XXI editores.
- Velasco, M. F., & Graus, M. E. (2024). Influencia esencial de la neuroplasticidad para optimizar el potencial de aprendizaje en la Educación Primaria. *Dilemas contemporáneos: Educación, Política y Valores*. <https://doi.org/https://doi.org/10.46377/dilemas.v11i3.4139>
- Vélez, G. (2023). *Implementación de interfaz en Common Lisp para la utilización de librerías C/C++ nativas GNU-GSL y OpenCV (Trabajo de grado)*. Universidad Politécnica de Madrid.
- Zainol, M. K., & Ismail, A. (2021). Desarrollo de un juego de acción y aventuras en 3D llamado “Code - E” para aprender códigos de función de C++. *Journal of ICT in Education*, 8(3), 13–26. <https://doi.org/https://doi.org/10.37134/jictie.vol8.sp.1.2.2021>



- Zoila, N. V. (2023). Los recursos didácticos como apoyo en el proceso de enseñanza-aprendizaje de los estudiantes. *MQRInvestigar*, 7(3), 4078-4105.
<https://doi.org/https://doi.org/10.56048/MQR20225.7.3.2023.4078-4105>
- Zurita, M. S. (2020). El aprendizaje cooperativo y el desarrollo de las habilidades cognitivas. *Revista EDUCARE - UPEL-IPB - Segunda Nueva Etapa 2.0*, 24(1), 51–74.
<https://doi.org/https://doi.org/10.46498/reduipb.v24i1.1226>



Validación de entrevista semiestructurada aplicada a docentes. Experto 2

ENTREVISTA DIRIGIDA A DOCENTES DE LA UNIDAD EDUCATIVA AMAZONAS

OBJETIVO: Validar la propuesta diseñada para el uso de la herramienta "CODING C++" en el mejoramiento del proceso de enseñanza-aprendizaje de la asignatura de programación estructurada en los estudiantes de segundo de bachillerato.

INDICACIONES: En una escala de 1 = nunca 2 = a veces 3 = casi siempre 4= siempre, seleccione la que usted considere pertinente

	SUFICIENCIA	CLARIDAD	COHERENCIA	RELEVANCIA	COMENTARIO
1. ¿La propuesta presentada cumple con las Funcionalidades tales como usar programación orientada a objetos; hacer un buen manejo de memoria; y utilizar plantillas y bibliotecas estándar?	4	4	4	4	
2. ¿Cómo observa el manejo de aplicaciones en la propuesta presentada; con respecto a ofrecer desarrollo de juegos y aplicaciones gráficas; manejo de grandes cantidades de datos de manera eficiente; y la evidencia de una respuesta rápida y uso eficiente de los recursos del sistema?	4	4	4	4	
3. Con referencia a los entornos de desarrollo de la propuesta, ¿Cuál es su opinión sobre la forma como se utilizan los IDEs (Entornos de Desarrollo Integrados) más populares para facilitar el proceso de desarrollo?	4	4	4	4	
4. ¿Cómo la propuesta diseñada, utiliza los recursos de aprendizaje disponibles para C++; y ofrece comunidades y grupos en línea?	4	4	4	4	
5. ¿Considera válida la propuesta presentada?	4	4	4	4	

C++

6
Guía para el uso de la Herramienta “CODING C++” en el proceso de enseñanza-aprendizaje de Programación Estructurada

Contenido a estudiar

1. Introducción a la Programación Estructurada

- Definición y principios de la programación estructurada.
- Importancia de la programación estructurada en el desarrollo de software.
- Comparación con otros paradigmas de programación.

2. Descripción de la Herramienta “CODING C++”

- Tipos de datos y variables.
- Operadores y expresiones.
- Estructuras de control (condicionales y bucles).
- Características y funcionalidades de la herramienta.
- Descripción de la Herramienta: Explicar qué es “CODING C++” y sus características principales, como la interfaz amigable y las funcionalidades interactivas.
- Acceso y Configuración: Instrucciones sobre cómo acceder a la herramienta y configurarla para su uso en el aula.
- Requisitos técnicos para su uso (hardware y software).
- Guía de instalación y configuración.
- Objetivos que se desean alcanzar con el uso de la herramienta “CODING C++”.
- Cómo la herramienta contribuye a mejorar el proceso de enseñanza-aprendizaje.