



REPÚBLICA DEL ECUADOR

UNIVERSIDAD ESTATAL DE MILAGRO

INSTITUTO DE INVESTIGACIÓN Y POSGRADO

**INFORME DE INVESTIGACIÓN PREVIO A LA OBTENCIÓN DEL
GRADO DE:**

**MAGÍSTER EN EDUCACIÓN
MENCIÓN TECNOLOGÍA E INNOVACIÓN EDUCATIVA**

TEMA:

El uso de herramientas TIC's en el aprendizaje de Lógica de Programación de los
estudiantes de Bachillerato de la Unidad Educativa Urdaneta, durante el Periodo
Lectivo 2020 - 2021

Autor:

Ing. Walter Chacón Valverde

Director:

Msc. Denis Darío Mendoza Cabrera

Milagro, 2021

Derechos de autor

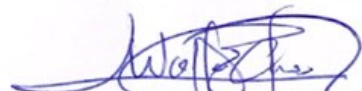
Sr. Dr.
Fabricio Guevara Viejó
Rector de la Universidad Estatal de Milagro
Presente.

Yo, **WALTER EGIDIO CHACON VALVERDE** en calidad de autor y titular de los derechos morales y patrimoniales de este informe de investigación, mediante el presente documento, libre y voluntariamente cedo los derechos de Autor de esta investigación, que fue realizada como requisito previo para la obtención de mi Grado, de Magister en Educación, como aporte a la Línea de Investigación **Tecnología e Innovación Educativa** de conformidad con el Art. 114 del Código Orgánico de la Economía Social de los Conocimientos, Creatividad e Innovación, concedo a favor de la Universidad Estatal de Milagro una licencia gratuita, intransferible y no exclusiva para el uso no comercial de la obra, con fines estrictamente académicos. Conservo a mi favor todos los derechos de autor sobre la obra, establecidos en la normativa citada.

Así mismo, autorizo a la Universidad Estatal de Milagro para que realice la digitalización y publicación de este Proyecto de Investigación en el repositorio virtual, de conformidad a lo dispuesto en el Art. 144 de la Ley Orgánica de Educación Superior.

El autor declara que la obra objeto de la presente autorización es original en su forma de expresión y no infringe el derecho de autor de terceros, asumiendo la responsabilidad por cualquier reclamación que pudiera presentarse por esta causa y liberando a la Universidad de toda responsabilidad.

Milagro, 12 de julio del 2021

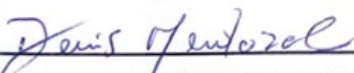


WALTER EGIDIO CHACON VALVERDE
1204319393

Aprobación del director del Trabajo de Titulación

Yo, **DENIS DARIO MENDOZA CABRERA** en mi calidad de director del trabajo de titulación, elaborado por **WALTER EGIDIO CHACON VALVERDE**, cuyo tema es **El uso de herramientas TIC's en el aprendizaje de Lógica de Programación de los estudiantes de Bachillerato de la Unidad Educativa Urdaneta, durante el Periodo Lectivo 2020 - 2021**, que aporta a la Línea de Investigación **Tecnología e Innovación Educativa**, previo a la obtención del Grado Magister en Educación, mención Tecnología e Innovación Educativa. Trabajo de titulación que consiste en una propuesta innovadora que contiene, como mínimo, una investigación exploratoria y diagnóstica, base conceptual, conclusiones y fuentes de consulta, considero que el mismo reúne los requisitos y méritos necesarios para ser sometido a la evaluación por parte del tribunal calificador que se designe, por lo que lo **APRUEBO**, a fin de que el trabajo sea habilitado para continuar con el proceso de titulación de la alternativa de Informe de Investigación de la Universidad Estatal de Milagro.

Milagro, 18 de mayo del 2021



MSG. DENIS DARIO MENDOZA CABRERA
C.I.: 0923489801

Dedicatoria

Quiero dedicar este trabajo a mis queridos padres Miguel Chacón y Victoria Valverde, a mi amada esposa María Puga y a mis preciosas hijas quienes han sido un pilar fundamental para la culminación de este proceso y constituyen lo mejor y más valioso que Dios me ha dado.

Agradecimiento

Agradezco en primer a Dios, por haberme dado los dones necesarios para alcanzar esta meta. Mi imperecedero agradecimiento al Master Denis Mendoza, por su dedicación y acertado asesoramiento ha hecho posible la realización oportuna de este proyecto. Quiero agradecer a la Universidad Estatal de Milagro en la persona del Dr. Jorge Córdova por la oportunidad que me ha brindado para cursar mis estudios de cuarto nivel. Un agradecimiento a mis padres y a mi familia por todo el apoyo brindado durante este proceso y un agradecimiento a todas las personas que de alguna manera han contribuido a la feliz culminación de este objetivo.

Índice de contenidos

Derechos de autor.....	i
Aprobación del director del Trabajo de Titulación.....	ii
Aprobación del tribunal calificador	iii
Agradecimiento.....	v
Índice de contenidos.....	vi
Glosario de términos.....	xi
Resumen	xiii
Abstract.....	xiv
Introducción.....	1
Capítulo I: El problema de la investigación.....	3
1.1 Planteamiento del problema	3
1.2 Delimitación del problema.....	4
1.3 Formulación del problema.....	4
1.4 Preguntas de investigación.....	4
1.5 Determinación del tema.....	5
1.6 Objetivo general.....	5
1.7 Objetivos específicos	5
1.8 Hipótesis	5
1.9 Declaración de las variables.....	5
1.10 Justificación	5
1.11 Alcance y limitaciones.....	7
CAPÍTULO II: Marco teórico referencial	8
2.1 Antecedentes	8
2.2 Fundamentación teórica.....	9
CAPÍTULO III: Metodología.....	39
3.1 Tipo y diseño de investigación	39
3.2 La población y la muestra	43
3.2.1 Características de la población.....	43

3.2.2	Delimitación de la población	44
3.2.3	Tipo de muestra	44
3.2.4	Tamaño de la muestra	45
3.2.5	Proceso de selección de la muestra.....	46
3.3	Los métodos y las técnicas.....	47
3.3.1	Métodos.....	47
3.3.2	Técnicas	48
3.4	Propuesta de procesamiento estadístico de la información.....	49
CAPÍTULO IV: Análisis e interpretación de resultados.....		50
4.1	Análisis de Descriptivo de los resultados	50
4.2	Análisis correlacional de los resultados	67
CAPÍTULO V: Conclusiones y Recomendaciones		69
5.1	Conclusiones.....	69
5.2	Recomendaciones.....	70
Bibliografía.....		71
ANEXOS.....		76

Lista de Tablas

Tabla 1.....	44
Tabla 2.....	46
Tabla 3.....	51
Tabla 4.....	52
Tabla 5.....	53
Tabla 6.....	54
Tabla 7.....	55
Tabla 8.....	56
Tabla 9.....	57
Tabla 10.....	58
Tabla 11.....	59
Tabla 12.....	60
Tabla 13.....	61
Tabla 14.....	62
Tabla 15.....	63
Tabla 16.....	64
Tabla 17.....	65
Tabla 18.....	66

Lista de figuras

Figura 1	15
Figura 2	17
Figura 3.	18
Figura 4.	21
Figura 5.	24
Figura 6	25
Figura 7	27
Figura 8	28
Figura 9	29
Figura 10	35
Gráfico estadístico 1	51
Gráfico estadístico 2	52
Gráfico estadístico 3	53
Gráfico estadístico 4	54
Gráfico estadístico 5	55
Gráfico estadístico 6	56
Gráfico estadístico 7	57
Gráfico estadístico 8	58
Gráfico estadístico 9	59
Gráfico estadístico 10	60
Gráfico estadístico 11	61
Gráfico estadístico 12	62
Gráfico estadístico 13	63
Gráfico estadístico 14	64
Gráfico estadístico 15	65
Gráfico estadístico 16	66

Lista de anexos

ANEXO 1	76
ANEXO 2	77
ANEXO 3	78
ANEXO 4	82
ANEXO 5	83
ANEXO 6	87
ANEXO 7	88
ANEXO 8	89

Glosario de términos

Alfa de Cronbach.- Es un coeficiente que sirve para medir la fiabilidad de una escala de medida, y cuya denominación Alfa fue realizada por Cronbach en 1951.

Aplicativo.- Sinónimo de aplicación, cuando se trata de un programa de computador.

Case Sensitive.- Expresión usada en informática que se aplica a los textos en los que tiene alguna relevancia escribir un carácter en mayúsculas o minúsculas.

Código.- En programación ha referencia al lenguaje por el cual funcionan las computadoras, comprende un conjunto de instrucciones y datos a ser procesados automáticamente.

Compilador.- Es un tipo de traductor que transforma un programa entero de un lenguaje de programación a otro. Usualmente el lenguaje objetivo es código máquina, aunque también puede ser traducido a un código intermedio o a texto.

Conectividad.- Es la capacidad de un dispositivo de conectarse y comunicarse con otro, con el fin de intercambiar información o establecer una conexión directa a base de información digital.

Estándar.- El concepto se utiliza para nombrar a aquello que puede tomarse como referencia, patrón o modelo.

Google Forms.- Es un software de administración de encuestas que se incluye como parte del paquete gratuito de editores de documentos de Google basado en la web que ofrece Google.

Hardware.- El hardware hace referencia a todos los componentes materiales y físicos de un dispositivo, es decir, aquellos que se pueden ver y tocar. El monitor, el ratón, la CPU, el teclado o la memoria RAM son algunos ejemplos de aquellas partes que, en su conjunto, forman el hardware.

IDE.- Conocido como Entorno de Desarrollo Integrado, es una aplicación informática que proporciona servicios integrales para facilitarle al desarrollador o programador el desarrollo de software.

Informática.- Es la rama de la ciencia que se encarga de estudiar la administración de métodos, técnicas y procesos con el fin de almacenar, procesar y transmitir información y datos en formato digital.

Interfaz.- Se utiliza en informática para nombrar a la conexión funcional entre dos sistemas, programas, dispositivos o componentes de cualquier tipo, que proporciona una comunicación de distintos niveles, permitiendo el intercambio de información.

Metalinguaje.- Un metalinguaje es un lenguaje que se usa para hablar acerca de otro lenguaje. Al lenguaje acerca del cual se está hablando se le llama el lenguaje objeto.

Programador.- Es aquella persona que elabora programas de computadora, es decir escribe, depura y mantiene el código fuente de un programa informático, que ejecuta el hardware de una computadora, para realizar una tarea determinada.

Sintaxis.- Es el conjunto de reglas que define las combinaciones de símbolos que se consideran enunciados o expresiones correctamente estructurados en un lenguaje de programación.

Software.- El software son las instrucciones para comunicarse con el ordenador y que hacen posible su uso. Resumiendo, el software en informática son los programas. Sin el software, las computadoras u ordenadores serían inútiles.

Web.- Se designa como 'la web' al sistema de gestión de información más popular para la transmisión de datos a través de internet.

Windows.- Es un sistema operativo creado por Microsoft. Consiste en un conjunto de programas que permiten la ejecución de los recursos que tiene un ordenador.

Resumen

El uso de la tecnología en la educación ha cambiado paulatinamente las metodologías de enseñanza y aprendizaje, lo cual se ve más acentuado aun cuando el ámbito de estudio está directamente relacionado con la tecnología, como es el caso de la Informática, en donde la asignatura de lógica de programación constituye una de las más críticas e importantes, pero sin embargo es la que mayores dificultades presenta a los estudiantes en el aprendizaje. El objetivo que se busca al desarrollar el presente trabajo es determinar la incidencia de las herramientas TIC's en el aprendizaje de lógica de programación de los alumnos del Bachillerato en la especialidad de Informática de la Unidad Educativa Urdaneta, para ello se ha realizado una breve revisión de las diferentes aplicaciones y lenguajes de programación empleados en la enseñanza de programación a lo largo de los tres niveles de bachillerato. Seguidamente se aplican métodos y técnicas de investigación, que consisten en encuestas dirigidas a los estudiantes y entrevista a la autoridad de la Institución antes mencionada con el objeto de recabar la información pertinente, la misma que luego fue procesada utilizando el software estadístico SPSS de IBM y a partir de los resultados obtenidos se procede a realizar el análisis de la información. De acuerdo con los resultados obtenidos se puede concluir que el uso adecuado de las herramientas TIC's juega un papel preponderante en la enseñanza de lógica de programación, pero debe estar acompañada de un efectivo uso de metodologías de enseñanza constructivistas que favorezcan el autoaprendizaje y el aprendizaje colaborativo mediante la aplicación de metodologías, como el aprendizaje basado en proyectos y el aprendizaje basado en problemas.

Palabras clave: aprendizaje de programación, herramientas TIC's, algoritmos

Abstract

The use of communication in education often changes the methodologies and learning, which is more marked even when the field of study is directly related to technology, like informatics, where the logic programming course is critical. However, it is the most difficult presenting to students in the learning processes. The study evaluates ICT tools in the programming logic among the college students of Urdaneta Educational Center. A brief review of the different applications and programming languages used in programming holds in three levels. Next, there are applied investigation methods that consist of surveys focused on the students and an interview with the authorities. The information is later processed using SPSS statistical software from IBM. The results show that the proper use of ICT tools is significant in the teaching of programming. Specifically, this happens through the engagement of self-learning and collaborative learning.

Keyword: programming learning, ICT tools, algorithms.

Introducción

Conforme se ha dado el uso masivo de la tecnología en diferentes ámbitos, como es el caso en la medicina, en las empresas, en la ciencia, etc., también ha surgido la necesidad de contar con personal que tenga las competencias necesarias en el manejo de los programas y aplicaciones, siendo en muchos casos necesario contar con la competencia de crear programas, para resolver diferentes tipos de problemas. Ante esto las instituciones de educación superior y nivel medio de todo el mundo ha incorporado paulatinamente la enseñanza de programación como una de las asignaturas básicas en sus programas educativos sobre el uso y manejo de la tecnología.

El Ministerio de Educación del Ecuador consciente de esta necesidad, ha fomentado en las instituciones educativas la enseñanza de computación, lo cual ha sido acentuado ya en una forma más estructurada mediante la incorporación del Bachillerato Técnico, con el fin de ofertar una formación dirigida a los jóvenes y enfocada en ofrecerles una formación que favorezca su incorporación al mundo laboral o a su vez le permita continuar a una formación técnica y tecnológica de educación superior.(Ministerio de Educación del Ecuador, 2016)

La Unidad Educativa Urdaneta, al igual que algunas instituciones que ofertan el bachillerato en la localidad, ha incorporado la enseñanza de Informática a sus especialidades de formación técnica, y con ello la enseñanza de la programación como una parte importante de su programa formativo.

La enseñanza de programación es todo un desafío para los docentes, ya que esto implica conocer profundamente los fundamentos técnicos para crear y desarrollar programas, caso contrario no podrá realizar un proceso efectivo en la transmisión de su conocimiento hacia el estudiante, por otro lado, también a la mayoría de los estudiantes les resultan bastante complejo poder desarrollar las destrezas necesarias para aprender a desarrollar programas. Por lo que es esencial el uso adecuado de los recursos tecnológicos y las metodologías de enseñanza que le permitan a estudiante lograr un aprendizaje significativo.

En este sentido se ha planteado como pregunta central de este trabajo ¿Cómo incide el uso de las herramientas TIC's en el aprendizaje de Lógica de programación?, ya que dada la naturaleza misma de la asignatura se ha establecido como hipótesis que el uso de las herramientas TIC's incide en el aprendizaje de lógica de programación en los estudiantes. El objetivo central es analizar las herramientas TIC's aplicadas por los docentes en la enseñanza de programación, la forma como afecta a los estudiantes la disponibilidad de los recursos tecnológicos y la medida en la que se emplean las metodologías de aprendizaje basado en proyectos y problemas en la asignatura de programación.

Para llevar a efecto esta investigación, el presente trabajo se ha estructurado en cinco capítulos. En el capítulo I “El problema de la investigación”, se realiza una descripción del problema que se ha planteado investigar, se incluyen las variables que se involucran en este caso, así como también la hipótesis planteada, los objetivos que se buscan alcanzar al finalizar el estudio, se describe también la justificación y el alcance del estudio. El capítulo II “Marco teórico referencial”, detalla los antecedentes y la fundamentación teórica sobre las diferentes herramientas TIC's y lenguajes empleados para la enseñanza de la programación. El capítulo III “Metodología”, hace referencia al tipo y diseño de investigación utilizada para la realización de este trabajo y también a los métodos y técnicas empleadas para llevar a cabo el proceso investigativo y de recolección de datos. En el capítulo IV “Análisis e interpretación de resultados”, se realiza el análisis y la interpretación de los resultados obtenidos en el trabajo investigativo a través de la encuesta aplicadas a los estudiantes. Se detalla también el análisis correlacional de los datos para determinar si se cumple o no la hipótesis planteada en los capítulos anteriores. Finalmente, en el Capítulo V se exponen las conclusiones y recomendaciones surgidas luego de la culminación de este trabajo.

Capítulo I: El problema de la investigación

1.1 Planteamiento del problema

En la sociedad actual, donde la tecnología prácticamente está inmersa en todos los aspectos; en el hogar, en el trabajo, y por supuesto también en el que hacer educativo; todos los dispositivos tecnológicos funcionan en base a programas desarrollados para realizar tareas específicas, desde los más sencillos como son juegos interactivos para niños, hasta los más complejos, como realizar operaciones asistidas por computadora. Para lograr todo esto se desarrollan programas basados en algoritmos que deben seguir una secuencia lógica y ordenada para llevar a un correcto funcionamiento y en muchos casos la consecución correcta de una tarea o resolución. Esto quiere decir que los algoritmos y programas están presentes en todos los dispositivos tecnológicos que usamos, y el usuario común lo puede notar en forma totalmente transparente, conociéndolos como aplicación y/o aplicativos.

El Ministerio de Educación, a través de sus órganos desconcentrados, ha promulgado desde hace varios años el Bachillerato Técnico, fomentando diversas especialidades bajo el nombre de figuras profesionales. Dentro de estas especialidades podemos encontrar a varias que se relacionan, directamente con la tecnología, conocidas como Informática, Administración de Sistemas o Aplicaciones Informáticas, cada una de ellas con una malla curricular que, si bien no incluyen las mismas asignaturas, aprender a desarrollar programas, es una parte medular de las figuras profesionales, por lo que dedican una significativa cantidad de horas a esta asignatura en los diferentes grados del Bachillerato Técnico.

La asignatura de Programación y Bases de datos es una asignatura prioritaria dentro de la formación de Bachillerato Técnico en la figura profesional de Informática, por lo que es una asignatura impartida desde el primer año de Bachillerato, iniciando con lo más básico como es la elaboración de algoritmos, encaminado a resolver problemas. Sin embargo, se ha visto que para el alumno es muy complejo aprender a desarrollar algoritmos que permitan resolver un problema adecuadamente, a pesar de las explicaciones y

ejemplos, no logra por sí solo captar e interiorizar la destreza para elaborar algoritmos.

La elaboración de los algoritmos requiere desarrollar el pensamiento lógico en el alumno, caso contrario no podrá encontrar los métodos que le permitan desarrollar algoritmos por sí solo, por lo tanto, se debe lograr que el alumno, sea para empezar analítico, luego reflexivo, y finalmente creativo e intuitivo para que adquiera la destreza lógica para resolver problemas.

Es imprescindible que los alumnos desarrollen la lógica para elaborar algoritmos, de no ser así, no podrá lograr la destreza para desarrollar programas en las computadoras que se requieren en los años superiores de la especialidad.

1.2 Delimitación del problema

Área de investigación: Educación y tecnología educativa.

Línea de investigación: Calidad de los sistemas educativos en los diferentes niveles de enseñanza.

Cobertura del proyecto: Estudiantes Bachillerato, especialidad Informática de la Unidad Educativa “Urdaneta”.

Campo de interés: Directivos, docentes y educandos.

Entidad responsable: Unidad Educativa “Urdaneta”.

1.3 Formulación del problema

¿Cómo incide el uso de las Herramientas TIC's en el aprendizaje de Lógica de programación en los alumnos de bachillerato de la Unidad Educativa Urdaneta, del periodo lectivo 2020-2021?

1.4 Preguntas de investigación

¿Cuáles son las herramientas tecnológicas utilizadas para la enseñanza de algoritmos y programación?

¿Qué importancia tiene el uso de las herramientas Tics en el aprendizaje de lógica de programación en los estudiantes de Bachillerato?

¿Cómo afecta la disponibilidad de internet y recursos tecnológicos en el aprendizaje de lógica de programación ?

¿Cuáles son las estrategias que utilizan los docentes para la enseñanza de Lógica de Programación a los estudiantes de Bachillerato?

1.5 Determinación del tema

El uso de herramientas TIC's en el aprendizaje de Lógica de Programación de los estudiantes de Bachillerato de la Unidad Educativa Urdaneta, durante el Periodo Lectivo 2020 - 2021

1.6 Objetivo general

Determinar la incidencia del uso de las herramientas TIC's en el aprendizaje de lógica de programación en los estudiantes de bachillerato, para mejorar su desempeño académico.

1.7 Objetivos específicos

- Analizar las herramientas TIC aplicadas por los docentes de Programación Estructurada y bases de datos.
- Evaluar la importancia del uso de herramientas TIC's en el aprendizaje de lógica de Programación en los estudiantes de Bachillerato.
- Determinar cómo afecta a los estudiantes la disponibilidad de internet y otros recursos tecnológicos en su aprendizaje de lógica de programación.
- Identificar las principales estrategias que emplean los docentes para la enseñanza de lógica de programación en los estudiantes de Bachillerato.

1.8 Hipótesis

El uso de las Herramientas TIC's incide en el aprendizaje de Lógica de programación en los alumnos de bachillerato.

1.9 Declaración de las variables

VARIABLE INDEPENDIENTE: Herramientas TIC´s"

VARIABLE DEPENDIENTE: Aprendizaje de Lógica de Programación"

1.10 Justificación

La demanda de profesionales técnicos debidamente preparados con conocimientos informáticos fue una de las razones por las que, en la Unidad educativa Urdaneta se implementa el Bachillerato Técnico con la especialidad de Aplicaciones Informáticas. Dentro de ella, Programación en lenguajes estructurados es una las principales asignaturas que el estudiante debe revisar durante los tres años de Bachillerato, sin embargo, al estudiante siempre le ha

resultado especialmente difícil de asimilar y aprender los conceptos que se revisan en esta asignatura.

La enseñanza de programación cobra especial importancia en la actualidad dado el nivel de globalización tecnológica que se ha alcanzado en la actualidad con el uso masivo de la tecnología. Las empresas y más instituciones demandan continuamente profesionales que tengan un amplio conocimiento de diferentes lenguajes de programación de tal forma que puedan resolver problemas de diferente índole mediante el uso y la programación de computadoras, robots y otras máquinas automatizadas.

De entre las variadas herramientas que existen para la programación, es visto que cada una posee un amplio conjunto de características propias, para el desarrollo de programas, por lo que se debe entonces determinar cuáles son las de mayor importancia y aplicabilidad en el proceso de enseñanza. Entre estos se tiene el lenguaje de programación soportado, debido que existen una gran cantidad de lenguajes de programación que prácticamente esta encaminados a un mismo fin y prácticamente todos lo consiguen de un modo y otro; la modalidad en la cual se trabaja también es importante, ya que muchos empiezan con la generación de código en modo textual, ya en niveles más avanzados se implementa con ambientes visuales que resultan mucho más atractivos y dinámicos.

En el presente trabajo investigativo se trata de determinar la incidencia que tienen el uso de herramientas TIC's para la enseñanza y aprendizaje de lógica de programación en los estudiantes, constatando cuales son las herramientas tecnológicas que usan en cada uno de los años de bachillerato y si tienen alguna preferencia por las herramientas de aprendizaje basado en código de bloques como la aplicación Scratch o similar.

Mediante el uso de las herramientas TIC's en el proceso de enseñanza y aprendizaje se pretende lograr que el estudiante pueda alcanzar un aprendizaje significativo, desarrollando un especial interés por esa asignatura que en muchos casos resulta ser la principal causa de desmotivación y el posterior efecto en el rendimiento académico. Seria de especial beneficio que los estudiantes cuenten con una herramienta determinada que les facilite la asimilación de los conceptos y procedimientos que les permitan por medio de la programación solucionar cualquier tipo de problema planteado o por lo

menos poseer los conocimientos técnicos para diseñar una solución adecuada siguiendo una metodología perfectamente definida.

1.11 Alcance y limitaciones

Este proyecto tiene carácter exploratorio-descriptivo, ya que se busca indagar y conocer la realidad del aprendizaje de la asignatura de lógica de programación en los estudiantes de bachillerato de la Unidad educativa Urdaneta durante el periodo lectivo 2020 – 2021. Se describe algunas de las herramientas TIC's y las metodologías empleadas en la enseñanza con el objetivo de determinar cómo inciden en su proceso de aprendizaje. Para ello se empleó una encuesta dirigida a los estudiantes, con los indicadores necesarios referente a la temática de estudio, lo que permitió recolectar la información necesaria y realizar los análisis para verificar la relación que existe entre las dos variables propuestas como son las herramientas TIC's y el aprendizaje de Lógica de programación, con el objeto de analizar y utilizar los datos obtenidos para la toma de decisiones en la institución en lo referente a la enseñanza de programación .

Una de las limitantes que se encontró al momento de realizar este procedimiento fue la falta de clases presenciales provocada por la pandemia de COVID-19 durante el periodo lectivo 2020 – 2021. Por lo que varios estudiantes se vieron afectados por la escasa posibilidad de tener conectividad, limitando en cierta medida el uso las herramientas TIC's y por ende la cantidad de estudiantes seleccionados para la muestra de estudio. Sin embargo, gracias a las ventajas que ofrecen las herramientas colaborativas, fue posible llevar a efecto los procedimientos necesarios para recabar y analizar la información obtenido.

Esta investigación abarca únicamente la temática de la incidencia de uso de las Herramientas TIC's en el aprendizaje de lógica de programación de los alumnos de bachillerato de la Unidad Educativa “Urdaneta”, por lo que no se abordan temas o temáticas que no estuvieran relacionados con dicha asignatura.

CAPÍTULO II: Marco teórico referencial

2.1 Antecedentes

Desarrollar programas es una de las tareas más complejas que se presentan en la especialidad de aplicaciones informáticas, se ha visto que, en casi todas las instituciones de enseñanza y preparación tanto a nivel medio como superior, resulta de singular complejidad la asignatura de programación, por lo que se presenta la necesidad de utilizar técnicas efectivas de programación mediante el uso de herramientas TIC's desde el principio. (Moroni & Señas, 2014)

Si bien es cierto que existe una amplia gama de herramientas tecnológicas que se pueden utilizar en el proceso de enseñanza y aprendizaje, es necesario también establecer cuáles serían idóneas y organizarlas adecuadamente para abordarlas en los diferentes niveles de aprendizaje, considerando características esenciales como, la facilidad de diseño, la generación automática de código y el manejo visual de los elementos de programación. (Guerrero et al., 2015)

(Barceló Adrover, 2020) propone introducir gradualmente a los estudiantes de nivel secundario, en este caso a los estudiantes de bachillerato en el aprendizaje basado en proyectos (ABP), esto debido a que puede resultar ser una metodología nueva para ellos, pero que se puede empezar con los conceptos considerados imprescindibles, empezando con la realización de proyectos pequeños, para luego aumentar poco a poco el grado de dificultad, de manera que los estudiantes desarrollen progresivamente aptitudes de trabajo organizado para poder abordar en forma colaborativa, proyectos de mayor complejidad.

Por lo señalado en el párrafo anterior, la metodología ABP, cobra singular importancia debido a que la enseñanza de la programación de computadoras utiliza entre otras estrategias la resolución de problemas a través del uso de computadoras. Este proceso se descompone en varias etapas que van desde la interpretación del enunciado del problema, el modelado de una solución, selección de las estructuras de datos más adecuadas a la situación planteada, escritura del algoritmo, implementación en un lenguaje de programación de alto nivel, compilación, ejecución y depuración del programa resultante.

Desde que las instituciones educativas encaminadas a innovar en su oferta educativa fueron implementando la especialidad de la informática, justificada por incursión de la tecnología en todos los ámbitos, la enseñanza de la asignatura de Programación estructurada ha sido siempre parte del proceso de formación, aun con el rediseño curricular del cual fue objeto hace dos años con la reestructuración del Bachillerato Técnico,(Jácome et al., 2017) estableciendo contenidos temáticos que están centradas en la formación del alumnado desde una perspectiva que les permita adquirir saberes y competencias que estén conectadas con su contexto actual y sean capaces de satisfacer alguna necesidad de la sociedad.(Moreno, 2016)

Sin embargo, hasta el presente no existe un consenso definido sobre el uso de una metodología que permita lograr un aprendizaje efectivo de la lógica de programación, ya que, siendo realistas son muy pocos los estudiantes que logran desarrollar las habilidades del pensamiento que les permita elaborar más fácilmente algoritmos para solucionar problemas e implementarlos como programas de computadora.

2.2 Fundamentación teórica

Las Herramientas de Tecnología de la Información y Comunicación (TIC'S) constituyen básicamente el conjunto de dispositivos y programas de computadora que permiten procesar información previamente ingresada, para luego presentar resultados, ya sea en forma visual o impresa. Estas tecnologías se han vuelto tan cotidianas, que prácticamente son utilizadas en todas partes, y para una gran cantidad de tareas.

Sin embargo, llegar al uso de estas tecnologías, ha requerido un desarrollo de habilidades y destrezas, que van desde las más básicas y elementales, como escribir una carta, hasta las más avanzadas como son el manejo y gestión de complejos sistemas de información. Para ello ha sido necesario incurrir en lo que se ha llamado como alfabetización digital. Lo cual es citado por (Falco et al., 2018) como *“la capacidad de utilizar las tecnologías de información y comunicación para encontrar, comprender, evaluar, crear y comunicar información digital, una habilidad que requiere tanto de habilidades cognitivas como técnicas”*.

En este sentido, las herramientas TIC's juegan un papel fundamental, ya que además de cambiar la práctica docente, ha hecho posible que nuevas especialidades y asignaturas sean implantadas en las Instituciones educativas de bachillerato; siendo Programación una de estas asignaturas, llevada de la mano de las especialidades relacionadas a la informática y la computación.

Programas de computadoras

También se los conoce como *software* o *aplicaciones*, los programas constituyen un conjunto de instrucciones escritas en otro programa que se conoce como Lenguaje de Programación. Estas instrucciones debidamente detalladas permiten controlar adecuadamente el Hardware de la computadora y la realización de las operaciones de un sistema informático, que se pueden resumir en tres aspectos básicos como son: recibir datos por medio de un dispositivo de entrada, procesar esta información y finalmente, en caso de ser necesario entregar resultados.

Tipos de programas

Antes de entrar al estudio de elaboración de los programas es necesario describir la clasificación generalmente aceptada de los tipos de software:

- **Software de sistemas.** – Es el conjunto de programas cuya finalidad es dar soporte al desarrollo y funcionamiento de otros programas. También se encargan de coordinar las diferentes partes del sistema informático, se conecta e interactúa entre el software de la aplicación y el hardware de la computadora. Entre los diferentes subtipos de software de sistema existen los que tienen como tarea administrar o controlar las actividades de la computadora y realizar tareas de proceso comunes. Estos programas se conocen también como programas utilitarios. (Joyanes Aguilar, 2008)

El software del sistema que administra y controla las actividades de la computadora se llama sistema operativo. El otro software del sistema es un traductor de lenguaje de computadora o un programa de traducción, que convierte un lenguaje de programación que es comprendido por el programador, en un lenguaje de máquina que la computadora puede entender.

- **Software de aplicación.** – Es una herramienta que permite a los usuarios realización de tareas específicas como contabilidad administrativa o redacción de texto.(Coila Villena, n.d.)

La función principal del software de aplicación es ayudar a los usuarios de computadoras a realizar tareas específicas. Se pueden utilizar diferentes herramientas de software para desarrollar aplicaciones. Por ejemplo, una aplicación de procesamiento de texto (como Word) puede ayudar a crear documentos; una hoja de cálculo (como Excel) puede ayudar a automatizar tareas de cálculo matemático tediosas y repetitivas, adicionalmente facilita el procesamiento de información estadística para generar gráficos o representaciones visuales (como PowerPoint) o creación de bases de datos (como Access o SQL Server) para ayudar en la gestión de archivos y registros de datos.

Este tipo de software es el más diversificado, esto debido a la amplia gama de aplicabilidad que tiene, siendo muchos de ellos de propósito general, pero en ciertos casos son diseñados y programados a medida y para propósitos específicamente determinados por el solicitante.

- **Software de programación.** - (Sánchez Lopez, 2016) Nos indica que el software de programación es un conjunto de herramientas que permite a los programadores utilizar diferentes métodos alternativos y lenguajes de programación para desarrollar programas informáticos de forma práctica. En la mayoría de los casos, estos programas se conocen también como lenguajes de programación. Actualmente ya se cuenta con una amplia variedad de programas que permiten crear y diseñar cualquier tipo de aplicaciones, cada uno con características y ventajas más avanzadas que otros.

Esta versatilidad es lo que ha permitido también clasificarlos en diferentes escalas de complejidad, lo que hace posible su utilización en la enseñanza de los métodos para programar aplicaciones, empezando desde las que pueden considerarse básicas, para aprender los primeros pasos en la programación, hasta llegar paulatinamente a los programas más estructurados, que ayudan a que los estudiantes aprendan a crear y

diseñar con un alto grado de precisión, aplicaciones complejas para la gestión y procesamiento automatizado de información.

Descripción de Algoritmos.

Los algoritmos se constituyen básicamente en un conjunto de instrucciones que, escritas en un orden secuencial adecuado, permite la resolución de prácticamente cualquier problema, de aquí su importancia para el desarrollo de programas. Al respecto (Universidad de Málaga, 1999) nos dice que la algoritmia es uno de los pilares de la programación, y su relevancia ha sido probada en el desarrollo de cualquier aplicación, no solo en la construcción de un determinado programa.

En su estructura más general un algoritmo está compuesto por tres partes, aunque no deben estar todas obligadamente, estas son:

Entrada.- Especifica el ingreso de datos o información, en caso de que sea necesario para realizar procesos o cálculos con dicha información.

Proceso.- En esta parte se detallan los procedimientos que deben realizarse para transformar o modificar la información que ha sido ingresada como insumos en el apartado anterior.

Salida.- En esta se muestran los resultados que se obtienen al realizar los cálculos en la parte de los procesos, también puede ser opcional, ya que en algunos casos no se requiere mostrar resultados, sino simplemente ejecutar un procedimiento.

Tipos de instrucciones.

A pesar de que los algoritmos son abiertos para ser escritos utilizando cualquier tipo de instrucciones, con el único requisito de que sean escritos en forma secuencial y ordenada, las instrucciones utilizadas se pueden clasificar, con el objeto de dotarlos de una estructura más adecuada, la clasificación que más se ha difundido es la siguiente:

Instrucciones de entrada/salida. - Estas instrucciones se utilizan para indicar que se deben ingresar información por un medio de entrada, generalmente a través del teclado, también se utilizan para mostrar los resultados de un proceso. Así tenemos que para ingresar datos se puede utilizar palabras como *escribir*, mientras que para indicar la salida de información se utiliza la palabra *mostrar*.

Instrucciones de control.- Estas instrucciones permite controlar el flujo de control de ejecución de las instrucciones, mediante la verificación del cumplimiento en determinadas condiciones, dentro de este grupo tenemos la instrucción *si... entonces*, y también la instrucción *en caso de*, que nos permite realizar comparaciones simultaneas

Instrucciones de repetición.- Son instrucciones que nos permiten repetir varias veces otro grupo de instrucciones, de esta manera podemos generar algoritmos mejor estructurados y más fáciles de entender, en caso de realizar posteriores revisiones minuciosas del proceso a realizarse.

Este conjunto de instrucciones es lo que se conoce más comúnmente como pseudocódigo, el cual una vez estructurado, puede ser llevado con suma facilidad a cualquier lenguaje de programación. Sin embargo, es necesario tener la seguridad que está adecuadamente estructurado y sigue una secuencia correcta, para evitar en lo posible errores en la ejecución o generación de resultados.

Todas las instrucciones y el flujo de ejecución de las instrucciones pueden y de hecho deben ser verificadas y comprobadas mediante procedimientos que se denomina pruebas de escritorio.(Rodríguez Monterosa, 2017)

La utilización de la prueba de escritorio servirá no solo para comprobar la veracidad del algoritmo, sino que permitirá ir viendo los valores que van tomando las variables, conforme se les vaya asignado valores. Esto es de vital importancia, ya que permitirá al estudiantado entender de una manera clara el funcionamiento de los algoritmos y el modo de operar de las diferentes instrucciones.

Generalmente, al iniciar a los estudiantes en la programación, se inicia creando algoritmos utilizando únicamente papel y lápiz, pero actualmente existen herramientas que nos permiten realizar estas tareas ya de una forma más automatizada, sin perder la esencia del diseño de algoritmos, ya sea en pseudocódigo o utilizando otra forma de representarlos como puede ser en diagramas de flujo o en diagrama de bloques.

Tipos de algoritmos

La secuencia de pasos de un algoritmo puede representarse de diferentes formas. La más sencilla, podría ser en forma de prosa, sin embargo, este método

no es recomendable ya que el lenguaje natural es muy amplio, lo que daría como resultado un algoritmo ambiguo y carente de precisión para describir la solución al problema.

La representación de un algoritmo requiere la utilización de un método que permita elaborarlo independientemente del lenguaje de programación que se haya escogido. Esto facilitará la codificación del algoritmo en cualquier lenguaje. Para lograr este objetivo es necesaria la representación gráfica o numérica de los algoritmos, de modo que los pasos y procesos que se lleven a cabo tengan total independencia de la sintaxis de los distintos lenguajes de programación, tratando de que la descripción realizada facilite su transformación en un programa o, dicho de otro modo, que la codificación pueda llevarse a efecto sin mayores complicaciones. (Joyanes Aguilar, 2008)

Los métodos más utilizados para representar algoritmos son:

- Pseudocódigo
- Diagramas de flujo
- Diagramas N-S

Pseudocódigo

Es un conjunto de instrucciones que describen la secuencia de pasos para resolver un problema dado, a diferencia de un algoritmo normal, el pseudocódigo combina el lenguaje natural con algunas notaciones sintácticas que son características propias de los lenguajes de programación, entre las que tenemos asignaciones, repeticiones y también las condicionales, aunque en la aplicación real no existe un estándar definido. Otro de los aspectos destacables es que, no ocupan demasiados espacios al momento de representar las instrucciones con mayor grado de complejidad. (Gomez & Salgado, 2017)

El pseudocódigo es especialmente aplicado como uno de los principios básicos en la enseñanza de la programación, ya que está diseñado para facilitar a los estudiantes la comprensión de los algoritmos, por lo que se pueden omitir detalles que no son importantes y que por lo tanto no son necesarios en la implementación de algoritmos. Por otro lado (Villadiego Alvarez et al., 2015) señala que el uso de pseudocódigo como paradigma de enseñanza de computación favorece en los estudiantes el desarrollo de la creatividad en diferentes ámbitos, pero para ello se requiere una perspectiva más audaz que

supere la simple productividad y busque en los estudiantes desarrollarse como pensadores creativos y puedan integrarse de una mejor manera a una sociedad creativa.

Se da el caso también de que los programadores utilizan a menudo diferentes convenciones, que pueden basarse en la sintaxis de un lenguaje de programación en particular. Sin embargo, en términos generales, el pseudocódigo es comprensible sin necesidad de comprender o utilizar un entorno de programación específico, y a la vez tiene la estructura suficiente para que sea implementado directamente en dicho entorno.

Tipos de instrucciones en pseudocódigo.

(Duque et al., 2017) menciona que, aunque no existe un estándar definido en una forma general, al utilizar el Pseudocódigo para elaborar algoritmos se pueden implementar los siguientes tipos de instrucciones, según sean necesarias, instrucciones de inicio-fin, cuya finalidad es delimitar claramente, el principio y final del algoritmo, también está la instrucción de asignación que se utiliza para expresar operaciones aritméticas, o también para inicializar o cambiara valores en las variables.

Por otra parte, están además las instrucciones de lectura y escritura que expresan la captura de datos y la salida de resultados, respectivamente. En la práctica no es necesario que todos los tipos de instrucciones estén presentes en el algoritmo, pues la implementación de cada tipo depende específicamente del problema que se está resolviendo.

Figura 1

Algoritmo representado en pseudocódigo

```
1 Algoritmo Sumar2Numeros
2   Definir B Como Entera
3   Definir A Como Entera
4   Escribir "Ingrese Un numero:"
5   Leer A
6   Escribir "Ingrese Un numero:"
7   Leer B
8   Si A > B Entonces
9       Escribir "Se realiza una suma:"
10      Resultado= A + B
11  SiNo
12      Escribir "Se realiza una Resta:"
13      Resultado= B - A
14  Fin Si
15  Escribir "El resultado es:"
16  Escribir Resultado
17 FinAlgoritmo
```

Nota: Ejemplo de Pseudocódigo. Captura realizada desde la aplicación PSeInt

Diagrama de Flujo

En el ámbito de la programación, los diagramas de flujo son representaciones gráficas de la secuencia de pasos y procesos que se realizan en los algoritmos, para llegar a la solución de un problema dado. (Shiguay, 2015) utiliza una simbología de figuras geométricas estandarizadas que, en forma similar al pseudocódigo, expresan instrucciones de inicio o fin, operaciones, decisiones, entrada y salida de datos.

Los diagramas de flujo son elaborados generalmente antes de iniciar la programación del código fuente, al ser una nomenclatura estandarizada, pueden ser traducidos en cualquier tipo de lenguaje de programación. Los diagramas de flujo ofrecen algunos beneficios, por ejemplo, son más fáciles de entender, por lo que facilitan significativamente la comunicación entre los programadores y los usuarios comunes o los clientes. También facilitan la comprensión de problemas complejos y sobre todo de los que resultan largos. (Rodríguez et al., 2018) Además, son prácticos a la hora de codificar desde un diagrama, pues una vez elaborado resulta fácil elaborar la codificación indistintamente del lenguaje de programación que se haya escogido, y por último constituyen un complemento ideal en la documentación de un programa, pues ofrece una significativa ventaja al tener que explicar el problema a otras personas.

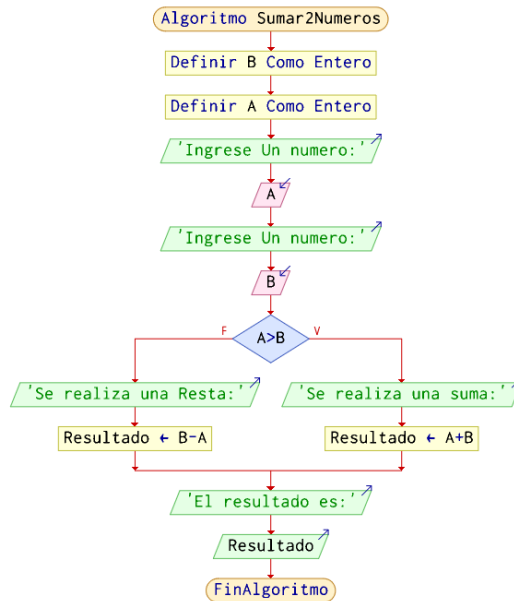
Reglas para dibujar diagramas de flujo.

(Rancel, 2021) muestra varias reglas que se deben tener muy presentes para una correcta elaboración de los diagramas de flujo, entre las principales se pueden mencionar que deben ser dibujados de arriba hacia abajo y de izquierda a derecha, para unir las figuras se utilizan únicamente líneas horizontales y verticales, que incluyen una flecha en un extremo para indicar el flujo de la información y los procesos.

Es importante evitar el cruce de líneas y tampoco deben existir líneas sueltas, pero si es necesario continuar el diagrama en sitio se deben utilizar los conectores, pero tomando en cuenta que los conectores solo se utilicen si son estrictamente necesarios y por último se debe utilizar la menor cantidad de palabras en los símbolos, y estas palabras deben ser claras y legibles.

Figura 2

Algoritmo representado en diagrama de flujo



Nota: Ejemplo de Diagrama de flujo. Captura realizada desde la aplicación Pseint

Diagrama S-N

El diagrama Nassi - Schneiderman (N/S) es una técnica que combina descripciones de texto utilizadas en pseudocódigo con representaciones gráficas de diagramas de flujo. (Zamora Lucio, 2018) En comparación con las dos primeras tecnologías, esta tecnología presenta una forma más compacta, en la que el conjunto de símbolos utilizado para representar los pasos a seguir por el algoritmo es muy limitado, por lo que para solucionar esta situación se utiliza expresiones del lenguaje natural que sean sinónimos de palabras propias de un lenguaje de programación (escribir, leer, repetir, ejecutar, etc.).

Los diagramas N/S se elaboran utilizando instrucciones de pseudocódigo y un conjunto simplificado de figuras básicas semejantes a una estructura de programación, en la que tenemos: secuencia, selección e iteración. No hace uso de flechas para indicar el flujo de las operaciones, sino que coloca en las casillas correspondientes a las instrucciones una seguida de otra o dentro en caso de ser necesario, de manera que se pueda apreciar la secuencia, bifurcación o repetición cuando se ejecuta el algoritmo. (Chavez Torres, 2017)

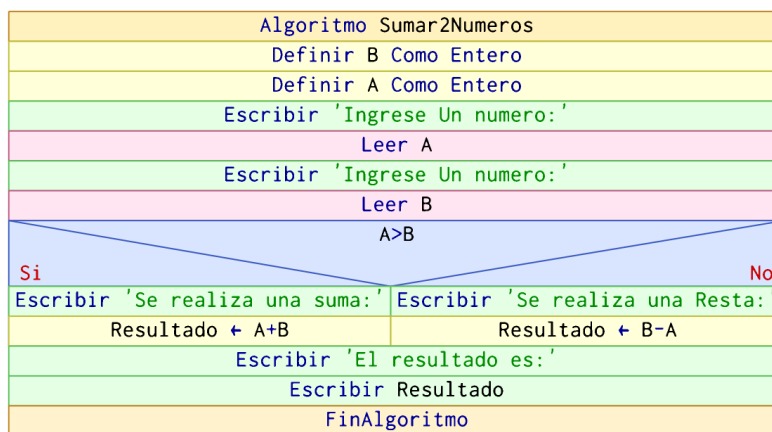
Los creadores de este método afirman que los diagramas por ellos diseñados presentan las siguientes ventajas:

- Las estructuras de decisión se visualizan y comprenden con facilidad.
- Las iteraciones están muy bien definidas.
- El alcance de las variables locales y globales es evidente en el diagrama.
- Jamás se transfiere el control arbitrariamente.
- La recursividad se representa de forma trivial.
- Los diagramas se adaptan fácilmente a las singularidades del lenguaje de programación que se escoja para la codificación.

A diferencia del pseudocódigo y los diagramas de flujo, los diagramas N-S permiten que los pasos del algoritmo se vean de una manera más estructurada, por lo que no solo es útil para el siguiente paso, que es la codificación, sino que también es fácil de entender y aprender.

Figura 3.

Algoritmo representado en diagrama de bloques N-S



Nota: Ejemplo de algoritmo utilizando Diagrama N-S. Captura realizada desde la aplicación PSeInt

Lenguaje de programación

(Trejos Buriticá, 2017) define este concepto como un conjunto de instrucciones que una computadora puede comprender y ejecutar, tienen su propia gramática y, por lo general, tienen reglas de entorno y desarrollo.

(Joyanes Aguilar, 2008) indica que los lenguajes de programación se utilizan para escribir programas que permiten la comunicación entre el usuario y la máquina. Un programa especial llamado traductor (compilador o intérprete) convierte instrucciones escritas en un lenguaje de programación en

instrucciones escritas en lenguaje máquina (bits 0 y 1) que la máquina puede entender.

Como se dijo anteriormente, cada lenguaje posee su propio conjunto de instrucciones, dependiendo del propósito para el que se ha creado, algunos usan exclusiva instrucciones de texto, siendo estos los más básicos y además utilizados para enseñar los principios básicos para adquirir la destreza de la programación. Dentro esta categoría están los lenguajes de programación PASCAL o C++.

Actualmente se utilizan herramientas con ambiente gráfico, se los conoce como lenguajes orientados a objetos, que constituyen todo un nuevo paradigma de programación y a la vez poseen herramientas que vuelven más amigable el diseño de programas y aplicaciones. Algunos ejemplos de estos lenguajes son Visual Basic, Java o Phyton.

Herramientas TIC'S en la enseñanza de programación

Desde el surgimiento de la educación técnica en las instituciones educativas, una de las figuras profesionales que más se ha venido implementado es la del estudio de la informática bajo diferentes denominaciones, justificado con el avance vertiginoso de la tecnología, ante lo cual surge la necesidad de contar con personal preparado para afrontar los retos que implica la modernización tecnológica y sobre todo la preponderante conectividad en la que vivimos inmersos actualmente.

La implementación de estas asignaturas implica precisamente un fuerte aprendizaje del uso de varias herramientas TIC's, que buscan desarrollar en los alumnos las destrezas necesarias y suficientes que le permitan manipular distintos dispositivos tecnológicos y gestionar aplicaciones para un adecuado manejo de la información y poder de esta manera, satisfacer las demandas cada vez más altas del ambiente laboral y empresarial.

La enseñanza de la programación se inicia en el primer año de bachillerato, siempre con la mejor predisposición de los estudiantes, sin embargo, el entusiasmo inicial disminuye paulatinamente debido a que la sintaxis del lenguaje utilizado resulta complicada. Por otro lado, en la enseñanza de programación se suelen proponer actividades que no están relacionadas con el interés del alumno. Además, muchas veces al trabajar en estos entornos no

se proporciona una guía para la corrección de los errores cuando estos se presentan.

Ante esto se ha propuesto implementar desde el inicio herramientas amigables e interactivas que a través de módulos de una guía para el estudiante y herramientas para el seguimiento y control de errores puedan por si solos, adquirir las destrezas de razonamiento lógico para la resolución de problemas. Para ello, es fundamental identificar los obstáculos que traen las herramientas, porque esto permite investigar y encontrar soluciones creativas para superar estos obstáculos. De esta manera, será posible abrir el alcance de los recursos a otras herramientas más complejas que, aunque no son tan populares, brindan alternativas efectivas para el desarrollo de la enseñanza de la programación y el pensamiento computacional.(Monjelat et al., 2018)

Herramientas para la introducción a la programación

La enseñanza de la programación no se limita a la interpretación de conceptos y teorías, sino que también busca desarrollar habilidades para ser utilizadas en conjunto con conceptos abstractos, como el procesamiento de datos a través de algoritmos, la sintaxis y semántica de los lenguajes de programación, y el metalenguaje utilizado en programación para desarrollar en primera instancia algoritmos y a partir de allí generar los programas.(Zuleta Medina & Chaves Torres, 2011)

De hecho, los algoritmos constituyen generalmente la herramienta para iniciar a los estudiantes en la programación de aplicaciones. Esto en primer lugar porque sirve de plataforma para mostrar a los estudiantes una secuencia lógica de pasos a seguir que permiten resolver cualquier tipo de problemas, desde los más cotidianos hasta luego avanzar a problemas que pueden requerir razonamiento matemático.

Se consideran dos herramientas básicas con las que se pueden realizar este cometido, estas son PSeInt y un lenguaje de texto estructurado, en este caso puede ser C++.

Software PseInt

La herramienta PseInt, es sencilla pero poderosa aplicación creada precisamente para iniciar a los estudiantes en el estudio de la construcción de programas o en el diseño de algoritmos computacionales.(Carrizo et al., 2015)

Fue creada por el argentino Pablo Novara y es de licencia GPL con código abierto, es decir es totalmente gratuito y no requiere de ningún pago para poder usarlo.

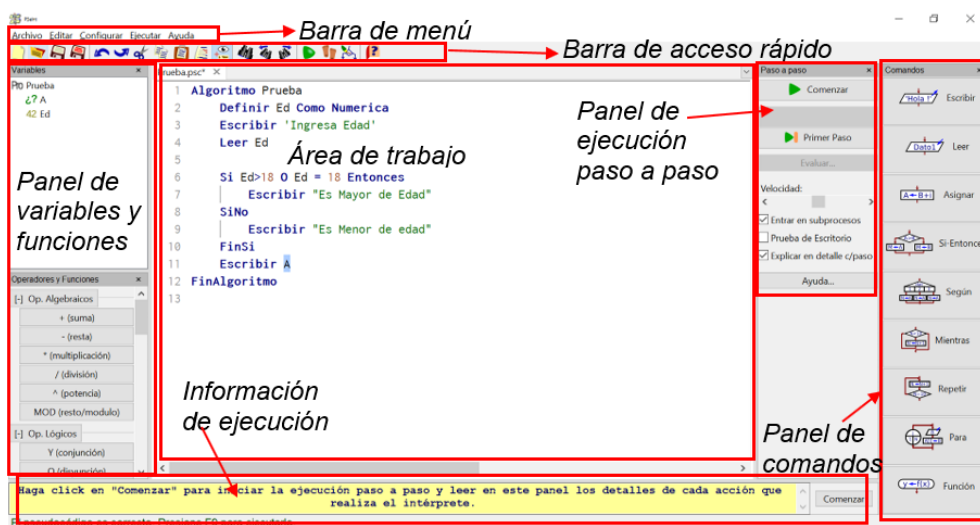
Este programa hace uso del pseudocódigo, el cual constituye una descripción más estructurada de un algoritmo y tiene como objetivo principal que el programador pueda centrarse totalmente en los aspectos lógicos de la programación, quedando la parte técnica para el momento en el que se lleve el algoritmo ya desarrollado a un verdadero lenguaje de programación. (Castillo & Caro, 2016)

De hecho, al crear el algoritmo en pseudocódigo, se puede generar automáticamente el diagrama de flujo del algoritmo realizado, o a la inversa, se puede generar el diagrama de flujo, y partir de ahí la herramienta generará el pseudocódigo del algoritmo resultante.

Cuando los alumnos especifiquen el algoritmo a través del pseudocódigo propuesto por la herramienta, se generará al mismo tiempo un diagrama de flujo. De esta manera, la integración de esta herramienta acelera y dinamiza significativamente la velocidad de aprendizaje del alumno con respecto a la construcción de algoritmos y diagramas de flujo. (Anderson Huerta & González-Bañales, 2020)

Figura 4.

Interfaz de la aplicación Pseint



Nota: Interfaz principal de la aplicación de PseInt. Captura tomada de la aplicación

Tal como se puede apreciar en la figura 4, la herramienta PSeInt ofrece todas las opciones que se encuentran comúnmente en una aplicación de Windows para elaborar algoritmos, guardarlos, y modificarlos en el momento que se desee. Pero lo que le hace especialmente ideal para la tarea de elaborar algoritmos es la capacidad de realizar un seguimiento del programa con la ejecución paso a paso de las instrucciones. Este aspecto muchas veces subestimado y poco utilizado, resulta fundamental para entender el algoritmo, el comportamiento de las variables, así como los valores que van tomando hasta llegar a la finalización del programa. Esto permite verificar la ejecución de las instrucciones en tiempo real, el orden en la que se ejecutan.

PSeInt dispone de un sencillo conjunto de instrucciones, realmente fácil de utilizar, muy similar a un lenguaje de programación, con la particularidad de que está en español y resulta por lo tanto fácil de entender. Buscando de esta manera desarrollar en el estudiante el pensamiento algorítmico, que consiste en habilidades relacionadas con la construcción y comprensión de algoritmos, tales como: analizar un problema dado, determinar con precisión el problema, encontrar las acciones básicas adecuadas para resolver un problema dado y construir el algoritmo correcto para realizar las acciones que conlleven a solucionar dicho problema, teniendo todas las posibles soluciones y verificando la eficiencia del algoritmo. (Enrique et al., 2019)

Lenguaje de programación C++

C++ es un lenguaje de programación de propósito general, muy utilizado para desarrollar una gran diversidad de aplicaciones, de hecho, muchas aplicaciones y hasta sistemas operativos completos han sido desarrollados y programados utilizando el lenguaje de programación C++.

Para elaborar programas, o dicho sea de otro modo, para pasar a código los algoritmos previamente elaborados, ya sea en papel o si fue elaborado en herramientas como PSeInt, se utilizan lo que se denominan compiladores.

Se pueden encontrar una gran variedad de compiladores en el mercado, algunos de los cuales son gratuitos y fácilmente descargables desde internet, mientras que también existen versiones profesionales, que son comercializados por diversos fabricantes. Algunos de los compiladores gratuitos son los

siguientes: Apple C++, Borland C++, Dev-C++ de Bloodshed, Microsoft Visual C++ Toolkit 2003, entre otros. (Joyanes Aguilar & Sanchez García, 2006)

En los inicios de la enseñanza de la programación en C++, se utilizaba un rustico programa con ambiente bastante precario, podría decirse en el que incluso no se podía utilizar el mouse de la computadora, sino que todas las instrucciones y los códigos se escribían utilizando el teclado. Pero actualmente podemos hacer uso lo que se denomina Entornos de Desarrollo Integrado (IDE por sus siglas en ingles).

Estos entornos facilitan enormemente el proceso de aprendizaje, ya que además del ambiente grafico que ofrece para presentar varias de las opciones, proporcionan además asistencia al momento de generar el código; esta asistencia consiste básicamente en mostrar pequeños cuadros con explicaciones de los que se puede realizar o errores que se estén dando al escribir el código, y también presentar opciones de autocompletar ciertas instrucciones, lo que hace la tarea programación significativamente más fácil y en cierto modo hasta más divertido.

Como en el caso de los compiladores, también existen diversos IDE's que permiten programar en C++, algunas de estas herramientas son de pago, es decir se debe adquirir una licencia para poder usarlas, se caracterizan por ser muy potentes y con una gran variedad de opciones. Por otro lado, también existen herramientas que son gratuitas y de código abierto, pero a pesar esto son muy potentes a la hora de generar programas en C++. Tenemos en este caso la herramienta *Dev C++* y la aplicación *Code::Blocks*, los cuales se describe brevemente a continuación.

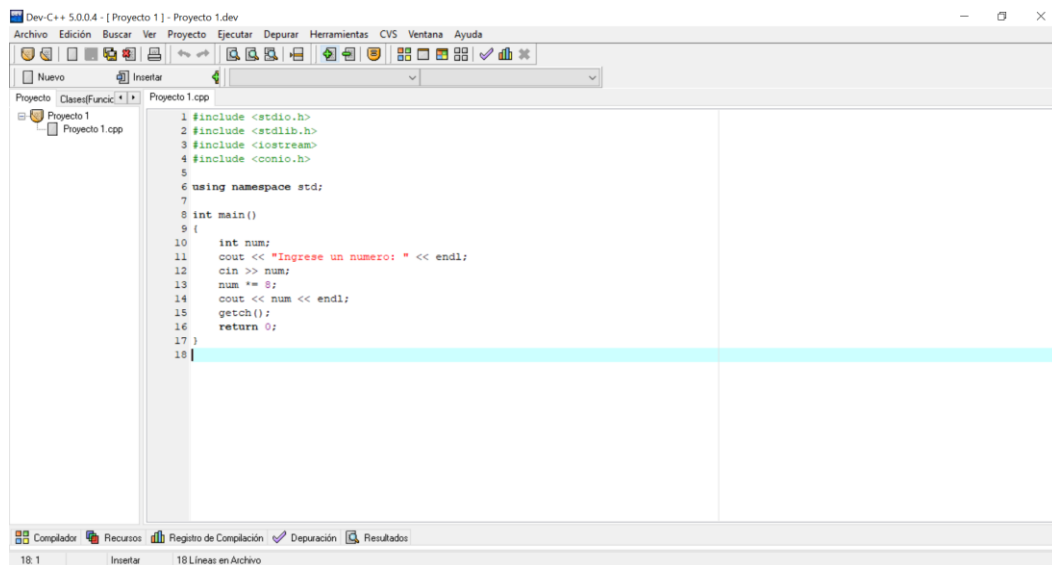
La herramienta Dev C++

Se trata de un magnifico entorno de desarrollo integrado para programar utilizando el lenguaje de programación C++. Posee integrado el compilador GCC(GNU Compiler Collection) aunque también se puede usar en combinación con otros compiladores basados en GCC. Permite crear programas ejecutables para el sistema operativo Windows, ya sea en modo consola o en modo gráfico. Se trata de una aplicación de código abierto, disponible para descargar desde internet.

Dentro de su interfaz, la aplicación presenta resaltado de sintaxis, depurador de código, varias librerías y archivos con ejemplos, etc. Como opciones extra, Dev-C++ incluye múltiples idiomas (incluido el español), explorador de clases y variables, plantillas, facilidad para instalar librerías adicionales, etc. Todo esto sólo es una pequeña parte de las posibilidades que se pueden encontrar en Dev-C++, un entorno de programación muy recomendado para el aprendizaje de programación.

Figura 5.

Interfaz de la aplicación Dev C++



Nota: Código del lenguaje de programación C++ escrito de la aplicación Dev C++. Captura tomada del programa Dev C++.

La herramienta IDE Code::blocks

Code::blocks, es una herramienta gratuita y su código está disponible en su portal de internet. Se trata de una herramienta muy versátil de fácil instalación e incluye en su configuración varios compiladores compatibles del lenguaje de programación C++, entre ellos GCC (MingW / GNU GCC), MSVC , Digital Mars, Borland C 5.5 u Open Watcom. El compilador que viene por defecto con este paquete de Code Blocks es MinGW.

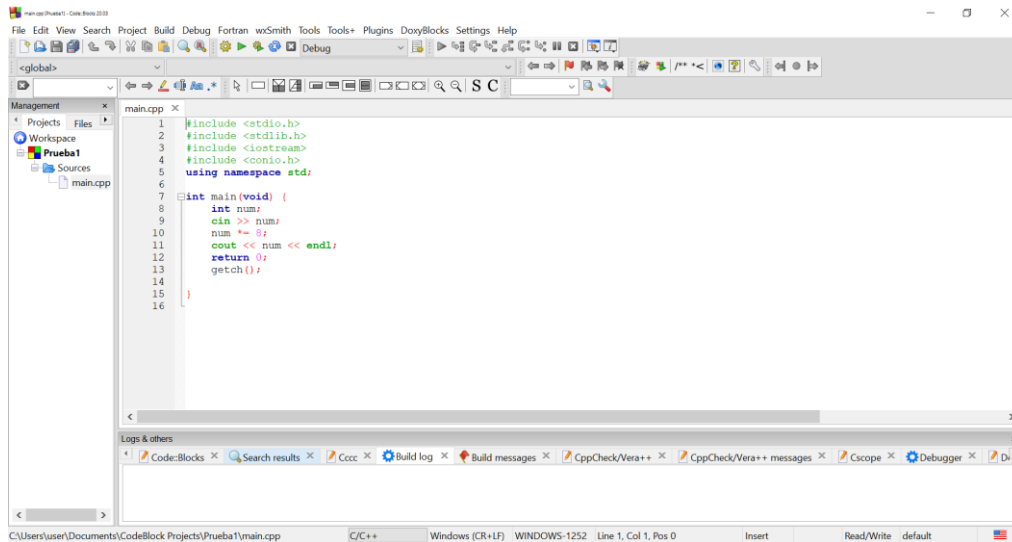
Presenta un ambiente grafico bastante agradable y muy fácil de utilizar. Tiene la particularidad de concebir cada programa que se desee editar como un proyecto, lo que hace que todos los archivos relacionados sean agrupados en una carpeta que normalmente lleva el nombre del proyecto.

Lo que hace destacable de esta herramienta es que colorea la sintaxis del código fuente, incluye la función de agrupación de código o 'code folding' para los lenguajes C y XML, posee una interfaz con pestañas, cuenta con un explorador de clases y una función para completar el código fuente.

La pantalla principal de Code::Blocks es la siguiente:

Figura 6

Interfaz de la aplicación Code::Blocks



Nota: Código de programa del lenguaje de programación C++ realizado en la aplicación Code::Blocks. Captura tomada del programa Code:Blocks

Lenguaje de programación Python

Se trata de un lenguaje programación de alto nivel, intuitivo y fácil de usar y aprender. Fue creado en el año 1989, por el entusiasta de la programación Guido van Rossum y poco a poco se ido convirtiendo en un lenguaje referente para la enseñanza de programación, por su facilidad de uso y sintaxis flexible. Python es un intérprete de instrucciones,(Ojeda, 2015) lo que permite usar el lenguaje de forma interactiva con gran conjunto de instrucciones que además de programar en forma estructurada, también incluye programación orientada a objetos, lo que lo hace ideal para desarrollar aplicaciones de propósito general, que ha ido ganando terreno poco a poco en el desarrollo aplicaciones web y aplicaciones de escritorio de todo tipo.(Monsálvez, 2017)

Ventajas

- Puede ejecutar el código en cuanto lo complete; no hay fases adicionales de traducción.

- El código totalmente portable, ya que se almacena en el mismo lenguaje de programación; lo que implica que puede ser compilado en cualquier computadora, aunque estas tengan diferentes lenguajes de máquina. Lo que asegura una compilación única al momento de ejecutarse el programa.

Desventajas

- El código compartirá la potencia de la computadora con el intérprete, por lo que no puede ser realmente rápido.
- Tanto el programador como el usuario final deben tener el intérprete para ejecutar su código.

Existen diversas formas de crear aplicaciones en Python, empezando desde la más básica como es la codificación en una terminal de consola, hasta los editores dinámicos y flexibles, que conllevan eventualmente a constituirse en un desarrollador Python. Para comenzar a trabajar es necesario contar con las siguientes herramientas:

- Un **editor** que servirá para escribir el código (debe tener algunas características especiales, no disponibles en herramientas simples); este editor dedicado proporciona más opciones que el equipo estándar del sistema operativo.
- Una **consola** en la que se puede iniciar el código recién escrito y detenerlo por la fuerza cuando se sale de control.
- Posee también un depurado, que tiene la capacidad de realizar la ejecución del código paso a paso, permitiendo su inspección paso a paso en todo momento.

La sintaxis de las instrucciones de Python es mucho más sencilla que en otros lenguajes de programación, pues a pesar de ser Case Sensitive, no requiere especificar símbolos como el punto y coma para finalizar una instrucción o el doble igual para la comparación de operadores. Aunque debe respetarse el uso de mayúsculas y minúsculas en el nombre de las sentencias y las funciones, ya sean propias del lenguaje o creadas por el programador.

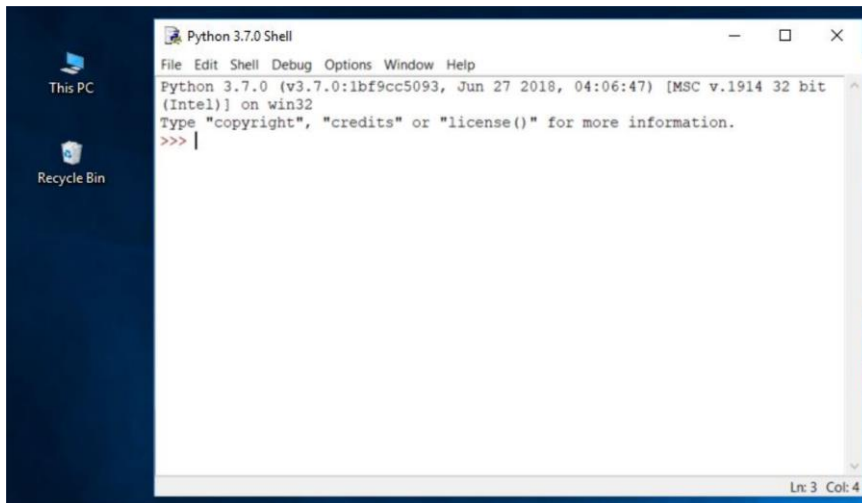
Además de sus muchos componentes útiles, la instalación estándar de Python 3 contiene una aplicación muy simple pero extremadamente útil llamada IDLE por su nombre en inglés. Integrated Development and Learning

Environment (Entorno Integrado de Aprendizaje y Desarrollo). Dicha herramienta viene incorporada al compilador, lo que permite la generación de código de manera rápida y sencilla.

La ventana de la herramienta nativa del lenguaje de programación Python es la siguiente:

Figura 7

Interfaz del entorno de programación Python



Nota: Pantalla principal de la aplicación IDLE de Python. Captura tomada del programa IDLE de Python.

Esta es una herramienta muy útil para dar los primeros pasos de programación en el lenguaje de Python, por su sencillez y fácil manejo, sin embargo, existen actualmente en el mercado muchas herramientas que permite utilizar el lenguaje de programación de una forma mucho más dinámica y con la capacidad de desarrollar programas de mayor grado de complejidad.

Existen herramientas gratuitas y de pago, tal es el caso de Pycharm, que tiene dos versiones, una profesional que por supuesto es de pago y también una versión para la comunidad Python, que es gratuita, pero que no incluye todas las opciones de la versión profesional. Por otro lado, también están herramientas totalmente gratuitas y muy funcionales como por ejemplo Anaconda Python o Visual Estudio Code, siendo este último del que se hará una breve descripción.

Visual Studio Code

Visual Studio Code es una potente herramienta para programar en diferentes lenguajes de programación. (Microsoft, 2020) Se la puede descargar

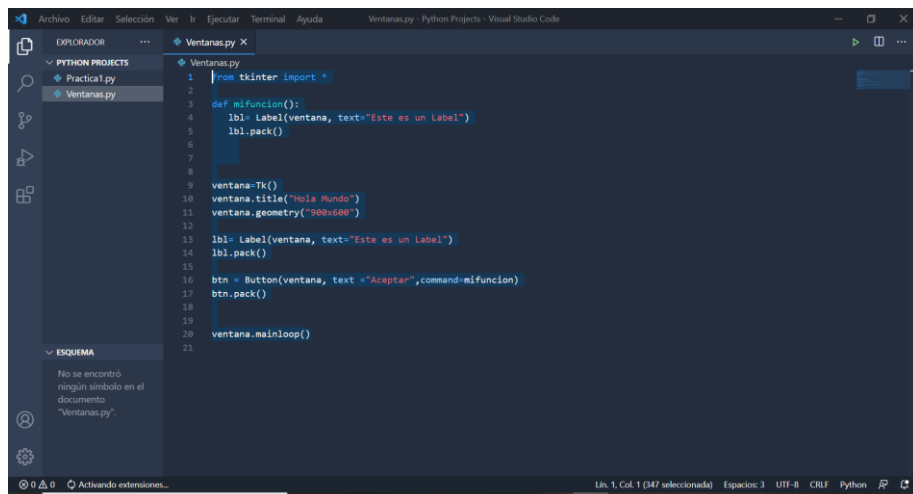
de internet sin ningún problema, ya que es una herramienta gratuita y muy personalizable, se pueden complementar con diferentes extensiones que le permiten mejorar el aspecto, como, los colores del texto del código que se genere en el área de trabajo o el idioma de la aplicación.

En su entorno posee el clásico menú de opciones, un explorador de carpetas y archivos que facilita el acceso a los mismos, barras de herramientas con iconos para acceder con un solo clic a las distintas opciones de la herramienta.

Cabe mencionar que se trata únicamente de un editor de código que se adapta perfectamente al lenguaje de programación que se esté editando, por lo que, si desea programar en tal o cual lenguaje de programación, el compilador correspondiente debe ser instalado aparte en la computadora. De ahí en adelante Visual Studio Code, reconocerá con que compilador está programando, por la extensión con la que sea grabado el archivo que va a contener el código. La interfaz es la siguiente:

Figura 8

Interfaz de la herramienta Visual Studio Code



Nota: Pantalla principal de la aplicación Visual Studio Code. Captura tomada del programa Visual Studio Code

Una vez definido el editor a utilizar, en cualquiera de los casos lo primero que se debe hacer crear un nuevo archivo y darle un nombre al mismo, este archivo debe ser guardado con la extensión py, para que sea reconocido como un archivo fuente del lenguaje de programación Python, tanto por el editor nativo IDLE, como por el entorno Visual Estudio Code.

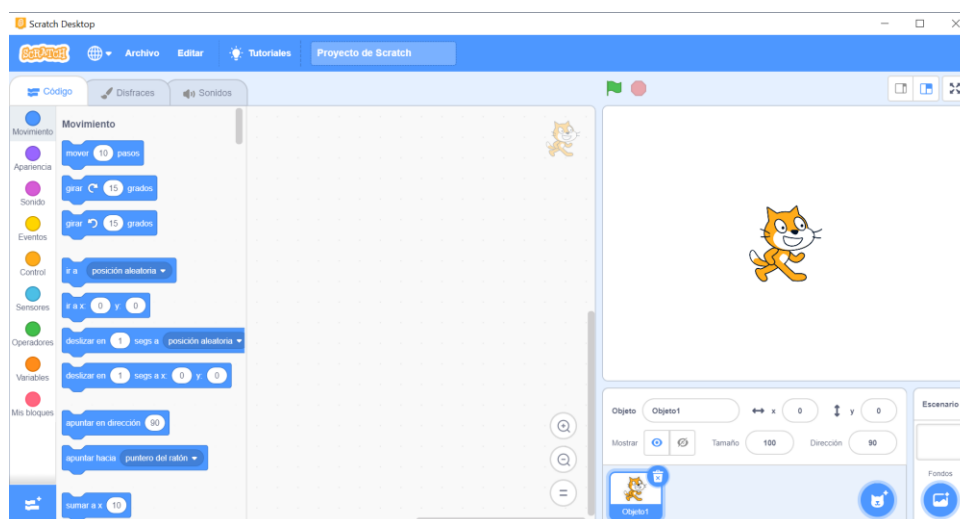
Herramienta de Programación Scratch

Scratch es un entorno de programación gráfica gratuito que permite crear fácilmente historias, juegos y animaciones interactivas, y compartir su trabajo con otras personas a través de internet. Originalmente fue diseñado para ser usado por los niños, permitiéndoles crear sus propios videojuegos, historias animadas y arte interactivo. (García López, 2013) Sin embargo, en muy poco tiempo, su audiencia continuó expandiéndose y atrajo exitosamente a maestros de todo el mundo, quienes comenzaron a usarlo durante sus clases en el aula.

Scratch hace uso de lo que se llamarían "piezas encajables" para animar objetos en la pantalla de una manera muy simple e intuitiva. El uso de Scratch se realiza desde su propio entorno de desarrollo, el cual muestra en una sola pantalla todos los elementos necesarios de un vistazo: paisajes, objetos y elementos del lenguaje. Permite utilizar los esquemas y objetos necesarios, ya proporcionados en la instalación estándar de la herramienta o crear nuevos esquemas y objetos. (Eug et al., 2016) Este es otro factor de motivación cuando se inicia los estudiantes en la programación. En esta modalidad, todas las operaciones se realizan usando el mouse para arrastrar y soltar elementos, usando el teclado únicamente para modificar solo los valores y el texto. La interfaz de Scratch es la siguiente.

Figura 9

Interfaz de la aplicación Scratch



Nota: Ventana Principal de la herramienta de programación Scratch. Captura realizada de la aplicación instalada.

Utilizar la herramienta Scratch para el aprendizaje programación ayuda a comprender fácilmente los conceptos matemáticos e informáticos que se encuentran bien integrados en el programa, tales como: Los procesos interactivos, los criterios condicionales, las coordenadas en un plano, los indicadores, etc.(Sormenezko Zerbitzuak, 2014)

Estos conceptos se aprenden en un contexto significativo y motivador. Por ejemplo, comprender el significado de las variables en el contexto del aprendizaje de programación tradicional no es lo mismo que entender cuando las variables se utilizan para controlar la visualización de animaciones o juegos que el mismo programador construye.

Los alumnos trabajan en sus propios proyectos con Scratch, de esta manera, pueden comprender sobre el proceso de diseño. A partir de la idea, deben crear prototipos funcionales (es decir, modelos) e intentar utilizar las soluciones que crean que son necesarias para llegar al producto final. Cuando estas ideas no funcionen, tendrán que volver atrás y corregirlas. Se crea una espiral continua: partiendo de una idea, se crea un proyecto, y el proyecto da lugar a nuevas ideas, y estas ideas dan lugar a nuevos proyectos, así que sucesivamente.

Aprender a programar se trata esencialmente de aprender el pensamiento lógico y aprender a resolver problemas de manera ordenada. La mayoría de la gente considera la programación informática como una actividad tediosa y compleja a la que solo pueden acceder personas con formación técnica avanzada. Los componentes que usan los programadores para construir soluciones son simples: secuencia, iteración y condición. En Scratch se han implementado estos bloques y para poder utilizarlos, basta con arrastrarlos al programa a construir. Además, puede cambiar dinámicamente el bloque de código y observar el resultado de inmediato.(Travieso Merino, 2017)

Gracias a esta modalidad de programa en Scratch se pueden resolver problemas rápidamente sin conocer la sintaxis, lo que es diferente a los lenguajes de programación tradicionales donde es necesario aprender a leer y escribir las instrucciones o comandos antes de comenzar a usarlos.

Por otro lado, Scratch atrae a los estudiantes, permitiéndoles cometer errores y copiar lo que otros están haciendo, que es la motivación para aprender. Permite a los estudiantes aprender de forma independiente y

compartir con sus compañeros a través de foros oficiales o con compañeros de clase y amigos. Además, tienen la posibilidad de aprender en una forma muy interesante lo que favorece retener los conceptos realmente importantes a largo plazo.

Diferentes temas pueden ser enfocados desde Scratch por lo que un alumno puede desarrollar un proyecto de cualquier asignatura utilizando los diferentes recursos que la herramienta pone a su disposición de este modo, Si "hace" en lugar de simplemente "mirar" y "escuchar", el aprendizaje en el estudiante será más eficaz.

De esta forma, Scratch es una herramienta muy útil y adecuada como recurso transversal, que apoya la enseñanza eficaz y aumenta en gran medida la motivación para aprender conceptos de otras áreas del conocimiento.(Cearreta-Urbieta, 2015) Por ejemplo, los alumnos pueden desarrollar proyectos con Scratch que representen hechos históricos, conocer los ríos de una determinada región de forma sencilla o modificar vocabulario en inglés relacionado con "elementos de cocina" con programas realizados por profesores o también desarrollando sus propias aplicaciones interactivas.

Por otro lado, al ser una herramienta gratuita, permite a los alumnos seguir trabajando desde casa según sus deseos. El uso de Scratch no se limita al aula, sino que también se aplica a cualquier lugar donde haya una computadora. Scratch es una herramienta muy fácil de usar, incluso si el nivel de programación inicial es cero, los resultados se pueden obtener muy rápidamente. Por tanto, también es muy adecuado para un entorno familiar donde padres e hijos puedan adentrarse juntos en el mundo de la programación. De esta forma, el tiempo que pueden ayudar a tus hijos a "pasar" en la computadora es para su propio bienestar, desarrollar sus habilidades y aumentar sus conocimientos.

En resumen, Scratch es una herramienta muy flexible y adaptable que se puede utilizar en cualquier situación, es muy eficaz para enseñar múltiples disciplinas mientras los estudiantes desarrollan y exploran todo su potencial y se benefician de la programación.

Relación de Scratch y las competencias básicas

(Sormenezko Zerbitzuak, 2014) sostiene que son varias de las competencias básicas que se pueden fomentar en los alumnos de educación secundaria, entre las principales se detallan la siguientes:

Competencia en comunicación lingüística: Hoy en día, la comunicación eficaz requiere no solo leer y escribir, sino también más habilidades. Con la ayuda de Scratch, los jóvenes aprenden a ser capaces de manipular e integrar varios tipos de información para expresarse de forma creativa y persuasiva.

Tratamiento de la información y competencia digital: Al usar Scratch, los estudiantes pueden aprender a seleccionar, crear y procesar varios tipos de información: texto, imágenes, secuencias de animación y sonidos. A medida que los estudiantes adquieren experiencia en el procesamiento de esta información, se vuelven cada vez más receptivos y críticos cuando analizan información del mundo que los rodea.

Identificación de problemas, formulación de hipótesis y solución: Scratch permite a los estudiantes aprender en un contexto significativo basado en el proceso de diseño. Para crear un proyecto con Scratch, debe tener una idea, dividir esa idea en varios pasos e implementar estos pasos a través del sistema de configuración de bloques del programa. Está diseñado para poder ver los resultados de la programación en el momento, para que los estudiantes puedan aprender el proceso de forma interactiva.

Competencia social y ciudadana: Al ser una herramienta muy fácil para compartir proyectos, se puede utilizar para animar a los jóvenes a debatir sobre temas de importancia social, no solo en un entorno educativo, sino también para discusiones globales, gracias a la comunidad Scratch.

Competencia para aprender a aprender: Al aprender a codificar con Scratch, los jóvenes pueden descubrir el razonamiento crítico y el pensamiento sistemático. En sus proyectos, necesitan coordinar el tiempo y la interacción entre los diferentes roles, su capacidad de programación les proporciona una experiencia directamente relacionada con la detección de problemas, crítica constructiva, ensayo y error, etc. conceptos fundamentales en el pensamiento sistemático.

Al trabajar en proyectos que son significativos para los jóvenes, sus propias ideas les proporcionan la motivación adecuada para sobrellevar las dificultades y retos que les plantea el proceso de diseño de sus programas.

Los proyectos de Scratch están contruidos con Objetos (Objeto móvil programable es la traducción al español dada a Sprite, en adelante se utiliza solo la palabra Objeto). Se puede modificar el aspecto cómo se ve un objeto dándole un disfraz diferente. también es posible hacer que el objeto parezca una persona, un tren, una mariposa o cualquier otra cosa. Se puede complementar con cualquier imagen como disfraz: puede dibujar una imagen en el Editor de Pinturas o importar una imagen del disco duro de su equipo. (García López, 2013)

Además, la herramienta permite dar instrucciones al objeto, indicándole que se mueva o toque música o que reaccione a otros objetos. Para señalar a un Objeto qué hacer, se encajan bloques gráficos unos con otros formando pilas, llamadas programas (scripts). Esto se hace dando doble clic sobre un programa, Scratch ejecuta en orden los bloques desde la parte superior del programa hacia abajo.

Una de las cosas más interesantes de Scratch es la capacidad de utilizar materiales capacidad de utilizar materiales externos a la propia herramienta, elaborados por el alumnado, de manera que ellos preparan su propio material(sus fotografías, sus dibujos e imágenes, su música...) que puede ser incorporado al programa a través de las opciones de opciones de importación opciones de importación.

Otra de las características que convierten a Scratch en una herramienta colaborativa interesante es la posibilidad de compartir las creaciones que son desarrolladas con la herramienta en la misma Web de la aplicación, a través de la opción "Compartir", disponible desde el propio entorno de desarrollo. Para esto hace falta haber creado previamente un perfil (una cuenta de usuario) en la web oficial de Scratch. El propio web programa se encarga de comprimir todo el material utilizado y enviarlo a la web.

También se puede fomentar entre el alumnado el compartir recursos, sin necesidad de mandarlos a la web. Así todos puedan utilizar los materiales elaborados por todos, y así crear trabajos derivados de los mismos.

Los estudiantes de bachillerato, aunque no hayan tenido un acercamiento anterior con Scratch, en poco tiempo estarán en la plena capacidad de trabajar más profundamente los conceptos abstractos. Se puede comenzar a asignar eventos a los personajes, a utilizar el teclado y el ratón para manejar los objetos y sobre todo a asignar acciones que ocurren cuando varios objetos interaccionan entre sí. Así se pueden conseguir efectos interesantes, que se irán complicando sucesivamente según se vayan adquiriendo nuevas habilidades.

Visual Basic.Net

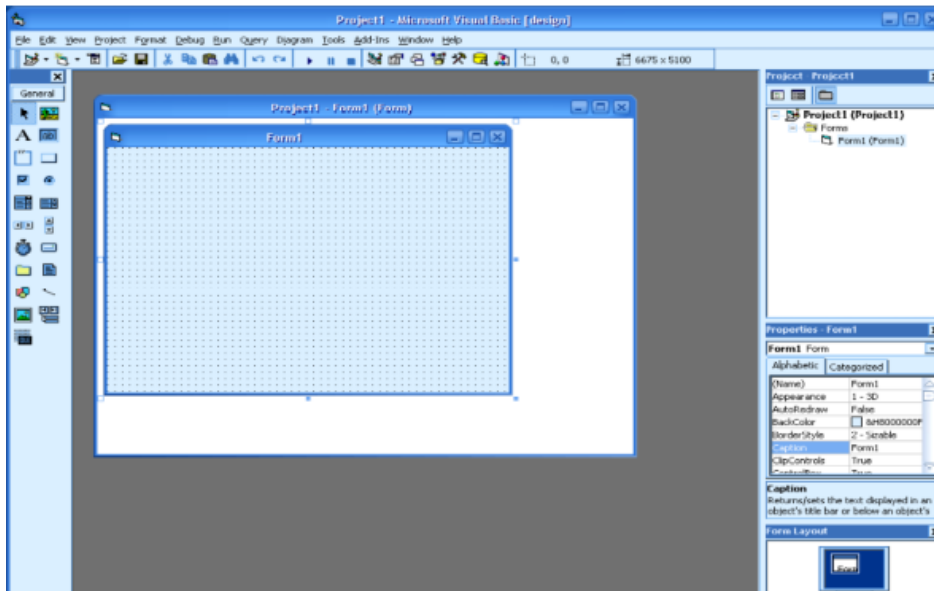
Esta una de las herramientas más utilizadas en la asignatura de programación especialmente en los niveles mas avanzados, esto debido al ambiente visual que proporciona en el que están presentes todos los elementos que tiene una ventana común en el ambiente Windows, permite realizar muchas tareas de diseño sin tener que escribir código, sino únicamente realizando algunas acciones sobre la pantalla principal.(Moreno Montiel & Arevalo Gamboa, 2018)

Visual Basic es un lenguaje de programación orientado a objetos creado por Microsoft, forma parte del Paquete Visual Studio, que incluye además otros lenguajes de programación. Este lenguaje contiene todas las herramientas necesarias para crear aplicaciones de Windows. (Rodríguez Bucarely, 2008)Con este lenguaje, puede crear cualquier cosa, desde simples calculadoras hasta auténticos procesadores de texto, aplicaciones comerciales como, por ejemplo, aplicaciones de facturación, programas para llevar la contabilidad de una empresa o cualquier aplicación que se le ocurra a un programador.(Livias Segura, 2018)

La principal fortaleza de Visual Basic es su potente Entorno de Desarrollo Integrado, tiene muchas herramientas que hacen que la programación de aplicaciones de Visual Basic sea más fácil e interactiva. Este entorno incluye elementos tales como: barra de menús, barra de controles, barra de herramientas, ventana de propiedades, ventana de proyectos, depurador, formularios, etc. Todos estos elementos se pueden apreciar desde que se ejecuta la ventana de Visual Basic.

Figura 10

Interfaz del lenguaje de programación Visual Studio



Nota: Ventana Principal de Visual Basic. Tomada de Libro de Oro de Visual Basic (Rodríguez Bucarely, 2008)

Entre los elementos más destacables que contiene el entorno de Visual Basic están la caja de herramientas, que contiene todos los objetos básicos, más comúnmente denominados *controles*, para diseñar un programa, como son las cajas de texto, botones de comando, etiquetas, etc. Para ubicarlos en el área de trabajo basta con dar doble clic o arrastrar el control seleccionado y hasta el área de la ventana que se está diseñando.

El otro elemento que también es muy importante es la ventana de propiedades, en esta parte es donde se pueden modificar el aspecto y atributos de cada uno de los controles ubicados en el área de desarrollo, lo cual vuelve más fácil e intuitiva la tarea de personalización de las ventanas, añadido a esto una reducción considerable del tiempo dedicado a la aplicación, ya que al hacerlos de este modo, el código de los objetos es generado en forma automática y totalmente transparente; de esta manera, el programador puede centrarse más en los aspectos lógicos y de presentación de aplicación que se está diseñando.

A diferencia de otros lenguajes de programación, en Visual Basic, la programación no se hace en forma secuencial, sino más bien, como se trata de un lenguaje orientado a objetos, la programación está basada en su mayoría

en eventos, es decir el código de las acciones que o comportamiento se programan en los distintos eventos de cada control.(Bravo Macías & Montesdeoca Zuleta, 2016) Los eventos tienen por nombre generalmente la acción que se puede realizar en ellos, así por ejemplo en los botones de comando poseen el evento *clic*, los cuadros de texto poseen el evento *keypress*, por mencionar algunos.

El diseño de las aplicaciones se realiza en un objeto que es el contenedor de los demás controles, dicho objeto se conoce como formulario, y una aplicación puede tener tantos formularios como ventanas vaya a tener al final la aplicación.(Universidad de Sevilla, 2015) Por defecto al iniciar el diseño aparece un formulario, en el cual se añadirán todos los controles necesarios, de allí se puede agregar más formularios según sean necesarios. Esto constituirá la interfaz de la aplicación.

Antes de comenzar a diseñar la interfaz de usuario, debe considerar el propósito de la aplicación. El diseño de una aplicación principal que se usa con frecuencia debe ser diferente de una aplicación que se usa solo ocasionalmente durante un corto período de tiempo. Uno de los objetivos principales es que las aplicaciones que proporcionan información tengan requisitos diferentes a las aplicaciones que se utilizan para obtener información.(Cevallos, 2008)

En este tipo de aplicaciones, los usuarios destinatarios, suelen formar parte en el diseño. Las aplicaciones para principiantes requieren un diseño simple, mientras que las aplicaciones para usuarios experimentados pueden requerir aplicaciones más complejas. El diseño de otras aplicaciones utilizadas por los usuarios puede afectar sus expectativas sobre el comportamiento de la aplicación. Si entre los planes está distribuir el producto a nivel mundial, debe considerarse el idioma y la cultura de destino como parte del diseño.

Después de crear una interfaz de aplicación con formularios y controles, deberá escribir código que defina el comportamiento de la aplicación. Como cualquier lenguaje de programación moderno, Visual Basic acepta ciertas estructuras de programación y elementos del lenguaje comunes. Teniendo en cuenta que se trata de un lenguaje orientado a objetos.

Una característica especial de Visual Basic es, que a medida que es escribe interpreta el código, detectando y resaltando la mayoría de los errores

de sintaxis actuales. Es casi como dejar que un experto observe cómo se escribe el código. A diferencia de los lenguajes tradicionales, Visual Basic utiliza un método interactivo para el desarrollo, que difumina la distinción entre los tres pasos (escribir, compilar y verificar el código). Para detectar errores sobre la marcha, Visual Basic también compila parcialmente el código según sea necesario. Cuando esté listo para ejecutar y probar su aplicación, llevará un poco de tiempo completar la compilación. Si el compilador encuentra un error, se resaltará en el código. Esto permite corregir el error y continuar compilando sin tener que empezar de nuevo.

Debido a que Visual Basic es interactivo, se encontrará ejecutando la aplicación con frecuencia cuando esté desarrollando, por lo que puede probar el efecto del código mientras lo escribe sin tener que esperar a que se compile más tarde.

Estructura de una aplicación en Visual Basic.

La estructura de la aplicación es la manera cómo se organizan las instrucciones. Es decir, el lugar donde se almacenan y el orden de ejecución de las instrucciones. Cuando solo se tiene una línea de código, la organización no es muy importante. A medida que las aplicaciones se vuelven cada vez más complejas, la necesidad de organización o estructura se vuelve obvia. Dado que la aplicación de Visual Basic está basada en objetos, la estructura de su código está muy cerca de su representación física en la pantalla, pues por definición, los objetos contienen datos y código.(Cevallos, 2008)

El código de una aplicación de Visual Basic está organizado en una estructura jerárquica. Una aplicación típica consta de uno o más módulos. Cada módulo contiene uno o más procedimientos que a su vez contienen código. Determinar qué programa pertenece a qué módulo dependerá del tipo de aplicación que esté creando. Dado que Visual Basic está basado en objetos, esto puede ayudar a pensar en la aplicación en términos de lo que representa.

Una aplicación simple puede contener un formulario y todo el código de la aplicación reside en el módulo del formulario. A medida que la aplicación crezca y se vuelva más compleja, agregará otros formularios.(García de Jalón et al., 1999) A veces, tendrá un código común que desea ejecutar en varios formularios. No es necesario duplicar el código en ambos formularios, por lo

que será posible o necesario un módulo separado que contiene un proceso para ejecutar código común. Este módulo independiente debe ser un módulo estándar. Con el tiempo, y algo de práctica se puede ir creando bibliotecas de módulos que contengan procesos compartidos.

Las herramientas que se han revisado en este apartado constituyen, constituyen los recursos en lo que ha aplicaciones se refiere que forman parte de las herramientas TIC's que son empleadas a lo largo de la formación de los estudiantes de bachillerato en lo que se refiere a la asignatura de Programación. Las mismas que se enmarcan desde las mas sencillas que permite generar los primeros algoritmos hasta llegar paulatinamente a los lenguajes de programación e interfaces que les permitan desarrollar aplicaciones para ser utilizadas en diferentes propósitos.

CAPÍTULO III: Metodología

3.1 Tipo y diseño de investigación

En la ejecución de este proyecto se ha constatado que los estudiantes de bachillerato presentan dificultades en lograr una resolución adecuada de problemas por medio de algoritmos, a pesar de que a inicio tratan de resolver problemas sencillos y relacionados con la vida cotidiana, son pocos los que logran desarrollar en cierta medida una metodología adecuada que les permita resolver problemas con un grado de dificultad más elevado en los niveles.

En ese contexto, en el presente trabajo se aplica una investigación no experimental, de tipo descriptiva con un enfoque cuantitativo y de diseño transversal, para el efecto se aplicó una investigación de campo para recabar la información a través de encuestas y entrevistas a los involucrados en el proceso de enseñanza de la asignatura de Programación en el nivel de Bachillerato de la Unidad Educativa Urdaneta.

También es palpable el poco uso de herramientas tecnológicas para el desarrollo de su aprendizaje, considerado fundamental en este proceso, ya que se puede lograr un mejor aprendizaje en la práctica y tratado enmarcarse en situaciones reales. Esto se debe en parte a que el único lugar donde pueden desarrollar estas prácticas es en la Institución educativa, lo que les impide desarrollar o profundizar lo aprendido en sus hogares.

El rol que desempeña el maestro es importante, sobre todo para motivar al estudiante en el aprendizaje de esta compleja asignatura, por lo que es fundamental que ante todo tenga un profundo conocimiento de programación, y de ahí debe ser diestro en el manejo de las diferentes herramientas que le puedan servir de apoyo para llegar al estudiantado, despertar y mantener el interés por aprender a desarrollar programas y aplicaciones. Esto implica plantearse como docentes el desafío de reorientar su rol en el aula, empleando estrategias didácticas que resulten innovadoras, administrar el tiempo y el espacio para llevar a cabo proyectos integradores y motivadores (Prado & Lara, 2018).

Investigación no Experimental

(Hernandez Sampieri et al., 2014) señala que la investigación no experimental estudia un determinado fenómeno o situación en el contexto en

el que sucede, sin realizar la manipulación deliberada de las variables, es decir, no es posible tener un control directo, ni tener algún tipo de influencia en las mismas, pues se efectúan análisis sobre hechos que ya se dieron y sus consecuencias.

(Fresno Chavez, 2019) afirma que, en estudios no experimentales u observacionales, no hay intentos de intervenir o cambiar el curso de la situación. Los investigadores se limitan a observar el mismo proceso en grupos con o sin factores a estudiar. Los sujetos seleccionados pueden o no seleccionarse de la población mediante un proceso aleatorio o mediante una selección de muestras no probabilística.

En este sentido se realiza el estudio del aprendizaje de un grupo estudiantes en lo que se refiere a la asignatura de programación, en el que se analiza la incidencia que tiene el uso de herramientas TIC's en su proceso de aprendizaje en los diferentes niveles, por lo que se toma en cuenta varias herramientas en forma general, sin hacer distinción en el número de horas dedicadas, ni la cantidad de herramientas utilizadas, lo que si se verifica es el acceso del que disponen a las herramientas y como afecta aquello en proceso. Luego de observar los resultados se realiza un análisis crítico con los resultados obtenidos.

Investigación Descriptiva

Se enmarca en esta clasificación ya que se busca recabar información sobre el fenómeno que se está estudiando, como es el aprendizaje de lógica de programación en los estudiantes de nivel medio, para realizar una descripción de las implicaciones que tiene y se encaminará a proporcionar una visión de cómo se realiza el proceso y cuáles son los aspectos principales que lo caracterizan, lo cual se puede sustentar con la investigación descriptiva, ya que según (Niño Rojas, 2011) su finalidad es describir la realidad de una situación estudiada, uno de sus aspectos, sus partes, sus categorías, su clasificación o las relaciones que se pueden establecer entre diversos objetos con el fin de esclarecer un hecho, confirmar un enunciado o verificar una hipótesis. Se entiende como un acto de utilizar palabras para representar las características de fenómenos, hechos, situaciones, cosas, persona, etc., para

que todo aquel que tenga la oportunidad de leerla, pueda interpretarla y obtener sus conclusiones.

En este caso lo que se pretende es dar a conocer la realidad del aprendizaje en la asignatura de programación en el nivel de bachillerato, en la que resulta muy dificultoso para la gran mayoría de los estudiantes aprender a desarrollar algoritmos y por lo tanto se vuelve aún más complicado aprender a desarrollar aplicaciones con el uso de herramientas más complejas, ya sea por no utilizar herramientas adecuadas o por no seguir una metodología que resulte efectiva en su proceso de aprendizaje.

Investigación de campo

La investigación de campo o la investigación directa se lleva a cabo precisamente en el momento y lugar donde se da el fenómeno de investigación. (Grajales, 2014) Dado que el objeto de investigación del presente proyecto es con los mismos estudiantes de bachillerato, se lleva a efecto una investigación de campo para obtener los datos directamente de los involucrados en el estudio, con el fin de recabar información que sea confiable y más que nada que refleje verazmente la realidad que viven en su ambiente de formación relacionado con la asignatura de Programación.

Ciertamente, la investigación de campo es aquella en la que se obtienen o miden datos sobre un evento específico en el lugar mismo donde ocurre dicho evento. En otras palabras, la investigación se lleva a cabo en el lugar donde ocurre el fenómeno que se desea estudiar, para recolectar información útil para su investigación. (Cajal Flores, 2020) Esta información puede ser utilizada de dos formas, Por un lado, a través de la investigación de campo, se pueden recopilar datos para ampliar el conocimiento para la investigación. Por otro lado, la información obtenida puede servir para realizar diagnósticos y proponer cambios para corregir cierto tipo de condiciones indeseables.

En lo que respecta concretamente a esta investigación, para constatación del caso en estudio en el que se ven afectados los estudiantes del Bachillerato, dado que las clases son íntegramente virtuales por motivo de la pandemia, se procedió a realizar sesiones virtuales de visitas áulicas, para conocer la realidad en la que reciben las clases de programación. Dichas sesiones fueron

coordinadas con la docente de Programación y también con el Rector del Plantel, para obtener la autorización y poder realizar el proceso investigativo.

Para recabar la información, dado que no se pudo aplicar encuestas en forma presencial, se recurrió al uso de herramientas digitales herramientas, como *Google Forms*, para hacerles llegar las preguntas y puedan ser respondidas desde sus hogares a través del medio que ellos utilicen para comunicarse y para recibir sus clases. Por lo tanto, esta investigación se enmarca en dos caracterizaciones que son:

- De acuerdo al nivel de conocimiento, se trata de una investigación explicativa, pues tiene como objetivo dar respuesta al tema de investigación “El uso de herramientas TIC’s en el aprendizaje de Lógica de Programación de los estudiantes de Bachillerato de la Unidad Educativa Urdaneta, durante el Periodo Lectivo 2020 - 2021”, además, dispone la relación que se da entre las variables, para de esta manera mostrar los aspectos y estructuras que forman parte del proceso.
- De acuerdo al manejo de variables, este trabajo es no experimental, pues se basa principalmente en la observación de fenómenos que ocurren en el medio natural para su posterior análisis. En este tipo de investigación, los sujetos no están expuestos a condiciones o estímulos. Se observa a los sujetos en su entorno natural.

Investigación Cuantitativa

La investigación cuantitativa busca medir las variables, con referencia una determinada magnitud.(Niño Rojas, 2011) Ofrece además, la posibilidad de amplificar los resultados de una manera muchos más general, proporcionando un control sobre los fenómenos de investigación y una perspectiva fundamentada en conteos de las magnitudes.(Hernandez Sampieri et al., 2014) Lo cual es pertinente con este proyecto, ya que se aplican encuestas y los resultados obtenidos son procesados con una herramienta estadística, para luego generar los gráficos necesarios para realizar el análisis respectivo.

Investigación Cualitativa

Por otra parte la investigación cualitativa según (Bernal, 2010) este tipo de investigación tiene como objetivo profundizar en casos concretos y no

generalizar de ningún modo. Su enfoque no está principalmente en la medición, sino en definir y describir los fenómenos sociales a partir de determinadas características, tal y como los mismos elementos perciben en la situación estudiada. En este caso se realiza el análisis del criterio y opinión de la primera autoridad del plantel, así como también de varios docentes que imparten la asignatura de Programación en los diferentes subniveles de bachillerato. Para ello se realizó entrevistas, en las que no se requiere aplicar una medición de tipo numérica, para analizar las respuestas de las preguntas planteadas en la investigación.

3.2 La población y la muestra

3.2.1 Características de la población

Para la realización de esta investigación la población que se ha considerado está compuesta por los estudiantes de los tres subniveles de Bachillerato Técnico de la Unidad Educativa Urdaneta, específicamente de la figura profesional de Informática, quienes presentan dificultades en el aprendizaje de la elaboración de algoritmos, lo cual es un factor determinante para poder desarrollar aplicaciones y programas en las herramientas que se estudian posteriormente, por lo que es necesario encontrar alternativas que ayuden a dar solución a esta situación en un corto y largo plazo, permitiendo de esta manera que los estudiantes desarrollen las habilidades necesarias que les permitan desarrollar programas con facilidad en un entorno que en la actualidad dados los avances y el alcance de la tecnología resulta muy exigente y competitivo.

Por otro lado, también se realiza entrevistas a varios docentes que dictan o están familiarizados con la cátedra de programación y a la autoridad del plantel, con el fin de recabar sus criterios y experiencias sobre la enseñanza de programación en el plantel, para conocer desde su punto de vista los problemas o dificultades a los que se enfrentan en la enseñanza de la asignatura de Programación, y cuáles son las estrategias que se implementan para mejorar de alguna manera el proceso de aprendizaje.

3.2.2 Delimitación de la población

La población seleccionada como objeto de estudio debe estar basado en ciertas características, las mismas que pueden ser de índole demográfica, social, económica, etc. (Arias et al., 2016) situando dicha población claramente definida por características como el lugar, tiempo y contenido (Hernandez Sampieri et al., 2014). De esta manera, al tener una población más específica para la realización del estudio, será más probable realizar la generalización de los resultados encontrados en la investigación.

Para el efecto de esta investigación se cuenta precisamente con una población finita debido a que está delimitada y se tiene definido con exactitud el número de participantes que constituyen el grupo en el cual se realiza el estudio. El total estimado de la población es de ciento veinte estudiantes, distribuidos en los tres subniveles de Bachillerato Técnico, además se ha considerado al director de la institución y dos expertos de la asignatura de programación, con el fin de verificar la validez de la investigación.

Tabla 1

Detalle de la delimitación de la población.

DETALLE	NUMERO	CANTIDAD SELECCIONADA
AUTORIDAD	1	1
EXPERTOS	3	3
ESTUDIANTES	120	54
TOTAL	123	58

Nota: Elaborado por Ing. Walter Chacón Valverde

3.2.3 Tipo de muestra

La muestra o muestreo como también se lo conoce es una herramienta utilizada para delimitar el grupo de estudio, es un subconjunto del universo o una parte representativa del total de la población. (Hernández & Carpio, 2019) En este caso dada la naturaleza de la investigación para la selección de la muestra se va a utilizar el método probabilístico, el mismo que ofrece dos técnicas que se pueden que se pueden aplicar para realizar la selección de la muestra. Está el muestreo aleatorio simple, que consiste en asignar un número a cada uno de los integrantes que conforman la población y luego a través de un método al azar escoger a cada individuo hasta completar la muestra; al usar

esta técnica es importante conocer todos los elementos que forman el grupo. Por otro lado, está el muestro aleatorio sistemático, en el cual se requiere conocer la población para luego aplicar una numeración a todos los elementos que van a formar parte de la muestra.

El grupo seleccionado, desde el cual se obtiene la muestra cumplen con la característica común de presentar dificultades en el aprendizaje de la elaboración de algoritmos y programas para computadoras.

3.2.4 Tamaño de la muestra

Algunas de las ventajas de la selección de una muestra de población radican en que permite realizar el estudio en menor tiempo, se reducen significativamente los gastos, posibilita un análisis profundo de las variables que forman parte del estudio y permite un mayor control sobre las mismas.(Fresno Chavez, 2019)

El cálculo del tamaño de la muestra es uno de los aspectos a concretar en las fases previas de la investigación el cual determina el grado de credibilidad que se concederá a los resultados obtenidos. Una fórmula muy extendida que orienta sobre el cálculo del tamaño de la muestra para datos globales es la siguiente, cuando la población es finita y se conoce con certeza su tamaño.

$$n = \frac{k^2 * PQ * N}{(e^2 * (N - 1)) + k^2 * PQ}$$

Valores			
k	PQ	N	E
1,96	0,25	120	0,05

$$n = \frac{1,96^2 * 0,25 * 120}{(0,10^2 * (120 - 1)) + 1,96^2 * 0,25}$$

$$n = \frac{3,8416 * 0,25 * 120}{0,01 * (119) + 3,8416 * 0,25}$$

$$n = \frac{115,248}{1,19 + 0,9604}$$

$$n = \frac{115,248}{2,1504}$$

$$n = 53,59$$

Tabla 2

Descripción de los elementos que conforman la fórmula para calcular la muestra

Descripción	
n	Es el tamaño de la muestra (número de encuestas que se va a realizar).
PQ	Constante de varianza de la población.
N	Es el tamaño de la población o universo, es decir el número total de encuestados. Se ha tomado como población a los 120 estudiantes de Bachillerato Técnico de la Unidad Educativa "Urdaneta"
e	Es el error muestral deseado. El error muestral es la diferencia que puede haber entre el resultado que se obtiene preguntando a una muestra de la población y el que se obtendría si preguntáramos al total de ella. Para el presente cálculo se ha tomado un margen de error del 10%.
k	Es una constante que depende del nivel de confianza que se asigne. El nivel de confianza indica la probabilidad de que los resultados de la investigación sean ciertos: un 95,5 % de confianza es lo mismo que decir que puede haber equivocación con una probabilidad del 4,5%. Para este cálculo se ha tomado el valor de 1,96 y un nivel de confianza del 95%.

Nota: Nota: Elaborado por Ing. Walter Chacón Valverde

El tamaño de la muestra que se ha considerado para la realización del presente estudio corresponde a 54 del total de 120 estudiantes que se encuentran matriculados en los diferentes subniveles de Bachillerato Técnico.

3.2.5 Proceso de selección de la muestra

El proceso de selección se lo realizó de forma probabilística, ya que se ha tomado en cuenta una muestra significativa de la población total que conforman el bachillerato Técnico en la especialidad de informática de la Unidad Educativa Urdaneta, lo que permite realizar correctivos que sean necesarios en las estrategias y metodologías empleadas en el aprendizaje de lógica de programación. Aunque las dos técnicas mencionadas anteriormente pueden aplicarse sin problemas a la población que se está estudiando en el presente proyecto, para tener una mayor confiabilidad de los resultados se ha considerado aplicar la técnica del muestreo aleatorio simple, en el que se ha considerado también el grado de compromiso y responsabilidad de los estudiantes, para obtener una muestra de 54 personas, para realizar el estudio.

3.3 Los métodos y las técnicas

3.3.1 Métodos

En la estructuración del presente proyecto de investigación se emplearon los siguientes métodos de investigación:

Método Analítico. Se empleó este método debido a que se ha descompuesto el proyecto en sus partes o componentes, de manera que se pueda observar la naturaleza, así como las causas y sus efectos. A través de este método se pretende conocer como la aplicación adecuada del modelo pedagógico del aprendizaje basado en proyectos, proporciona significativos beneficios a los estudiantes, ya que les facilita el trabajo colaborativo y en equipos.

Método inductivo. Este método forma parte de esta investigación por el hecho de que permite realizar un proceso de sistematización, en el que se parte del análisis particular de los resultados obtenidos de cada estudiante para encontrar una relación que permita explicar el fenómeno del aprendizaje de programación en los estudiantes, pero en una forma generalizada.

Método Deductivo. Lo que se ha realizado en este proyecto constituye una actividad científica, y de hecho es de lo que se trata el método deductivo. Ya que en la ejecución se aplicaron algunos de los pasos esenciales que forman parte del método científico, entre los que están la observación del fenómeno, aun cuando se trataron de clases virtuales, también se definió una hipótesis, para explicar el fenómeno estudiado, se dedujeron las consecuencias derivadas de las hipótesis, para finalmente comprobar la veracidad de los resultados.

Método sintético. Con la utilización de este método, fue posible realizar una reconstrucción del fenómeno estudiado, integrando todos los componentes del proyecto en un todo para ser analizados en su totalidad. Obteniendo una síntesis que nos permita entender la correlación entre las diferentes variables, y poderlas estudiar en forma holística.

Método de observación científica. Constituye un método empírico en este caso, ya que permite observar directamente la realidad en la que se desenvuelve el estudiante en su proceso de aprendizaje, pero además posee

cualidades como la sistematización, objetividad y los mecanismos de control necesarios para evitar errores que pueden generar confusión o ambigüedad.

Para llegar a obtener las conclusiones en primer lugar se realizó la observación de las aulas, con el fin observar cuales son las herramientas utilizadas para el aprendizaje de lógica de programación, en la que se concluyó que, si bien hacen uso de varias herramientas, existen otras herramientas que ayudaría significativamente en el aprendizaje. Se dedujo de esto que existe poca motivación en los estudiantes en el aprendizaje de la asignatura de programación y que, si bien al principio existe un gran interés, el mismo disminuye paulatinamente en los niveles subsiguientes. De los cual se concluyó que al final, los estudiantes se vuelven poco investigativos y también carentes de las habilidades que les permita desarrollar programas y aplicaciones de computación y proceso de información.

3.3.2 Técnicas

Las técnicas utilizadas para el desarrollo del presente trabajo de investigación fueron la encuesta y la entrevista.

Técnica: Encuesta

Instrumento: Cuestionario

Se empleó esta técnica para recabar la información directamente de los sujetos que forman parte del objeto de estudio, en este caso a los estudiantes que forman parte de la muestra seleccionada. Se aplicó un cuestionario de dieciséis preguntas, con respuestas de selección múltiple, utilizando en todas las preguntas el modelo de respuesta de la escala de Likert.

Este cuestionario se aplicó por medio de internet, diseñando las preguntas en la aplicación *Google Forms*, compartida a los 54 estudiantes que componen la muestra en cuestión.

Técnica: Entrevista.

Instrumento: Cuestionario

Esta técnica nos permite recabar información directamente de las personas consideradas dentro del dimensionamiento de la muestra en este proyecto, en este caso del director de la Institución y tres expertos. Para ellos emplea seis preguntas estructuradas, planteadas previamente y formuladas a todos en los mismos términos y orden.

Aunque la entrevista se realizó directamente, fue necesario utilizar medios virtuales, para efectuar dichas entrevistas, utilizando para ello la aplicación *Zoom*, esta manera tanto el director como los expertos pudieron dar a conocer sus posturas y realidad en los que respecta a la enseñanza de programación a los estudiantes de Bachillerato Técnico.

3.4 Propuesta de procesamiento estadístico de la información.

La información fue recogida en modo estandarizado, es decir aplicando cuestionarios iguales a todos los involucrados, luego los datos obtenidos fueron procesados por medios electrónicos, informáticos y estadísticos; a través de estos medios, se almacenan, procesan y tabulan todos los datos que se obtuvieron mediante las encuestas y entrevistas aplicadas.

Una vez tabulados los datos se ingresan en la herramienta informática SPSS (“Statistical Package for the Social Sciences”), a través de la cual se podrán generar las tablas de datos y su representación gráfica desde dichas tablas, lo que servirá para establecer análisis en forma particular primeramente y en base a esto obtener puntos de vista más generales.

Finalmente, esta información se le dará a conocer a través de gráficos de barras y la distribución de frecuencia de las variables con proporciones nominales y progresivas. De esta manera se determinará, el diagnóstico de cada uno de los componentes se determinará mediante números absolutos o porcentuales para fines de comparación.

CAPÍTULO IV: Análisis e interpretación de resultados

4.1 Análisis de Descriptivo de los resultados

Para el desarrollo del presente trabajo se diseñó un total de dieciséis preguntas para los estudiantes y un modelo de entrevista para el Rector de la Unidad Educativa Urdaneta. Las preguntas generadas para los estudiantes fueron sometidas al criterio de tres expertos para verificar su validez, y por otro lado se aplicó la técnica de Alpha de Cronbach, para comprobar los resultados obtenidos en la encuesta, antes de realizar la tabulación de resultados.

La finalidad del trabajo de investigación fue determinar cómo inciden las herramientas TIC's en el aprendizaje de lógica de programación en los estudiantes de bachillerato de la Unidad Educativa Urdaneta, del cantón Urdaneta.

La información tabulada permite graficar y analizar las variables que están influyendo en el proceso de aprendizaje de lógica de programación, con el fin de tomar decisiones.

Luego realizar el análisis e interpretar cada una de las preguntas que formaron parte de la encuesta se puede apreciar que la mayoría de los estudiantes tiene una mejor predisposición al uso de las herramientas TIC's en su proceso de aprendizaje, para lo cual como es importante para ellos contar con las herramientas tecnológicas que les permita aprovechar las ventajas que ello ofrece.

En los resultados se puede observar también que les favorece mucho la diversidad de las herramientas que se aplican a lo largo del proceso, ya que en este sentido puede iniciar aprendiendo algoritmos sencillos, que les permita introducirse gradualmente en la programación, para luego ir explorando herramientas cada vez con un mayor grado de complejidad con las cuales serán capaces de desarrollar diferentes tipos de aplicación que sean de utilidad en el ámbito en el cual se desarrollan.

Por último, las metodologías de aprendizaje basado en proyectos y aprendizaje basado en problemas, es significativo, en el sentido de que a través de ellos pueden alcanzar un aprendizaje mas efectivo, ya que tienen una mayor interacción con sus pares, logrando aprender en forma colaborativa.

Encuesta dirigida a los estudiantes de Bachillerato de la Unidad Educativa Urdaneta.

1. ¿Es importante el uso de la herramienta Pselnt para la resolución de algoritmos?

Tabla 3

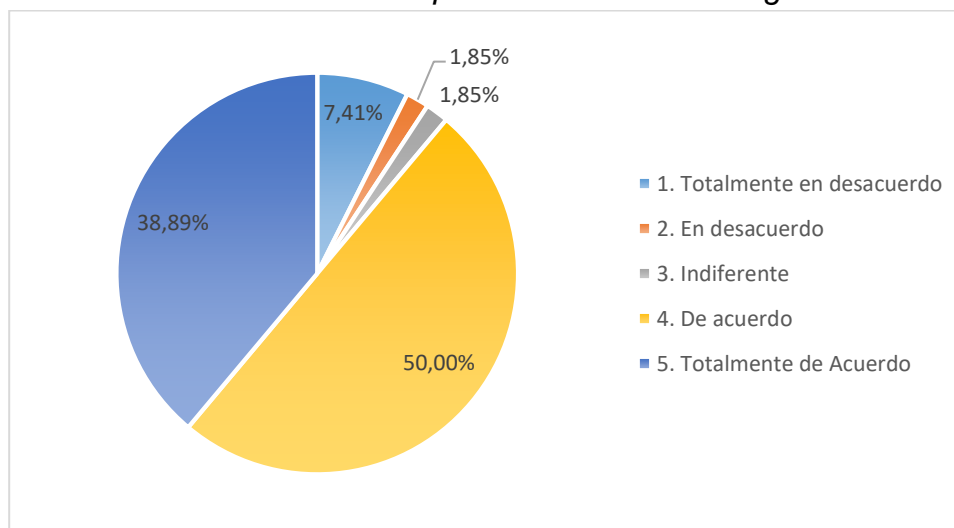
Uso de la herramienta Pselnt para la resolución de algoritmos

Alternativas	Frecuencia	Porcentaje
1. Totalmente en desacuerdo	4	7,41%
2. En desacuerdo	1	1,85%
3. Indiferente	1	1,85%
4. De acuerdo	27	50,00%
5. Totalmente de Acuerdo	21	38,89%
Total	54	100,00%

Nota: Elaborada mediante la herramienta estadística SPSS

Gráfico estadístico 1

Uso de la herramienta Pselnt para la resolución de algoritmos



Nota: Gráfico generado mediante la aplicación SPSS.

ANÁLISIS E INTERPRETACIÓN: En la pregunta 1, de los 54 estudiantes encuestados, el 50,00% manifiesta que está de acuerdo en la importancia de la herramienta Pselnt para la elaboración de algoritmos; 38,89% manifiesta que está totalmente de acuerdo; 7,41% manifiesta estar totalmente en desacuerdo; 1,85% manifiesta estar en desacuerdo y finalmente al 1,85% restante le es indiferente. Los estudiantes muestran una clara preferencia del uso de la herramienta Pselnt en la resolución de algoritmos, esto puede ser debido a que es la primera herramienta que utilizan para el aprendizaje de programación, a estar incluida en los primeros módulos de la asignatura.

2. ¿La herramienta Pselnt es adecuada para representar los algoritmos mediante diagramas de flujo?

Tabla 4

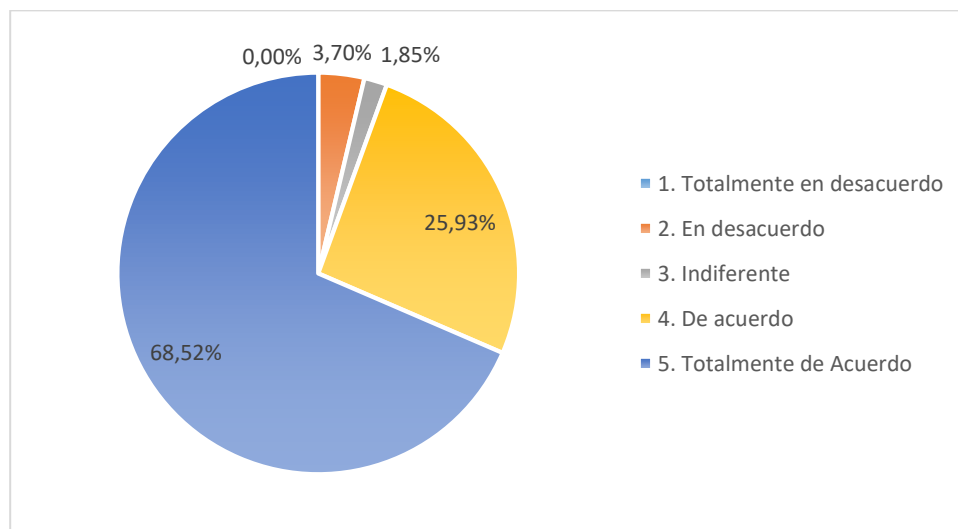
La herramienta Pselnt y su uso para representar los algoritmos mediante diagramas de flujo

Alternativas	Frecuencia	Porcentaje
1. Totalmente en desacuerdo	1	1,85%
2. En desacuerdo	0	0,00%
3. Indiferente	4	7,41%
4. De acuerdo	35	64,81%
5. Totalmente de Acuerdo	14	25,93%
Total	54	100,00%

Nota: Elaborada mediante la herramienta estadística SPSS

Gráfico estadístico 2

La herramienta Pselnt como herramienta para representar diagramas de flujo



Nota: Gráfico generado mediante la aplicación SPSS.

ANÁLISIS E INTERPRETACIÓN: En la pregunta 2, de los 54 estudiantes encuestados, el 51,85% concuerda que la herramienta Pselnt, es la más adecuada para representar algoritmos; 33,33% manifiestan estar totalmente de acuerdo; 9,26% manifiesta que le es indiferente y finalmente al 5,56% señala estar en desacuerdo. La diversidad de formas de representar un algoritmo tanto en diagrama de flujo como en pseudocódigo en la misma herramienta es un factor fundamental para que los alumnos prefieran hacer uso de esta, al momento de generar los algoritmos para resolver un problema determinado.

3. ¿El lenguaje de programación C++ presenta una estructura sencilla que brinda facilidad para escribir sus primeros programas?

Tabla 5

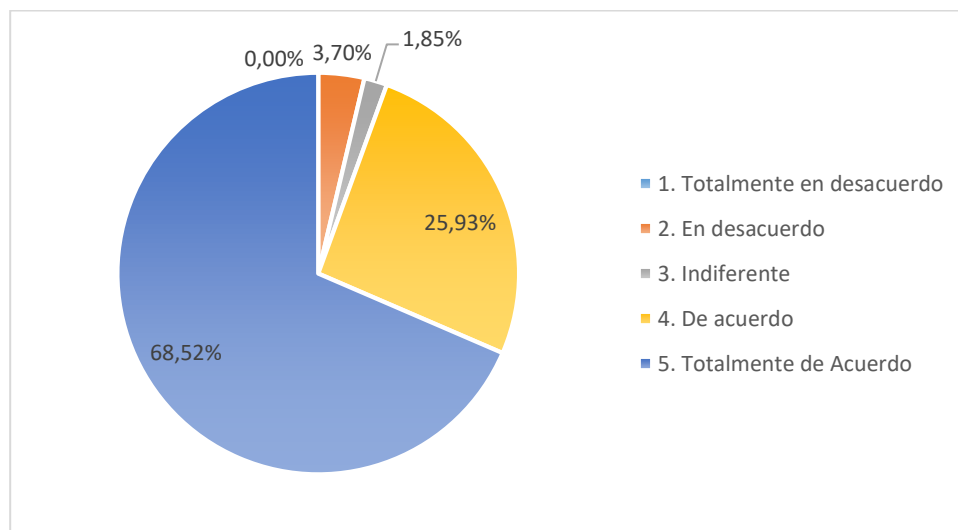
El lenguaje de programación C++ y su sencillez para escribir los primeros programas.

Alternativas	Frecuencia	Porcentaje
1. Totalmente en desacuerdo	0	0,00%
2. En desacuerdo	3	5,56%
3. Indiferente	5	9,26%
4. De acuerdo	28	51,85%
5. Totalmente de Acuerdo	18	33,33%
Total	54	100,00%

Nota: Elaborada mediante la herramienta estadística SPSS

Gráfico estadístico 3

El lenguaje de programación C++ y su estructura para escribir los primeros programas



Nota: Gráfico generado mediante la aplicación SPSS.

Análisis e Interpretación: En la pregunta 3, de los 54 estudiantes encuestados, el 51,85% está de acuerdo con que el lenguaje de programación C++, presentan una sencilla estructura para escribir los primeros programas; 33,33% manifiestan estar totalmente de acuerdo; 9,26% manifiesta que le es indiferente y finalmente el 5,56% se manifiesta en desacuerdo. El Lenguaje de programación C++ es el primero que deben aprender y se puede apreciar con claridad que los estudiantes se manifiestan que pueden comprender fácilmente la estructura de un programa a través de este lenguaje.

4. ¿El lenguaje de programación Python ofrece flexibilidad al momento de codificar los programas?

Tabla 6

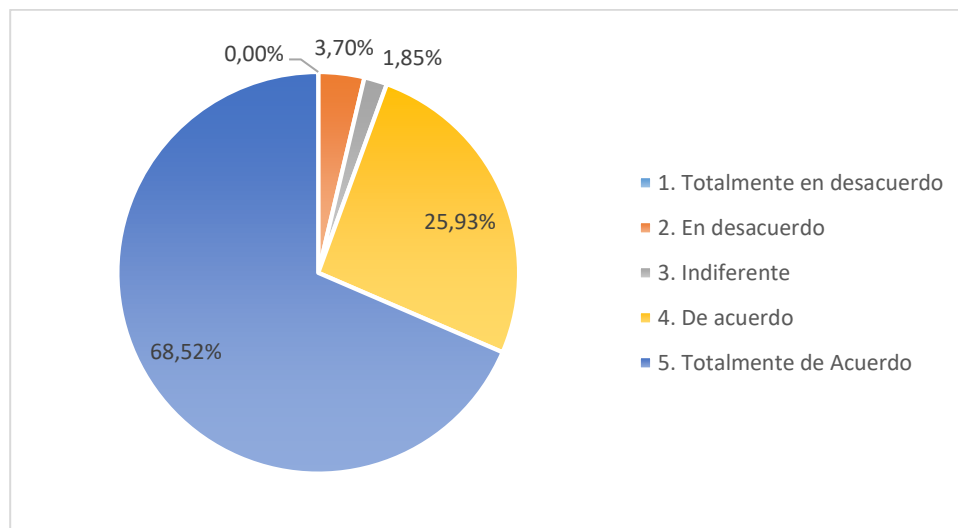
La flexibilidad del lenguaje de programación Python al momento de codificar los programas

Alternativas	Frecuencia	Porcentaje
1. Totalmente en desacuerdo	1	1,85%
2. En desacuerdo	4	7,44%
3. Indiferente	8	14,81%
4. De acuerdo	33	61,11%
5. Totalmente de Acuerdo	8	14,81%
Total	54	100,00%

Nota: Elaborada mediante la herramienta estadística SPSS

Gráfico estadístico 4

La flexibilidad del lenguaje de programación Python al momento de codificar los programas.



Nota: Gráfico generado mediante la aplicación SPSS.

Análisis e Interpretación: En la pregunta 4, de los 54 estudiantes encuestados, el 61,11% está de acuerdo en que el lenguaje de programación Python ofrece flexibilidad al momento de codificar los programas; 14,81% manifiestan estar totalmente de acuerdo; otro 14,81% manifiesta que le es indiferente; 7,41% se manifiesta en desacuerdo y el restante 1,85% manifiesta estar totalmente en desacuerdo. Del lenguaje de programación Python les resulta muy atractiva a los estudiantes la flexibilidad que ofrece al momento de generar el código necesario para desarrollar los programas.

5. ¿El ambiente grafico de la herramienta Scratch permite entender con facilidad la estructura de un programa?

Tabla 7

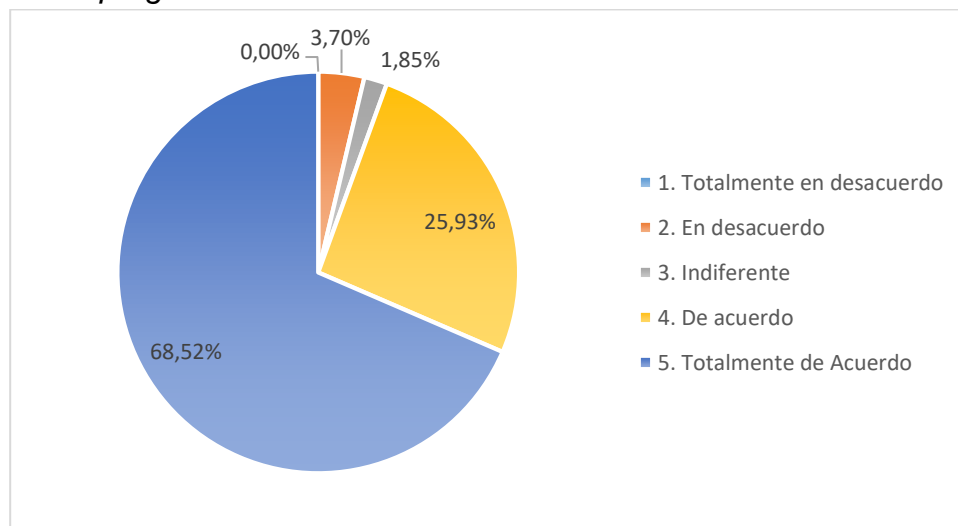
El ambiente gráfico de la herramienta Scratch como factor para entender la estructura de un programa.

Alternativas	Frecuencia	Porcentaje
1. Totalmente en desacuerdo	1	1,85%
2. En desacuerdo	3	5,56%
3. Indiferente	6	11,11%
4. De acuerdo	30	55,56%
5. Totalmente de Acuerdo	14	25,93%
Total	54	100,00%

Nota: Elaborada mediante la herramienta estadística SPSS

Gráfico estadístico 5

El ambiente gráfico de Scratch como factor para entender la estructura de un programa.



Nota: Gráfico generado mediante la aplicación SPSS.

Análisis e Interpretación: En la pregunta 5, de los 54 estudiantes encuestados, el 55,56% está de acuerdo con que el ambiente grafico de la herramienta Scratch si permite entender con facilidad la estructura de un programa; 25,93% manifiestan estar totalmente de acuerdo; el 11,11% manifiesta que le es indiferente; 5,56% se manifiesta en desacuerdo y el restante 1,85% manifiesta estar totalmente en desacuerdo. Los estudiantes prefieren usar Scratch, precisamente por el ambiente visual y gráfico que ofrece al momento de generar programas, haciendo que dicha tarea resulte mucho más sencilla que en los lenguajes de código estructurado.

6. ¿El diseño de aplicaciones es más sencillo utilizando la interfaz de controles y formularios de la herramienta VB.Net?

Tabla 8

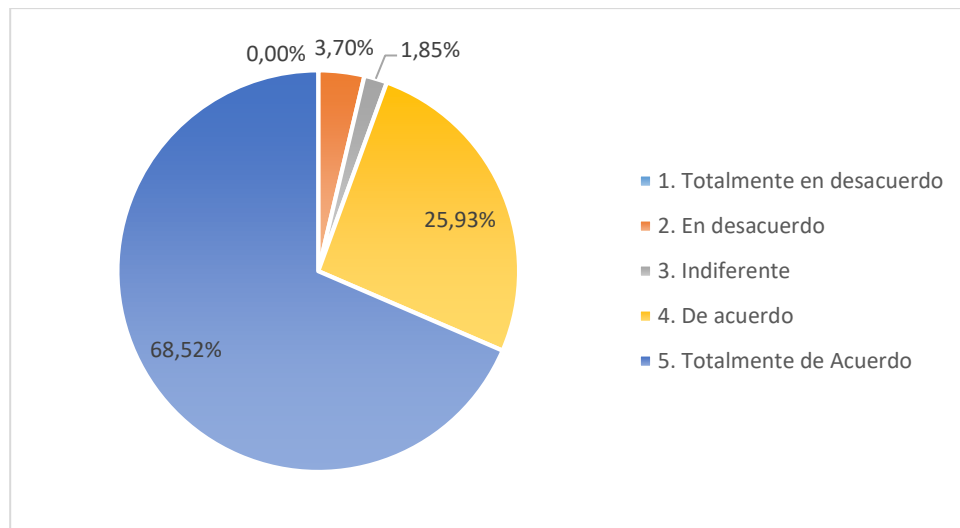
El diseño de aplicaciones mediante la interfaz de controles y formularios de VB.Net

Alternativas	Frecuencia	Porcentaje
1. Totalmente en desacuerdo	2	3,70%
2. En desacuerdo	7	12,96%
3. Indiferente	7	12,9%
4. De acuerdo	25	46,30%
5. Totalmente de Acuerdo	13	24,07%
Total	54	100,00%

Nota: Elaborada mediante la herramienta estadística SPSS

Gráfico estadístico 6

El diseño de aplicaciones utilizando la interfaz de controles y formularios de la herramienta VB.net



Nota: Gráfico generado mediante la aplicación SPSS.

Análisis e Interpretación: En la pregunta 6, de los 54 estudiantes encuestados, el 46,30% está de acuerdo en que el diseño de aplicaciones es más sencillo utilizando la interfaz de controles y formularios de la herramienta VB.Net; 24,07% manifiestan estar totalmente de acuerdo; el 12,96% indica que le es indiferente; 12,96% se manifiesta en desacuerdo y el restante 3,70% manifiesta estar totalmente en desacuerdo. Dentro del contenido de la asignatura, VB.Net, es una de las herramientas preferidas, por ser la más avanzada y la que se aproxima al diseño de aplicaciones ejecutables para Windows, esto añadido al ambiente visual y la poca necesidad escribir código.

7. ¿Disponer de acceso a internet es esencial para el aprendizaje de la asignatura de programación?

Tabla 9

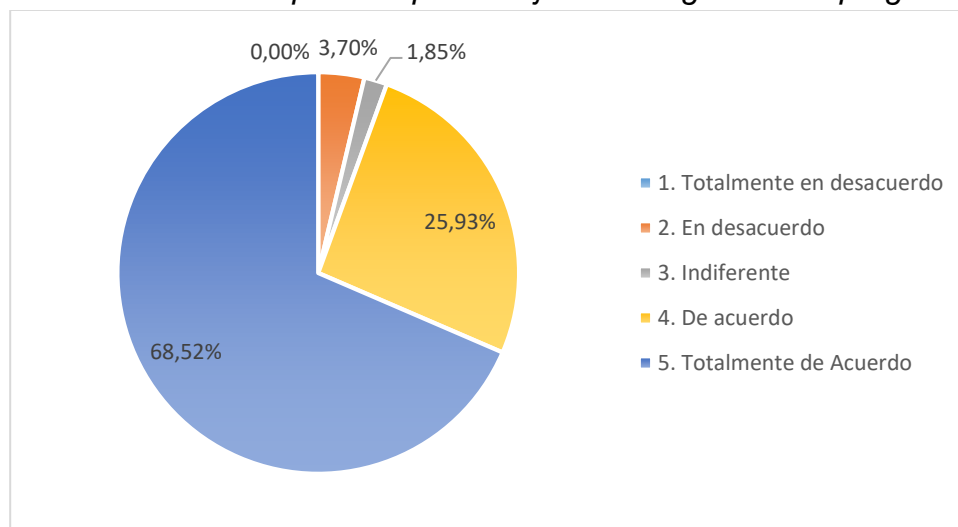
El acceso a internet, herramienta para el aprendizaje de programación

Alternativas	Frecuencia	Porcentaje
1. Totalmente en desacuerdo	2	2,70%
2. En desacuerdo	1	1,85%
3. Indiferente	4	7,44%
4. De acuerdo	19	35,19%
5. Totalmente de Acuerdo	28	51,85%
Total	54	100,00%

Nota: Elaborada mediante la herramienta estadística SPSS

Gráfico estadístico 7

El acceso a internet para el aprendizaje de la asignatura de programación.



Nota: Gráfico generado mediante la aplicación SPSS.

Análisis e Interpretación: En la pregunta 7, de los 54 estudiantes encuestados, el 51,85% manifiesta estar totalmente de acuerdo en que disponer de acceso a internet es esencial para el aprendizaje de la asignatura de programación; 35,19% manifiestan estar de acuerdo; el 7,44% indica que le es indiferente; 3,70% señala estar totalmente en desacuerdo y el restante 1,85% manifiesta estar en desacuerdo. El uso de la Internet se vuelve fundamental en el aprendizaje programación, debido a que mediante esta herramienta puede acceder a una gran cantidad de recursos que facilitan el aprendizaje de programación, tales como tutoriales, ejemplos, herramientas y los diferentes lenguajes de programación, para ser descargados e instalados en sus dispositivos.

8. **¿Contar con las herramientas tecnológicas necesarias en la institución educativa o en el hogar permite al estudiante alcanzar un mejor nivel aprendizaje?**

Tabla 10

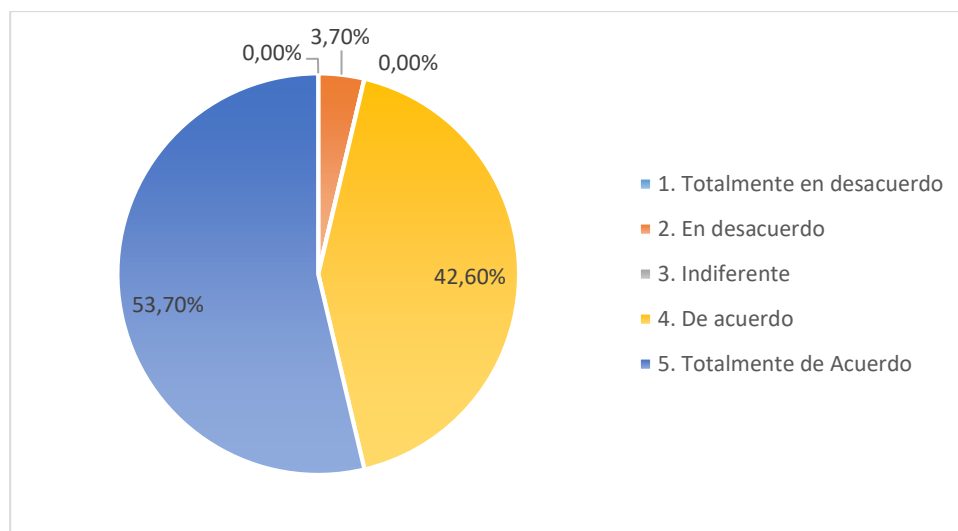
Disponibilidad de las herramientas tecnológicas en la institución educativa o en el hogar para alcanzar un mejor nivel de aprendizaje

Alternativas	Frecuencia	Porcentaje
1. Totalmente en desacuerdo	0	0,00%
2. En desacuerdo	2	3,70%
3. Indiferente	0	0,00%
4. De acuerdo	23	42,60%
5. Totalmente de Acuerdo	29	53,70%
Total	54	100,00%

Nota: Elaborada mediante la herramienta estadística SPSS

Gráfico estadístico 8

Disponibilidad de las herramientas tecnológicas



Nota: Gráfico generado mediante la aplicación SPSS.

Análisis e Interpretación: En la pregunta 8, de los 54 estudiantes encuestados, el 53,70% manifiesta estar totalmente de acuerdo en que contar con las herramientas tecnológicas necesarias en la institución educativa o en el hogar permite al estudiante alcanzar un mejor nivel aprendizaje; 42,59% manifiestan estar de acuerdo; y el restante 3,70% manifiesta estar en desacuerdo. En la asignatura de programación resulta esencial contar con las herramientas y aplicaciones necesarias para que el estudiante pueda desarrollar las prácticas que le posibiliten alcanzar un aprendizaje significativo.

9. ¿La realización de actividades grupales y/o colaborativas resultan efectivas para el aprendizaje en la elaboración programas?

Tabla 11

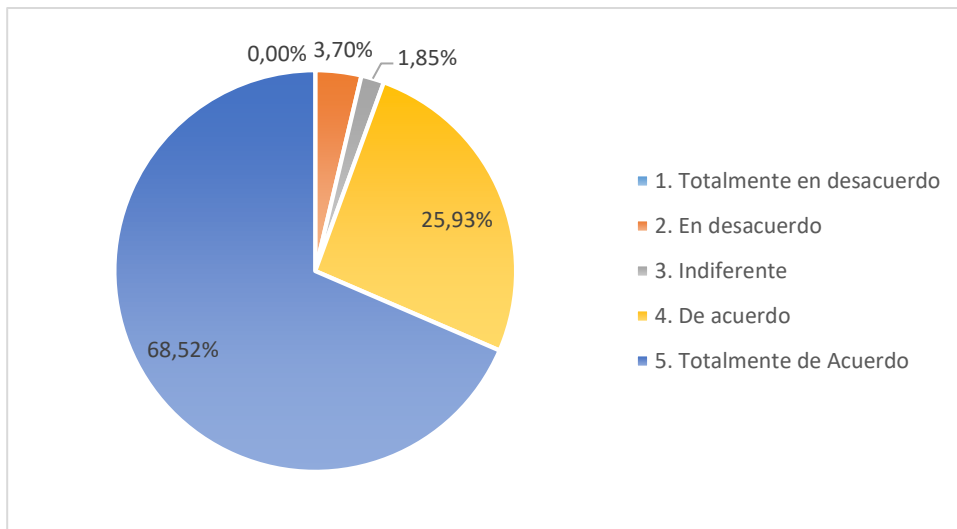
Las actividades grupales y/o colaborativas en el aprendizaje de la elaboración de programas.

Alternativas	Frecuencia	Porcentaje
1. Totalmente en desacuerdo	1	1,85%
2. En desacuerdo	3	5,56%
3. Indiferente	5	9,26%
4. De acuerdo	24	44,44%
5. Totalmente de Acuerdo	21	38,89%
Total	54	100,00%

Nota: Elaborada mediante la herramienta estadística SPSS

Gráfico estadístico 9

Las actividades grupales y/o colaborativas en el aprendizaje de programación



Nota: Gráfico generado mediante la aplicación SPSS.

Análisis e Interpretación: En la pregunta 9, de los 54 estudiantes encuestados, el 44,44% están de acuerdo en que la realización de actividades grupales y/o colaborativas resultan efectivas para el aprendizaje en la elaboración programas; 38,89% manifiestan estar totalmente de acuerdo; el 9,26% indica que le es indiferente; 5,56% se manifiesta en desacuerdo y el restante 1,85% manifiesta estar totalmente en desacuerdo. Los estudiantes simpatizan con el aprendizaje grupal y colaborativo, en razón que mejora la interacción entre ellos y el docente y les favorece la retención académica, etc.

10. ¿Es importante que las actividades propuestas para desarrollar algoritmos estén basadas en un contexto real?

Tabla 12

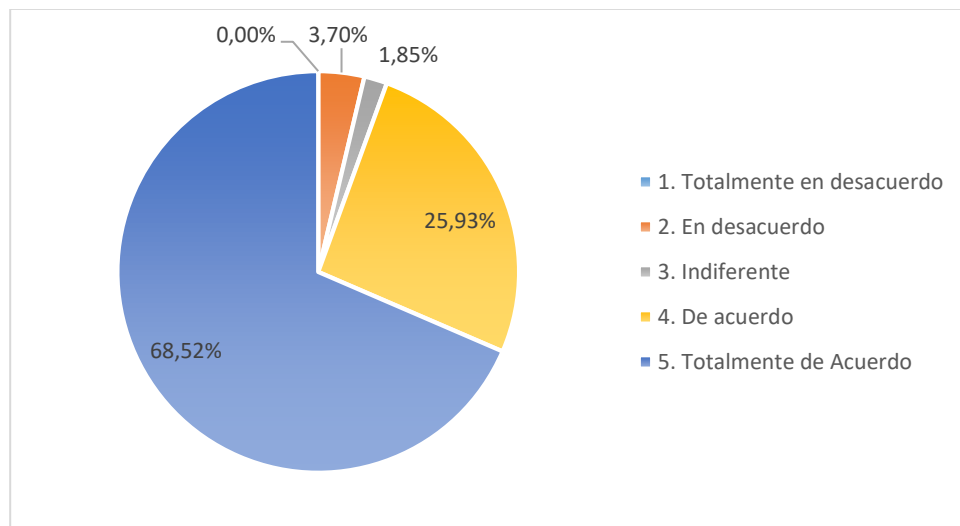
Las actividades propuestas para desarrollar algoritmos basadas en un contexto real.

Alternativas	Frecuencia	Porcentaje
1. Totalmente en desacuerdo	1	1,85%
2. En desacuerdo	1	1,85%
3. Indiferente	7	12,96%
4. De acuerdo	27	50,00%
5. Totalmente de Acuerdo	18	33,33%
Total	54	100,00%

Nota: Elaborada mediante la herramienta estadística SPSS

Gráfico estadístico 10

Actividades para desarrollar algoritmos basados en un contexto real.



Nota: Gráfico generado mediante la aplicación SPSS.

Análisis e Interpretación: En la pregunta 10, de los 54 estudiantes encuestados, el 50,00% está de acuerdo en que es importante que las actividades propuestas para desarrollar algoritmos estén basadas en un contexto real; 33,33% manifiestan estar totalmente de acuerdo; el 12,96% indica que le es indiferente; 1,85% se manifiesta en desacuerdo y el restante 1,85% manifiesta estar totalmente en desacuerdo. Esto es muy importante para alcanzar un aprendizaje significativo, ya que resulta motivante para los estudiantes que lo aprendido en el aula lo puedan llevar a la práctica en la realidad en la que se desenvuelven.

11. ¿La metodología basada en proyectos es fundamental para lograr un adecuado desarrollo de los programas?

Tabla 13

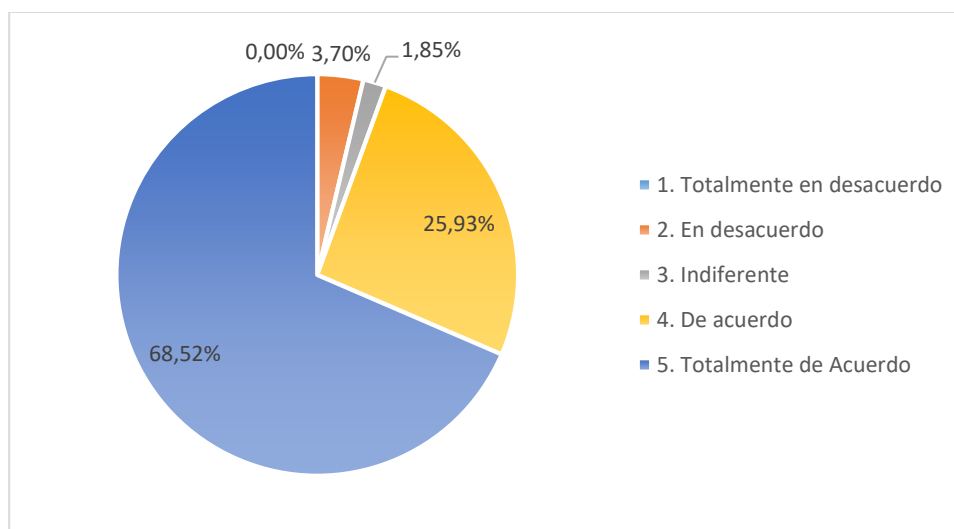
La metodología basada en proyectos para lograr un adecuado desarrollo de los programas de computadora

Alternativas	Frecuencia	Porcentaje
1. Totalmente en desacuerdo	0	0,00%
2. En desacuerdo	4	7,41%
3. Indiferente	4	7,41%
4. De acuerdo	33	61,11%
5. Totalmente de Acuerdo	13	24,07%
Total	54	100,00%

Nota: Elaborada mediante la herramienta estadística SPSS

Gráfico estadístico 11

La metodología basada en proyectos para el desarrollo de programas



Nota: Gráfico generado mediante la aplicación SPSS.

Análisis e Interpretación: En la pregunta 11, de los 54 estudiantes encuestados, el 61,11% está de acuerdo que la metodología basada en proyectos es fundamental para lograr un adecuado desarrollo de los programas; 24,07% manifiestan estar totalmente de acuerdo; el 7,41% indica que le es indiferente y el restante 7,41% manifiesta estar en desacuerdo. La metodología basada en proyectos es esencial en la asignatura de programación, debido a que la concepción de un programa, parte desde un proyecto en sí donde se debe tener en cuenta etapas o procesos como el análisis, planificación y ejecución para llegar a desarrollar un programa.

12. ¿Es necesario el uso frecuente de herramientas TIC's para mejorar la destreza en el proceso de elaboración de programas?

Tabla 14

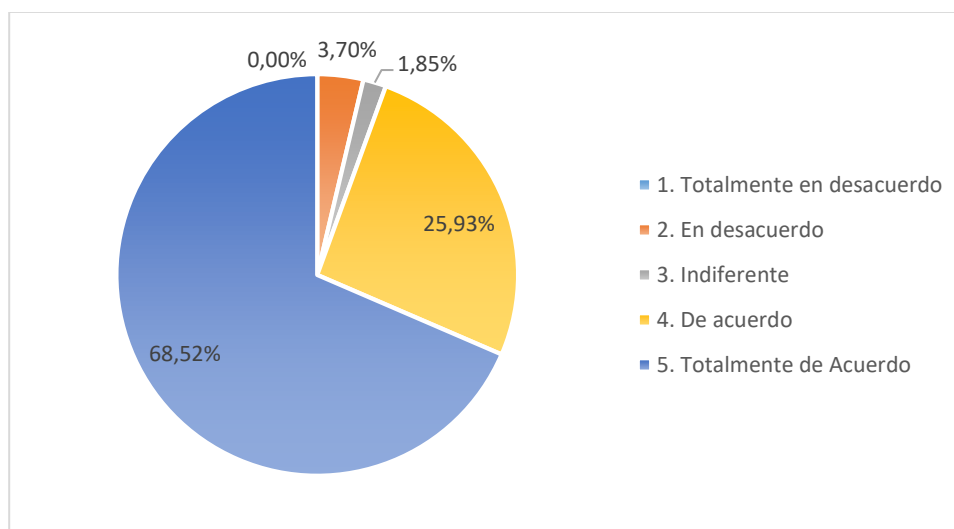
El uso frecuente de herramientas TIC's para mejorar la destreza en la elaboración de programas.

Alternativas	Frecuencia	Porcentaje
1. Totalmente en desacuerdo	2	3,70%
2. En desacuerdo	1	1,85%
3. Indiferente	9	16,67%
4. De acuerdo	31	57,41%
5. Totalmente de Acuerdo	11	20,37%
Total	54	100,00%

Nota: Elaborada mediante la herramienta estadística SPSS

Gráfico estadístico 12

El uso de las herramientas TIC's para la elaboración de programas.



Nota: Gráfico generado mediante la aplicación SPSS.

Análisis e Interpretación: En la pregunta 12, de los 54 estudiantes encuestados, el 57,41% está de acuerdo en que es necesario el uso frecuente de herramientas TIC's para mejorar la destreza en el proceso de elaboración de programas; 20,37% manifiestan estar totalmente de acuerdo; el 16,67% indica que le es indiferente; 3,70% señalan que están totalmente en desacuerdo y el restante 1,85% manifiestan que están en desacuerdo. Un aspecto fundamental para mejorar la destreza al desarrollar programas es la práctica frecuente en las herramientas o lenguajes de programación, de esta manera podrá apropiarse las habilidades necesarias para esta tarea.

13. ¿El uso de algoritmos es el método más adecuado para resolver determinados problemas cotidianos?

Tabla 15

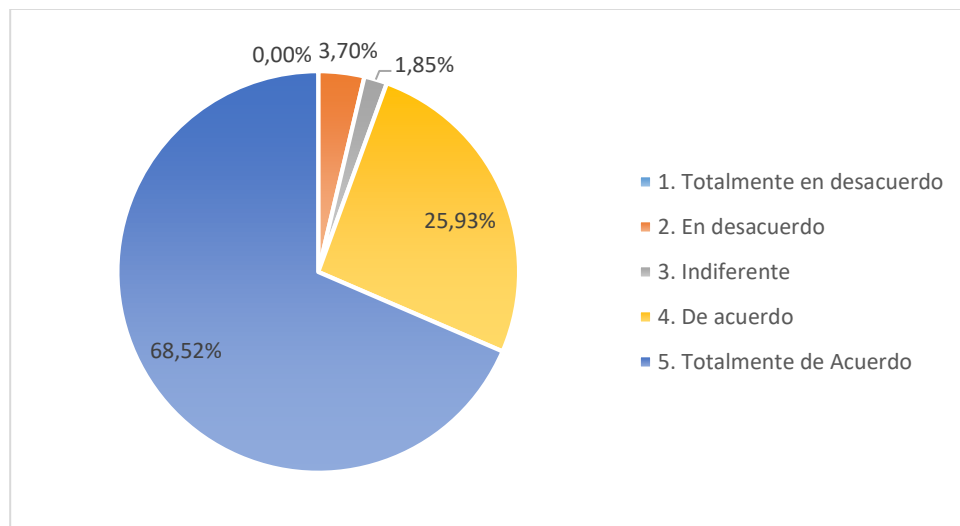
El uso de algoritmos como el método más adecuado para resolver determinados problemas cotidianos

Alternativas	Frecuencia	Porcentaje
1. Totalmente en desacuerdo	0	0,00%
2. En desacuerdo	6	11,11%
3. Indiferente	6	11,11%
4. De acuerdo	20	37,04%
5. Totalmente de Acuerdo	22	40,74%
Total	54	100,00%

Nota: Elaborada mediante la herramienta estadística SPSS

Gráfico estadístico 13

El uso de algoritmos en la resolución de problemas cotidianos



Nota: Gráfico generado mediante la aplicación SPSS.

Análisis e Interpretación: En la pregunta 13, de los 54 estudiantes encuestados, el 40,74% está totalmente de acuerdo en que el uso de algoritmos es el método más adecuado para resolver determinados problemas cotidianos; 37,04% manifiestan estar de acuerdo; el 11,11% indica que le es indiferente y el restante 11,11% señala que están en desacuerdo. Se puede notar que los estudiantes aprecian con claridad las ventajas de utilizar algoritmos en la resolución de problemas, esto en la medida que permite desarrollar una solución sistemática siguiendo un conjunto de pasos ordenados que permitirá encontrar la solución más efectiva y en el menor tiempo posible.

14. ¿Los diagramas de flujos constituyen una de las mejores alternativas para la representación de algoritmos?

Tabla 16

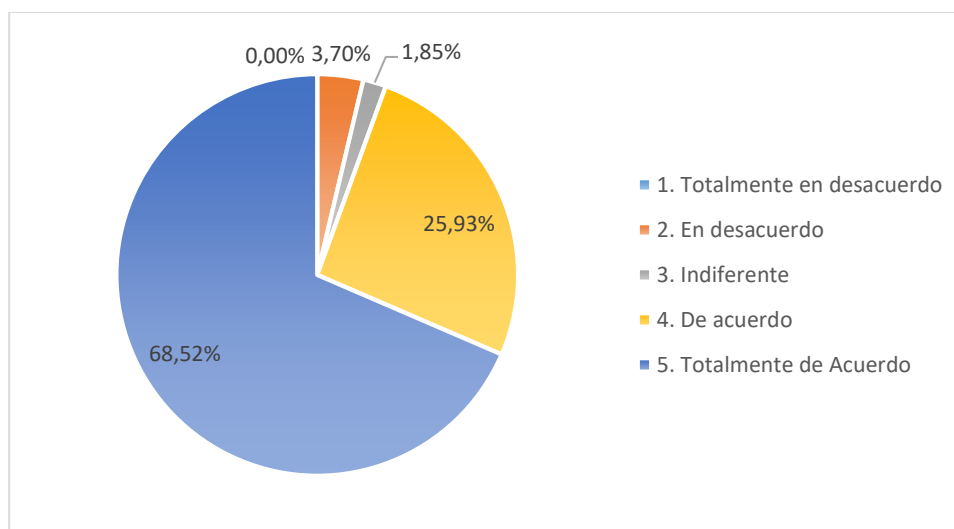
Uso de diagramas de flujo para la representación de algoritmos

Alternativas	Frecuencia	Porcentaje
1. Totalmente en desacuerdo	0	0,00%
2. En desacuerdo	1	1,85%
3. Indiferente	3	5,56%
4. De acuerdo	31	57,41%
5. Totalmente de Acuerdo	19	35,19%
Total	54	100,00%

Nota: Elaborada mediante la herramienta estadística SPSS

Gráfico estadístico 14

Los diagramas de flujo constituyen una de las mejores alternativas para la representación de los algoritmos.



Nota: Gráfico generado mediante la aplicación SPSS.

Análisis e Interpretación: En la pregunta 14, de los 54 estudiantes encuestados, el 57,41% está de acuerdo en que los diagramas de flujo constituyen una de las mejores alternativas para la representación de algoritmos; 35,19% manifiestan estar totalmente acuerdo; 5,56% señala que le es indiferente y el restante 1,85% señala que está en desacuerdo. Los diagramas de flujo permiten realizar una representación fácil de los pasos de un algoritmo, además de que al representarse en forma gráfica resulta fácil entender el flujo de los pasos que se siguen, incluso para las personas que no han participado en su desarrollo.

15. ¿Los diagramas de bloque N-S ofrecen una alternativa más simple de representar los pasos de un algoritmo?

Tabla 17

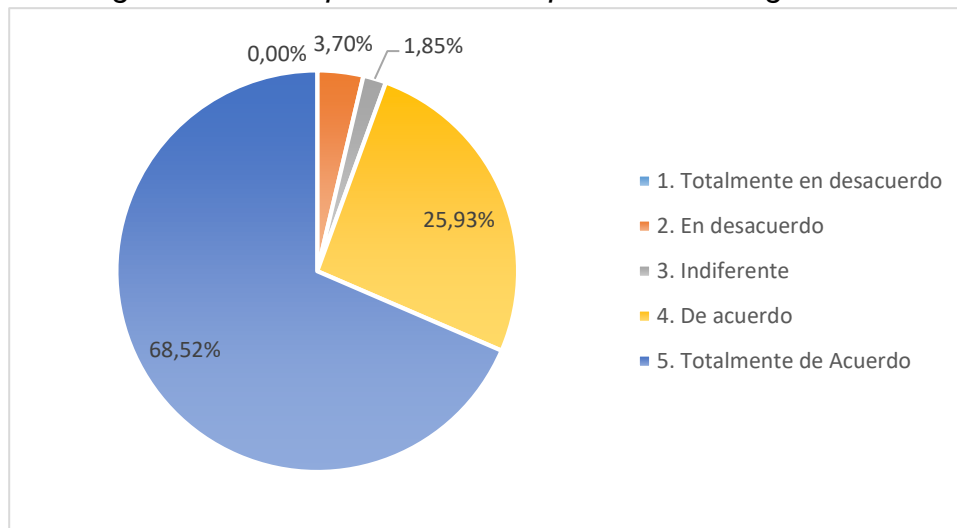
Los diagramas de bloque N-S como alternativa más simple para la representación de los pasos de un algoritmo.

Alternativas	Frecuencia	Porcentaje
1. Totalmente en desacuerdo	4	7,41%
2. En desacuerdo	4	7,41%
3. Indiferente	5	9,26%
4. De acuerdo	29	53,70%
5. Totalmente de Acuerdo	12	22,22%
Total	54	100,00%

Nota: Elaborada mediante la herramienta estadística SPSS

Gráfico estadístico 15

Los diagramas de bloque N-S en la representación algoritmos.



Nota: Gráfico generado mediante la aplicación SPSS.

Análisis e Interpretación: En la pregunta 15, de los 54 estudiantes encuestados, el 53,70% está de acuerdo en que los diagramas de bloque N-S ofrecen una alternativa más simple de representar los pasos de un algoritmo; el 22,22% manifiestan que están totalmente de acuerdo; el 9,26% indica que le es indiferente; el 7,41% señalan que están totalmente en desacuerdo y el restante 7,41% manifiestan que están en desacuerdo. Alternativamente existe esta modalidad para representar los pasos de un algoritmo, si bien es conocido que resulta sencillo su uso, no está muy difundido entre los estudiantes, ya que el programa de contenidos contempla al inicio el Pseudocódigo, dejando como opción alternativa la representación por medio de Bloques N-S.

16. ¿Los docentes que dictan la asignatura de programación poseen el conocimiento y los recursos necesarios para un efectivo proceso de enseñanza?

Tabla 18

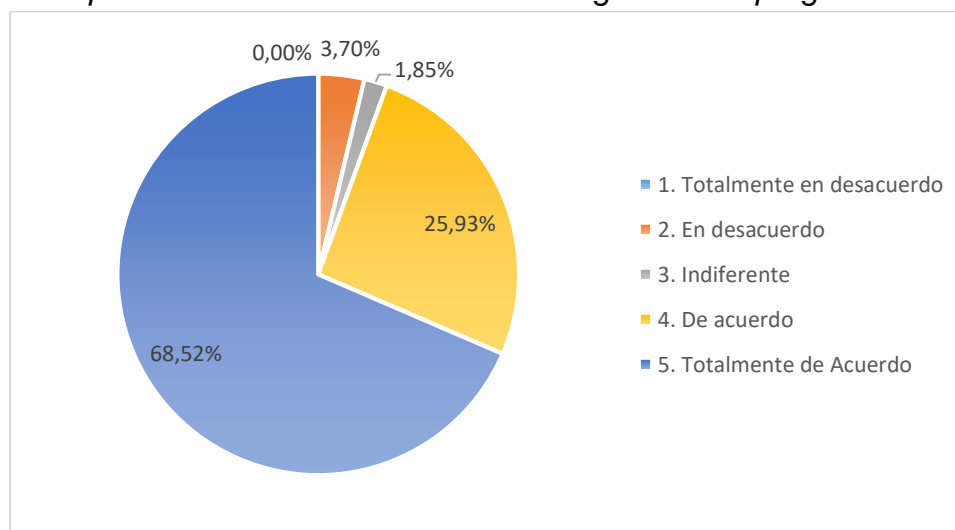
La capacitación de los docentes y los recursos necesarios para un efectivo proceso de enseñanza

Alternativas	Frecuencia	Porcentaje
1. Totalmente en desacuerdo	0	0,00%
2. En desacuerdo	2	3,70%
3. Indiferente	1	1,85%
4. De acuerdo	14	25,93%
5. Totalmente de Acuerdo	37	68,52%
Total	54	100,00%

Nota: Elaborada mediante la herramienta estadística SPSS

Gráfico estadístico 16

La capacitación de los docentes en la asignatura de programación



Nota: Gráfico generado mediante la aplicación SPSS.

Análisis e Interpretación: En la pregunta 16, de los 54 estudiantes encuestados, el 68,52% está totalmente de acuerdo en que los docentes que dictan la asignatura de programación poseen el conocimiento y los recursos necesarios para un efectivo proceso de enseñanza; el 25,93% manifiestan que están de acuerdo; el 4,70% indica que está en desacuerdo y restante el 1,85% señala que le es indiferente. Dentro de la malla curricular de Informática, la asignatura de programación es de importancia crítica, por lo que resulta esencial que el docente tenga un conocimiento profundo de la materia, que le permita guiar al estudiante hacia una efectiva construcción de su conocimiento.

4.2 Análisis correlacional de los resultados

Para la verificar la correlación entre la variable independiente y la variable dependiente como son, las herramientas TICs y su incidencia en el aprendizaje programación respectivamente, se ha utilizado la herramienta IBM SPSS, por la facilidad que ofrece para importar los datos desde los cuales se va a trabajar en el análisis y, por otro lado, tiene la capacidad de generar automáticamente las tablas y gráficos que permiten una fácil interpretación de los resultados obtenidos.

Luego de incorporar los datos obtenidos, se proceden a agrupar los ítems correspondientes a las dimensiones de la variable independiente en una nueva variable, repitiendo el proceso para la variable dependiente. En seguida mediante la herramienta SPSS se someten las dos nuevas variables a un análisis bivarial utilizando el coeficiente de relación de Pearson, de donde se obtiene la siguiente tabla:

Tabla 19

Correlaciones

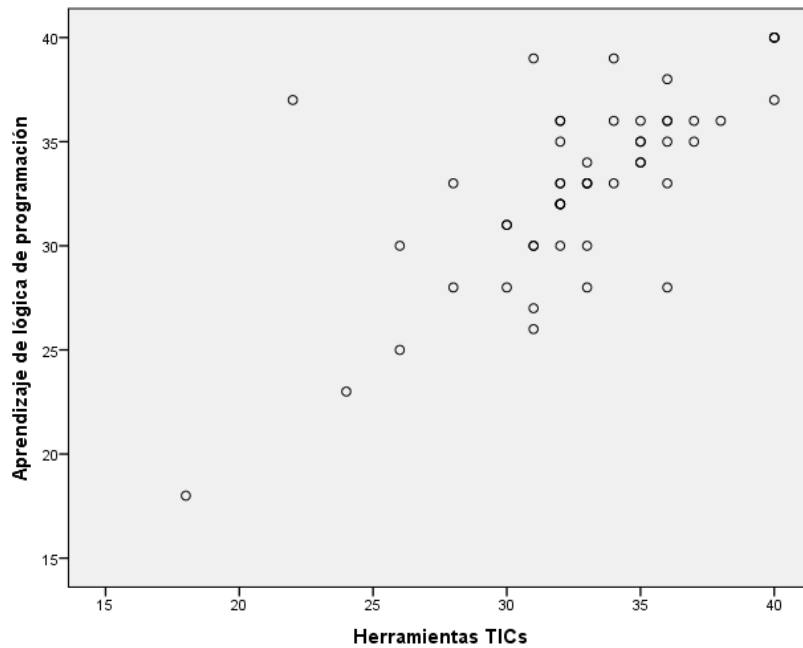
Descripción		Herramientas TICs	Aprendizaje de lógica de programación
Herramientas TICs	Correlación de Pearson	1	0,704**
	Sig. (bilateral)		0,000
	N	54	54
Aprendizaje de lógica de programación	Correlación de Pearson	0,704**	1
	Sig. (bilateral)	0,000	
	N	54	54

Nota: **. La correlación es significativa en el nivel 0,01 (bilateral). Elaborada mediante la herramienta estadística SPSS

Interpretación: En este caso en ambas variables se puede observar que el índice del coeficiente de correlación Pearson es 0.704, lo que indica una correlación positiva moderada. En cuanto a la significancia está por debajo de 0.001, por lo que tenemos un índice de confianza de 99% de que la correlación sea verdadera y un 1% de probabilidad de error. Es decir, se acepta la hipótesis de investigación en el nivel 0.01, en este sentido, la correlación entre las herramientas TIC's y el aprendizaje de lógica de programación es positiva y considerablemente fuerte.

Gráfico estadístico 17

Correlación de Variables de acuerdo a la prueba de Pearson



Nota: Gráfico generado mediante la aplicación SPSS.

En el gráfico N° 17, podemos apreciar una clara relación lineal positiva bastante fuerte entre las dos variables, corroborando así la hipótesis planteada al inicio del proyecto de que, el uso de las herramientas TIC's incide en el aprendizaje de lógica de programación en los alumnos de bachillerato.

CAPÍTULO V: Conclusiones y Recomendaciones

5.1 Conclusiones

La asignatura de lógica de programación constituye un aspecto medular dentro de la especialidad de Informática, luego de haber realizado el presente trabajo investigativo en la Unidad Educativa Urdaneta se obtienen las siguientes conclusiones:

- La enseñanza de la programación en sí requiere de la accesibilidad a distintas herramientas informáticas que servirán de apoyo tanto al docente para enseñar, como al estudiante para aprender, cada uno de los cuales ofrecen diferentes ventajas que aprovechadas adecuadamente posibilitaran alcanzar un aprendizaje significativo.
- Mediante los instrumentos utilizados para recabar los datos del presente trabajo investigativo, se ha logrado constatar que es de suma importancia el uso de herramientas TIC's en su proceso de enseñanza y aprendizaje, en el sentido de que resulta más interesante aprender mediante el uso inherente de aplicaciones que les permitan hacer uso de su creatividad e imaginación en el desarrollo de aplicaciones.
- En los datos recabados en el presente estudio, casi todos los estudiantes manifiestan estar totalmente de acuerdo en el uso de las herramientas TIC's en su proceso de aprendizaje de la asignatura de lógica de programación, lo que permite deducir en base a los datos obtenidos, lo importante que resulta integrar tecnologías como internet, aplicaciones y lenguajes de programación en las acciones pedagógicas de dicha asignatura.
- La información analizada durante el proceso de investigación demuestra una marcada preferencia en los estudiantes por la aplicación de los métodos de aprendizaje colaborativo y el aprendizaje basado en proyectos, lo mismo se puede notar en los docentes entrevistados para el efecto, en razón de que dichas metodologías ofrecen mayores ventajas en la consecución de los objetivos de aprendizaje propuestos en la asignatura de lógica de programación.

5.2 Recomendaciones

- Es necesario buscar los mecanismos que posibiliten la reducción de la brecha digital existente en los estudiantes del Bachillerato, la cual se ve más acentuada en las áreas rurales, para esto puede apoyarse en la dotación de equipos informáticos suficientes en la Unidad Educativa y también fomentando la utilización de herramientas de código abierto y con licencias GPL, que no requieran algún tipo de pago para poder hacer uso de las mismas.
- Existen actualmente un sin número de aplicaciones informáticas que son recomendables utilizar para la realización de las actividades académicas de los estudiantes, estas herramientas que se conocen como herramientas emergentes, se puede adaptar perfectamente al aprendizaje programación permitiendo verificar y evaluar el progreso académico.
- Es importante diversificar la gama de herramientas utilizadas, con cuidado por supuesto de no sobrecargar la cantidad de aplicaciones revisadas, ya que de otro modo tampoco se lograría un efectivo aprendizaje, en este sentido se deben aplicar las herramientas que permitan conseguir determinados objetivos de aprendizaje y que también el contexto en el que se desenvuelve lo permita.
- Es recomendable hacer uso extensivo de metodologías activas que propicien el trabajo en equipo y el aprendizaje colaborativo, ya que se puede notar la preferencia de los estudiantes por aquellos procesos en los que son más interactivos, tanto entre los pares como con el docente, quien por supuesto debe conocer a profundidad estos métodos y por supuesto poseer un conocimiento profundo de la materia.

Bibliografía

- Anderson Huerta, J. A., & González-Bañales, D. L. (2020). PSeInt como herramienta para mejorar el proceso de enseñanza aprendizaje de algoritmos, pseudocódigo y diagramas de flujo. *Tecnologías de La Información En Educación: Sistematización de Experiencias Docentes*, 91.
- Arias, J., Villasís, A., Keever, M., Guadalupe, M., & Novales, M. (2016). Metodología de la investigación. *Alergia México*, 63, 201–206. www.nietoeditores.com.mx
- Barceló Adrover, S. (2020). Aprendizaje basado en proyectos mediante competición con Arduino en Tecnología de 4º ESO. *Re-UNIR Repositorio Digital*. <https://reunir.unir.net/handle/123456789/9895>
- Bernal, C. (2010). Metodología de la Investigación. In □□□□□ □□□□□ (Tercera Ed, Vol. 4, Issue 3). Prentice Hall. <http://marefateadyan.nashriyat.ir/node/150>
- Bravo Macías, R. Y., & Montesdeoca Zuleta, S. (2016). Influencia de Visual Basic 2010 en el Rendimiento Académico de la asignatura de Programación. *UNIVERSIDAD DE GUAYAQUIL FACULTAD DE FILOSOFÍA, LETRAS Y CIENCIAS DE LA EDUCACIÓN*.
- Cajal Flores, A. (2020). *Investigación de campo: características, diseño y técnicas*. Liferder. <https://www.liferder.com/investigacion-de-campo/>
- Carrizo, B. R., Corso, C. L., & Olmedo, A. (2015). *Aplicación de una herramienta basada en Software Libre para la enseñanza de Algoritmos y Lógica de Programación*. 1. http://www.edutecne.utn.edu.ar/coini_2015/trabajos/F030_COINI2015.pdf
- Castillo, R., & Caro, A. (2016). *Programación con PSeInt*. 87. <https://pdfslide.net/download/link/manual-pseintdoc>
- Cearreta-Urbieto, I. (2015). *Scratch como recurso didáctico para el desarrollo del Pensamiento Computacional de los alumnos de Secundaria y Bachillerato en la asignatura de Informática y como recurso transversal en el resto de asignaturas*. 1–94. <https://reunir.unir.net/handle/123456789/3150>
- Cevallos, F. J. (2008). Enciclopedia de Microsoft Visual Basic. *Alfaomega*.
- Chavez Torres, A. (2017). Algoritmos. *Libros Universidad Nacional Abierta Ya*

Distancia, 41–59.

- Coila Villena, H. (n.d.). *Software de Aplicación*. Academia.Edu. https://www.academia.edu/39358917/SOFTWARE_DE_APLICACIÓN
- Duque, F., Saint-Priest Velásquez, Y., Segovia, P., & Loaiza, D. F. (2017). *Algoritmos Y Programación En Pseudocódigo*. www.editorialtecnologica.tec.ac.cr
- Enrique, I. J., Cañete, B., Anelys, M. S., & Ricardo, V. (2019). *The introduction of the didactical tool PSeInt in the teaching and learning process: a proposal for Linear Algebra*. 15(1), 144–153.
- Eug, F., Filot, A. L., & Brum, S. (2016). *Manual Básico De Scratch*. 1–25.
- Falco, M., Nuñez, I. J., Perea, L., Carlevari, R., & Tanzi, F. (2018). Herramienta Software como Soporte al Proceso de Enseñanza-Aprendizaje de la Programación Lineal. *Simposio Argentino de Enseñanza Superior En Informática (SAESI)-JAIIO 47*, 33–45. <http://sedici.unlp.edu.ar/handle/10915/71017>
- Fresno Chavez, C. (2019). *Metodología de la Investigacion. Así de fácil*. El Cid Editor.
- García de Jalón, X., Rodríguez, J. I., & Brazáles, A. (1999). *Aprenda Visual Basic 6.0. Universida de Navarra*.
- García López, J. C. (2013). Guía de referencia de Scratch. *Eduteka*, 1, 1–21. <http://www.eduteka.org/pdfdir/ScratchGuiaReferencia.pdf>
- Gomez, E., & Salgado, N. (2017). *LOGICA DE ALGORITMOS. Introducción a los Fundamentos de Programación en C*.
- Grajales, T. (2014). Tipos de Investigación. *DIVULGARE Boletín Científico de La Escuela Superior de Actopan*, 1(1), 4–7. <https://doi.org/10.29057/esa.v1i1.1580>
- Guerrero, M., Guamán, D. S., & Caiza, J. C. (2015). Revisión de Herramientas de Apoyo en el Proceso de Enseñanza- Aprendizaje de Programación. *Revista Politécnica Nacional (Ecuador)*, 35(1), 84–90.
- Hernández, C. E., & Carpio, N. (2019). Introducción a los tipos de muestreo. *ALERTA Revista Científica Del Instituto Nacional de Salud*, 2(1), 75–79. <https://doi.org/10.5377/alerta.v2i1.7535>
- Hernandez Sampieri, R., Fernandez Collado, C., & Baptista Lucio, P. (2014).

Metodología de Investigación (Sexta). McGraw Hill.

Jácome, H., Villareal, A., & Zapara, W. (2017). Oferta Formativa de Bachillerato Técnico. In *Ministerio de Educación del Ecuador*. <https://doi.org/10.1192/bjp.112.483.211-a>

Joyanes Aguilar, L. (2008). *Fundamentos de Programación*. (Mc Graw Hill (ed.)).

Joyanes Aguilar, L., & Sanchez García, L. (2006). *Programación en C++*. *Un enfoque práctico* (Primera Ed).

Livias Segura, N. (2018). UNIVERSIDAD NACIONAL DE EDUCACIÓN PROGRAMACIÓN VISUAL.NET Conceptos de programación visual, fundamentos del Visual NET, elementos de MS Visual, sentencias de control, procedimientos y matrices de controles, acceso a una base de datos, asistente para aplic. *Monografía*.

Microsoft. (2020). *Información de Visual Studio Code*. Herramientas de Codificación Standar. <https://code.visualstudio.com/docs>

Ministerio de Educación del Ecuador. (2016). *Bachillerato Técnico*. Oferta de Bachillerato Técnico. <https://educacion.gob.ec/bachillerato-tecnico/>

Monjelat, N. G., Cenacchi, M. A., & San Martín, P. S. (2018). ¿Programación para Todos? Herramientas y Accesibilidad: Un Estudio de Caso. *Revista Latinoamericana de Educación Inclusiva*, 12(1), 213–227. <https://doi.org/10.4067/s0718-73782018000100213>

Monsálvez, J. C. G. (2017). Python como primer lenguaje de programación textual en la Enseñanza Secundaria TT - Python as First Textual Programming Language in Secondary Education. *Education in the Knowledge Society*, 18(2), 147–162. <https://search.proquest.com/docview/1933863890?accountid=14542%0Ahttp://dn3nh3eq7d.search.serialssolutions.com/?genre=article&sid=ProQ:&atitle=Python+como+primer+lenguaje+de+programación+textual+en+la+Enseñanza+Secundaria&title=Education+in+the+Knowledge+>

Moreno, J. (2016). REDISEÑO MICRO-CURRICULAR DE LA ASIGNATURA FUNDAMENTOS DE PROGRAMACIÓN DE LA FIEC. PROPUESTA DE GUÍA METODOLÓGICA. *Repositorio Institucional de La Universidad Nde Guayaquil*.

<http://repositorio.ug.edu.ec/handle/redug/24354>

Moreno Montiel, I. D. A., & Arevalo Gamboa, L. M. (2018). *EVALUACIÓN DEL LABORATORIO DE PROGRAMACIÓN DE LA UNEMI*.

Moroni, N., & Señas, P. (2014). Estrategias Para La Enseñanza De La Programación. *Slideshare*, 1–2. <https://es.slideshare.net/Kryst77/estrategias-para-la-enseanza-de-la-geografia>

Niño Rojas, V. M. (2011). *Metodología de la Investigación. Diseño y Ejecución* (Primera Ed). Ediciones de la U. <https://doi.org/10.1515/botm.1980.23.2.117>

Ojeda, L. R. (2015). *Programación Python*. 1–315.

Prado, A., & Lara, L. R. (2018). Herramientas TIC para la enseñanza de programación , empleando aula invertida Resumen Introducción. *XIII Congreso de Tecnología En Educación y Educación En Tecnología*.

Rancel, M. R. (2021). *Normas o reglas para la creación de diagramas de flujos*. http://aprenderaprogramar.com/index.php?option=com_content&view=article&id=309:normas-o-reglas-para-la-creacion-de-diagramas-de-flujos-ejemplos-y-ejercicios-resueltos-cu00139a&catid=28:curso-bases-programacion-nivel-i&Itemid=59

Rodríguez Bucarelly, C. M. (2008). *V I S U A L B a S I C 6 . 0*. Grupo Experto Bucarelly, 3(programación orientada a objetos), 1–4.

Rodríguez, C., Rico, R., & Gutierrez, D. (2018). Manual de Prácticas de Ingeniería de Software. *UNIVERSIDAD AUTÓNOMA DEL ESTADO DE MÉXICO*, 29.

Rodríguez Monterosa, I. (2017). Algoritmos. *Boletín Científico De La Escuela Superior Atotonilco De Tula*, 4(8). <https://doi.org/10.29057/esat.v4i8.2412>

Sánchez Lopez, J. (2016). SOFTWARE 1. Sistema Operativo. Software de Aplicación. *RedCirculos*, 1–10. <https://proyectocirculos.files.wordpress.com/2013/11/software.pdf>

Shiguay, G. (2015). Las Tecnologías de la Información y Comunicación en el Aprendizaje de algoritmos y diagramas de flujo. In *2015*.

Sormenezko Zerbitzuak. (2014). Guía Didáctica para Profesores. *Isuri*, 06.

<http://paginaspersonales.deusto.es/garaizar/scratch/doc/ScratchGuíaDidácticaProfesores.pdf>

- Travieso Merino, C. (2017). Presentación de Scratch. *Academia Crsalis*, 7.
- Trejos Buriticá, O. I. (2017). PROGRAMACIÓN Imperativa CON LENGUAJE C. *Ecoe Ediciones*, 19–23. <https://www.ecoediciones.com/wp-content/uploads/2017/11/Programación-imperativa-con-lenguaje-C.pdf>
- Universidad de Málaga. (1999). Técnicas de diseño de algoritmos. In *Universidad de Málaga* (Vol. 2).
- Universidad de Sevilla. (2015). *Introducción al Desarrollo de Aplicaciones con Visual Basic. 1*. <https://www.cs.us.es/cursos/ai-2005/VB/VB.pdf>
- Villadiego Alvarez, A., Lopez Gonzales, J., & Sierra Pineda, I. (2015). *Desarrollo Del Pensamiento Creativo En Estudiantes De Educación Media*.
- Zamora Lucio, M. A. (2018). *Todos deberíamos saber crear algoritmos*. Con-Ciencia Boletín Científico de La Escuela Preparatoria No. 3. <https://repository.uaeh.edu.mx/revistas/index.php/prepa3/article/download/2829/2852?inline=1>
- Zuleta Medina, A., & Chaves Torres, A. (2011). Uso de herramientas Informáticas como estrategia para la Enseñanza de la Programación de Computadores. *Revista Unimar*, 23–32.

ANEXOS

ANEXO 1



UNIVERSIDAD ESTATAL DE MILAGRO INSTITUTO DE POSTGRADO Y EDUCACIÓN CONTINUA

CARTA AVAL DE APROBACION DEL TEMA DE INVESTIGACIÓN



MAESTRÍA EN EDUCACIÓN <maestria_educacionit@unemi.edu.ec>
para mí ▾

lun, 14 dic 2020 13:22



Estimado: Chacon Valverde Walter Egidio

La comisión encargada de evaluar las propuestas de trabajos de titulación para la convocatoria 2020-1, del programa de Maestría en Educación, mención Tecnología e Innovación Educativa, luego de revisar su propuesta de: Informe de investigación, con el tema: *El uso de herramientas TIC's en el aprendizaje de Lógica de Programación de los estudiantes de Bachillerato..*

Ha decidido: **Aprobar la propuesta de trabajo de titulación e iniciar el proceso de asignación de tutor.**
Se sugiere modificar el título del proyecto con el fin de establecer de manera específica el alcance del mismo, en concordancia con lo planteado en el objetivo general y objetivos específicos.

Saludos cordiales

ANEXO 2



UNIVERSIDAD ESTATAL DE MILAGRO
INSTITUTO DE POSTGRADO Y EDUCACIÓN CONTINUA
CRONOGRAMA DE DESARROLLO DEL PROYECTO DE INVESTIGACIÓN

DESCRIPCIÓN	DETALLE DE SEMANAS DE TRABAJO										
	01-mar-21	08-mar-21	15-mar-21	22-mar-21	29-mar-21	05-abr-21	12-abr-21	19-abr-21	26-abr-21	03-may-21	07-may-21
CAPÍTULO I: El problema de la investigación											
CAPÍTULO II: Marco teórico referencial											
CAPÍTULO III: Metodología											
CAPÍTULO IV: Análisis e interpretación de resultados											
CAPÍTULO V: Conclusiones y Recomendaciones											
Entrega del Informe de investigación											

ANEXO 3



UNIVERSIDAD ESTATAL DE MILAGRO

INSTITUTO DE POSTGRADO Y EDUCACIÓN CONTINUA

HOJA DE REGISTRO PARA LA VALIDACIÓN POR EXPERTOS

Objetivo: Determinar la incidencia de las herramientas TIC's en el aprendizaje de la asignatura de Programación.

1. ¿Es importante el uso de la herramienta PSeInt para la resolución de algoritmos?
 1. Totalmente en desacuerdo
 2. En desacuerdo
 3. Indiferente
 4. De acuerdo
 5. Totalmente de acuerdo

2. ¿La herramienta PSeInt adecuada para representar los algoritmos mediante diagramas de flujo?
 1. Totalmente en desacuerdo
 2. En desacuerdo
 3. Indiferente
 4. De acuerdo
 5. Totalmente de acuerdo

3. ¿El lenguaje de programación C++ presenta una estructura sencilla que brinda facilidad para escribir sus primeros programas?
 1. Totalmente en desacuerdo
 2. En desacuerdo
 3. Indiferente
 4. De acuerdo
 5. Totalmente de acuerdo

4. ¿El lenguaje de programación Python ofrece flexibilidad al momento de codificar los programas?
 1. Totalmente en desacuerdo
 2. En desacuerdo
 3. Indiferente
 4. De acuerdo
 5. Totalmente de acuerdo

5. ¿El ambiente grafico de la herramienta Scratch permite entender con facilidad la estructura de un programa?
 1. Totalmente en desacuerdo
 2. En desacuerdo
 3. Indiferente
 4. De acuerdo
 5. Totalmente de acuerdo

6. ¿El diseño de aplicaciones es más sencillo utilizando la interfaz de controles y formularios de la herramienta VB.Net?
 1. Totalmente en desacuerdo
 2. En desacuerdo
 3. Indiferente
 4. De acuerdo
 5. Totalmente de acuerdo

7. ¿Disponer de acceso a internet es esencial para el aprendizaje de la asignatura de programación?
 1. Totalmente en desacuerdo
 2. En desacuerdo
 3. Indiferente
 4. De acuerdo
 5. Totalmente de acuerdo

8. ¿Contar con las herramientas tecnológicas necesarias en la institución educativa o en el hogar permite al estudiante alcanzar un mejor nivel aprendizaje?
 1. Totalmente en desacuerdo
 2. En desacuerdo
 3. Indiferente
 4. De acuerdo
 5. Totalmente de acuerdo

9. ¿La realización de actividades grupales y/o colaborativas resultan efectivas para el aprendizaje en la elaboración programas?
 1. Totalmente en desacuerdo
 2. En desacuerdo
 3. Indiferente
 4. De acuerdo
 5. Totalmente de acuerdo

10. ¿Es importante que las actividades propuestas para desarrollar algoritmos estén basadas en un contexto real?

1. Totalmente en desacuerdo
2. En desacuerdo
3. Indiferente
4. De acuerdo
5. Totalmente de acuerdo

11. ¿La metodología basada en proyectos es importante para lograr un adecuado desarrollo de los programas?

1. Totalmente en desacuerdo
2. En desacuerdo
3. Indiferente
4. De acuerdo
5. Totalmente de acuerdo

12. ¿Es necesario el uso frecuente de herramientas TIC's para mejorar la destreza en el proceso de elaboración de programas?

1. Totalmente en desacuerdo
2. En desacuerdo
3. Indiferente
4. De acuerdo
5. Totalmente de acuerdo

13. ¿El uso de algoritmos es el método más adecuado para resolver determinados problemas cotidianos?

1. Totalmente en desacuerdo
2. En desacuerdo
3. Indiferente
4. De acuerdo
5. Totalmente de acuerdo

14. ¿Los diagramas de flujos constituyen una de las mejores alternativas para la representación de algoritmos?

1. Totalmente en desacuerdo
2. En desacuerdo
3. Indiferente
4. De acuerdo
5. Totalmente de acuerdo

15. ¿Los diagramas de bloque N-S ofrecen una alternativa más simple de representar los pasos de un algoritmo?

1. Totalmente en desacuerdo
2. En desacuerdo
3. Indiferente
4. De acuerdo
5. Totalmente de acuerdo

16. ¿Los docentes que dictan la asignatura de programación poseen el conocimiento y los recursos necesarios para un efectivo proceso de enseñanza?

1. Totalmente en desacuerdo
2. En desacuerdo
3. Indiferente
4. De acuerdo
5. Totalmente de acuerdo

Nota Aclaratoria: Debido a las circunstancias por las que estamos atravesando por la pandemia del COVID 19, la encuesta fue aplicada mediante la herramienta *Google Forms*.



HERRAMIENTAS TIC'S Y EL APRENDIZAJE DE PROGRAMACIÓN

Objetivo: Determinar la incidencia de las herramientas TIC's en el aprendizaje de la asignatura de Programación.

Procedimiento

Lea detenidamente cada una de las siguientes preguntas y responda seleccionando una respuesta, en las Totalmente en Desacuerdo representa el valor mas bajo de concordancia y Totalmente de Acuerdo representa el valor mas alto de concordancia.



UNIVERSIDAD ESTATAL DE MILAGRO

INSTITUTO DE POSTGRADO Y EDUCACIÓN CONTINUA

ENTREVISTA AL RECTOR DE LA UNIDAD EDUCATIVA “URDANETA”

1. ¿Cree usted que les resulta complejo a los estudiantes de la especialidad Informática adquirir las habilidades necesarias para diseñar y construir programas de computadoras?

.....
.....

2. ¿Cuáles son las herramientas de software que aplican los docentes en la enseñanza de la asignatura de programación?

.....
.....

3. Indique si conoce alguna metodología en particular con considere adecuada para enseñar a programar a los estudiantes de especialidad de Informática.

.....
.....

4. ¿Cómo cree usted que afecta la disponibilidad de los recursos tecnológicos al proceso de enseñanza y aprendizaje de la asignatura de programación?

.....
.....

5. ¿En que beneficiaría a los estudiantes de informática, desarrollar las destrezas para elaborar algoritmos y programas de computadoras?

.....
.....

ANEXO 5



UNIVERSIDAD ESTATAL DE MILAGRO

INSTITUTO DE POSTGRADO Y EDUCACIÓN CONTINUA

HOJA DE REGISTRO PARA LA VALIDACIÓN POR EXPERTOS

Maestrante: Ing. Walter Egidio Chacon Valverde

Tutor: Msc. Denis Darío Mendoza Cabrera

Datos del Experto

Nombres y Apellidos	
Última titulación académica	
Institución de adscripción	
Cargo	
Teléfono celular	
Dirección de correo	

Instrumento.

Formato de encuesta para estudiantes de Programación

Sobre el instrumento.

Se presenta, para su validación, el formato de encuesta para estudiantes, cuyo objetivo es: "Determinar la incidencia del uso de las herramientas TIC's en el proceso de aprendizaje de Lógica de programación a través de un cuestionario"

El presente cuestionario se ha elaborado a partir del Cuadro de operacionalización de variables.

La definición conceptual y operacional de la variable independiente *Herramientas TIC's* es:

"Las herramientas TIC ofrecen tanto a docentes como a los alumnos multitud de posibilidades para facilitar el aprendizaje colaborativo, un enfoque didáctico con grandes ventajas para los estudiantes."

La definición conceptual y operacional de la variable dependiente *Aprendizaje de Lógica de programación* es:

"El aprendizaje de los lenguajes de programación constituye una de las necesidades a atender por todo centro educativo, en cualquier nivel de enseñanza".

Sobre la validación

A continuación, se presentan dos tablas, con la referencia numérica de los ítems o aspectos sobre los que se indaga a través de cada cuestionario.

Por favor, valore cada ítem de acuerdo con los siguientes criterios:

- **(S) Suficiencia:** Los ítems que evalúan el mismo componente bastan para obtener la medición de este.
- **(CI) Claridad:** El ítem se comprende fácilmente, es decir, su sintáctica y semántica son adecuadas.
- **(Co) Coherencia:** El ítem tiene relación lógica con el componente sobre el que se supone que indaga.
- **(R) Relevancia:** El ítem es esencial o importante, es decir debe ser incluido.

Para ello, coloque en la casilla correspondiente un número del uno (1) al cuatro (4) de acuerdo con la siguiente escala:

1. No cumple con el criterio	2. Bajo nivel	3. Moderado nivel	4. Alto nivel
------------------------------	---------------	-------------------	---------------

Además de su valoración, por favor, agregue las observaciones que explican su valoración o ayudan a la mejora de la pregunta.

Instrumento 1: Encuesta para estudiantes de la asignatura Programación

Pregunta por componente	(S)	(CI)	(Co)	(R)	Observación
1. ¿Es importante el uso de la herramienta PSeInt para la resolución de algoritmos?					
2. ¿La herramienta PseInt es adecuada para representar los algoritmos mediante diagramas de flujo?					
3. ¿El lenguaje de programación C++ presenta una estructura sencilla que brinda facilidad para escribir sus primeros programas?					
4. ¿El lenguaje de programación Python ofrece flexibilidad al momento de codificar los programas?					
5. ¿El ambiente grafico de la herramienta Scratch permite entender con facilidad la estructura de un programa?					
6. ¿El diseño de aplicaciones es más sencillo utilizando la interfaz de controles y formularios de la herramienta VB.Net?					

Pregunta por componente	(S)	(CI)	(Co)	(R)	Observación
7. ¿Disponer de acceso a internet es esencial para el aprendizaje de la asignatura de programación?					
8. ¿Contar con las herramientas tecnológicas necesarias en la institución educativa o en el hogar permite al estudiante alcanzar un mejor nivel aprendizaje?					
9. ¿La realización de actividades grupales y/o colaborativas resultan efectivas para el aprendizaje en la elaboración programas?					
10. ¿Es importante que las actividades propuestas para desarrollar algoritmos estén basadas en un contexto real?					
11. ¿La metodología basada en proyectos es fundamental para lograr un adecuado desarrollo de los programas?					
12. ¿Es necesario el uso frecuente de herramientas TIC's para mejorar la destreza en el proceso de elaboración de programas?					
13. ¿El uso de algoritmos es el método más adecuado para resolver determinados problemas cotidianos?					
14. ¿Los diagramas de flujos constituyen una de las mejores alternativas para la representación de algoritmos?					
15. ¿Los diagramas de bloque N-S ofrecen una alternativa más simple de representar los pasos de un algoritmo?					
16. ¿Los docentes que dictan la asignatura de programación poseen el conocimiento y los recursos necesarios para un efectivo proceso de enseñanza?					

Consideraciones sobre el instrumento revisado.
Sugerencias y recomendaciones.



ANEXO 6

MATRIZ DE CONSISTENCIA

Problema	Objetivos	Hipótesis	Variables	Dimensiones	Indicadores	Instrumento
¿Cómo incide el uso de las Herramientas TIC's en el aprendizaje de Lógica de programación en los alumnos de bachillerato de la Unidad Educativa Urdaneta, del periodo lectivo 2020-2021?	<p>Objetivo General Determinar la incidencia del uso de las Herramientas TIC's en el aprendizaje de Lógica de programación, para mejorar su desempeño académico.</p> <p>Objetivos Específicos</p> <ul style="list-style-type: none"> – Analizar las herramientas TIC's aplicadas por los docentes de Programación Estructurada y bases de datos. – Evaluar mediante encuestas y entrevistas la importancia de del uso de herramientas TIC's en el aprendizaje de lógica de Programación de los estudiantes de Bachillerato. – Determinar cómo afecta a los estudiantes la disponibilidad de internet y otros recursos tecnológicos en su aprendizaje de lógica de programación. – Identificar las principales estrategias que emplean los docentes para la enseñanza de lógica de programación a los estudiantes de Bachillerato. 	El uso de las Herramientas TIC's incide en el aprendizaje de Lógica de programación en los alumnos de bachillerato	Independiente: Herramientas TIC's	Herramientas Básicas	Software PseInt	Encuesta
					Lenguaje de Programación C++	Encuesta
				Herramientas Avanzadas	Lenguaje de Programación Python	Encuesta
					Software Scratch	Encuesta
					Herramienta de Diseño	Encuesta
				Disponibilidad Tecnológica	Conectividad	Encuesta
					Dispositivos Tecnológicos	Encuesta
				Dependiente: Aprendizaje de Lógica de Programación	Estrategias Metodológicas	Aprendizaje colaborativo
			Aprendizaje Basado en Problemas			Encuesta
			Aprendizaje Basado en Proyectos			Encuesta
			Uso de Herramientas TIC's		Importancia	Encuesta
			Ejes temáticos		Algoritmos	Encuesta
					Diagramas de Flujos	Encuesta
					Diagramas de N-S	Encuesta
Capacitación Docentes	Conocimientos.	Encuesta				

ANEXO 7

AUTORIZACION DE LA UNIDAD EDUCATIVA

	UNIDAD EDUCATIVA "URDANETA" Dir.: Coop. 6 de Abril 2da Etapa e-mail: col_urdaneta@hotmail.es Telf.: 0988217790			
	CÓDIGO AMIE: 12H01030	DISTRITO: 12D02	CIRCUITO 12D01C05_06	

CERTIFICACION

Ricaurte, 17 de diciembre del 2020

Señores

UNIVERSIDAD ESTATAL DE MILAGRO

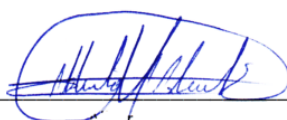
Presente.-

De mis consideraciones:

Por medio de la presente certifico que he concedido la AUTORIZACIÓN al Ing. Walter Egidio Chacón Valverde, para que desarrolle en la Unidad Educativa "Urdaneta", institución que dirijo, para que desarrolle su proyecto de Maestría "EL USO DE HERRAMIENTAS TIC'S EN EL APRENDIZAJE DE LÓGICA DE PROGRAMACIÓN DE LOS ESTUDIANTES DE BACHILLERATO", a partir del mes de enero del 2021.

Este documento puede ser presentado a las Autoridades Educativas de la Universidad Estatal de Milagro, como requisito previo a la obtención de su título de Máster en Educación mención Tecnología en Innovación.

Atentamente,



Lic. Alberto Alvarado Yáñez

RECTOR



ANEXO 8

SOLICITUD AL RECTOR DE LA UNIDAD EDUCATIVA URDANETA

Ricaurte, 15 de diciembre del 2020

Lic.
Alberto Alvarado Yáñez
RECTOR DE LA UNIDAD EDUCATIVA URDANETA.
Presente.-

De mis consideraciones:

Por medio de la presente me dirijo a usted expresándole un cordial saludo y al mismo tiempo distraigo su atención para manifestarle que estoy realizando mis estudios de cuarto nivel, previo a la obtención del título de **MASTER EN EDUCACIÓN MENCIÓN TECNOLOGÍA E INNOVACIÓN EDUCATIVA**, dentro de las varias gestiones académicas demandadas por la universidad, consta la aplicación de una encuesta con el respectivo instrumento e indicadores para el efecto. Por lo expuesto señor Rector muy comedidamente le solicito, salvo su criterio, me conceda la autorización para aplicar el referido trabajo en la institución que usted acertadamente dirige.

Por la favorable acogida que se digne dar a la presente le anticipo mis sinceros agradecimientos.

Atentamente,



Ing. Walter Chacon Valverde
Docente de la Unidad Educativa Urdaneta

