



Sistema de Tutoría Inteligente como herramienta de apoyo en el aprendizaje de algoritmos computacionales a nivel licenciatura.

Presentado por:

Mario Humberto Rodríguez Chávez

En cumplimiento de los requisitos para el grado de Doctor en Gestión e Innovación
Educativa

Universidad Autónoma de Tamaulipas

Unidad Académica Multidisciplinaria de Ciencias de la Educación y Humanidades

Doctorado en Gestión e Innovación Educativa

Director de tesis: Dr. Arturo Amaya Amaya.

Febrero del 2022

Dedicatoria

Primeramente, a Dios por ofrecerme la oportunidad de seguir superándome, a mi familia que con cariño me han brindado de su tiempo para el desarrollo de esta investigación, a mis amores mis hijos Marianita y Humberto quienes son mi razón de ser y mi mayor orgullo y a mi gran amor mi esposa Karla quien con amor siempre ha estado en los momentos de apoyo alentándome a seguir adelante, los amo con todo mi corazón. Esto va por ustedes y para ustedes.

Agradecimientos

En primera instancia quiero agradecer a todas las y los docentes doctores quienes con su sabiduría y apoyo me acompañaron durante el proceso del desarrollo del proyecto de investigación.

A mi asesor y director de tesis el Dr. Arturo Amaya Amaya quien con sus aportaciones y seguimiento oportuno me brindó de sus conocimientos para desarrollar un trabajo de calidad.

A las doctoras Evelia Reséndiz Balderas, Celia Reyes Anaya y Teresa de Jesús Guzmán Acuña quienes siempre estuvieron al pendiente de mi trabajo brindándome su apoyo en el desarrollo de cada una de las secciones de la tesis.

A la Dra. Ma. del Rosario Contreras Villarreal, directora del posgrado de la UAMCEH quién siempre estuvo al pendiente de mi trabajo y me brindó de su apoyo académico – administrativo para continuar de manera oportuna el desarrollo del proyecto de investigación.

A todos, ¡Muchas gracias!

Tabla de contenidos

	Pág.
Resumen	I
Abstract	II
Capítulo I. Introducción	1
Contexto de la investigación	1
Antecedentes de la investigación	4
Planteamiento del problema	8
Preguntas de investigación	14
Pregunta general	14
Preguntas específicas	14
Objetivos	15
Objetivo general	15
Objetivos específicos	15
Definición de variables	15
Justificación de la investigación	16
Capítulo II. Marco Teórico	18
Proceso enseñanza y aprendizaje	18
Estilos de aprendizaje	19
Teorías del aprendizaje	23
Teoría psicología cognitiva	23
Características de la psicología cognitiva	24
La psicología cognitiva en la investigación	26
Tendencias de la psicología cognitiva	28
Justificación de la teoría utilizada en la investigación	30
Sistemas de tutoría inteligente (ITS)	31
Contexto de los ITS	31
Modelos de sistemas de tutoría inteligente	33
Sistemas de tutoría inteligente en el contexto educativo	35
Ejemplos de Sistemas de Tutoría Inteligente	38
Metodologías para el desarrollo de ITS	40
Selección de la plataforma gestor de aprendizaje (SGA)	43
Diseño de módulo del alumno	44
Diseño de módulo de tutor	44
Diseño de módulo de dominio	45
Rutas de aprendizaje en el diseño de algoritmos computacionales	46
Habilidades para el desarrollo de algoritmos	46
Especificaciones para la construcción de algoritmos computacionales	48
Capítulo III. Metodología de la Investigación	51
Diseño de la investigación	51
Sujetos de investigación	54
Procedimiento	55
Técnicas	57
Validez de los instrumentos	59
Confiabilidad	60

Aspectos éticos	61
Planificación de la metodología	62
Caracterización del conocimiento	63
Actividades	64
Implementación de un modelo ITS	65
Evaluación de los aprendizajes del grupo experimental	68
Actividades	69
Capítulo IV. Análisis de Resultados	71
Validación del instrumento de caracterización del conocimiento	72
Resultados caracterización del conocimiento	78
Modelo propuesto: ALGO-ITS	93
Validación del instrumento uso del modelo ALGO-ITS	95
Modelo ALGO – ITS evaluación del conocimiento	98
Capítulo V. Conclusiones	110
Validación de cumplimiento de los objetivos de investigación	110
Discusión de los conocimientos	114
Limitaciones de la investigación	117
Trabajo futuro	117
Glosario	119
Referencias bibliográficas	120
Anexos	131
Anexo 1: Instrumento de recolección de datos.	131
Anexo 2: Matriz de consistencia	139
Anexo 3: Base de datos software SPSS	140

Lista de tablas

	Pág
Tabla 1. Población estudiantil de la Universidad Politécnica de Victoria de agosto 2020 a diciembre 2021.	9
Tabla 2. Estadísticas de no acreditación de asignaturas periodo de enero del 2019 a agosto del 2020 de la Ing. en TI de la UPV.	12
Tabla 3. Concentrado general de materias con mayor índice de reprobación del periodo enero 2019 – agosto 2020 del programa académico (PE) de ITI de la UPV.	13
Tabla 4. Definición de variables de investigación.	16
Tabla 5. Concentrado general de los modelos cognitivos de estilos de aprendizaje.	20
Tabla 6. Concentrado general de los modelos de estilos de aprendizaje de orientación psicológica.	21
Tabla 7. Concentrado general de factores para el aprendizaje.	22
Tabla 8. Concentrado general de las operaciones cognitivas que desarrollan los estudiantes.	22
Tabla 9. Taxonomía de los eventos cognitivos.	26
Tabla 10. Fases para la construcción de algoritmos computacionales.	50
Tabla 11. Validez de los instrumentos de recolección de datos.	60
Tabla 12. Planificación de la metodología de investigación.	62
Tabla 13. Nivel de conocimiento de algoritmos a partir de la taxonomía de Marzano Kendall (2007).	65
Tabla 14. Ítems de post-test clasificados en la taxonomía de Marzano Kendall (2007).	70
Tabla 15. Estadísticas de fiabilidad.	72
Tabla 16. Estadísticas del elemento.	73
Tabla 17. Ítems identificados con mayor índice de dificultad.	74
Tabla 18. Estadísticas de elemento resumen.	75
Tabla 19. Estadísticas del total de elementos.	76
Tabla 20. Estadísticas de la escala.	77
Tabla 21. Nivel de conocimiento de los estudiantes encuestados en cada una de las categorías de la taxonomía de Marzano y Kendall (2007).	89
Tabla 22. Resultados de cuestionamientos de la muestra.	91
Tabla 23. Resultados de los estudios, conocimientos y habilidades.	91
Tabla 24. Resultados de actividades de tutorías y asesorías.	92
Tabla 25. Temáticas que los alumnos encuestados solicitaron asesorías académicas.	92
Tabla 26. Estadísticas de fiabilidad.	95
Tabla 27. Estadísticas del elemento.	96
Tabla 28. Estadísticas de elemento resumen.	97
Tabla 29. Estadísticas del total de elementos.	97
Tabla 30. Estadísticas de la escala.	98
Tabla 31. Categorización del conocimiento del grupo experimental a partir de la taxonomía de Marzano y Kendall (2007).	109
Tabla 32. Evolución del conocimiento respecto al desarrollo de algoritmos.	115

Lista de figuras

	Pág.
Figura 1. Arquitectura de un ITS.	34
Figura 2. Arquitectura SQL-TUTOR.	36
Figura 3. Metodología para la construcción de un ITS.	43
Figura 4. Habilidades ejecutoras necesarias para el desarrollo de algoritmos computacionales.	47
Figura 5. Proceso de transformación de algoritmo a programa ejecutable.	49
Figura 6. Dimensiones del conocimiento de la taxonomía de Marzano y Kendall.	57
Figura 7. Bachillerato estudiado por los estudiantes entrevistados.	79
Figura 8. Especialidad estudiada por los estudiantes entrevistados.	79
Figura 9. Experiencia en el desarrollo de algoritmos.	80
Figura 10. ¿Cuál es la estructura de datos que permite múltiples valores organizados de forma indexada?	82
Figura 11. Elección de la opción correcta que enlista cada tipo de representación de un algoritmo.	82
Figura 12. Identificación el tipo de dato que admite valores con punto decimal de 16 bits.	83
Figura 13. Procedimiento sobresaliente para la construcción de diagramas de flujo.	84
Figura 14. Identificación de arreglos (array).	85
Figura 15. Identificación de la salida de un diagrama de flujo.	86
Figura 16. Identificación de la declaración correcta de variables.	87
Figura 17. Identificación de la salida de un algoritmo.	87
Figura 18. Identificación de la suma correcta de variables.	88
Figura 19. Arquitectura de ALGO-ITS	93
Figura 20. Ítem relacionado a los conocimientos de lenguajes de programación	99
Figura 21. Estructura de datos de valores múltiples.	100
Figura 22. Representación gráfica de algoritmos.	101
Figura 23. Tipos de datos en el desarrollo de algoritmos.	102
Figura 24. Procedimientos en la construcción de diagramas de flujo.	103
Figura 25. Identificación de arreglos (array).	104
Figura 26. Identificación de la salida de un diagrama de flujo.	104
Figura 27. Identificación de la declaración correcta de variables.	106
Figura 28. Identificación de la salida de un algoritmo.	107
Figura 29. Identificación de la suma correcta de variables.	108

Resumen

El uso de la tecnología en el aula proporciona soporte para acelerar el aprendizaje en los estudiantes, así como la identificación de debilidades de conocimiento para su fortalecimiento gracias a las herramientas tecnológicas. Quispe (2014) determina que la evolución de la tecnología ha desarrollado nuevas herramientas didácticas para fortalecer la educación, entre ellas se encuentran los Sistemas Tutores Inteligentes (ITS) los cuales son herramientas instruccionales que hacen uso de la Inteligencia Artificial (IA) para tomar decisiones en lo que se debe enseñar a cada estudiante y las estrategias más convenientes para ayudar a comprender claramente los conocimientos. Con las tecnologías innovadoras, la educación personalizada es posible y a través de los ITS se puede determinar el aprendizaje de los alumnos y ofrecer una alternativa para el conocimiento de los temas en particular.

Una de las principales características de los ITS es su capacidad de transformarse en expertos en el área, toma decisiones y aprenden de los casos que se les presentan, así mismo evalúan constantemente al estudiante para poder decidir qué contenido debe enseñarse y cuál es la mejor forma de mostrárselo de manera particular (ECURED, 2019).

El proyecto objeto de esta investigación es el análisis del impacto de la implementación de un ITS que proporcione soporte en el aprendizaje de algoritmos computacionales en los alumnos de la generación 2020 de la Universidad Politécnica de Victoria (UPV) para mejorar el rendimiento académico y reducir el índice de reprobación y bajo aprendizaje sobre el desarrollo de software.

Palabras clave: Tutor inteligente, aprendizaje, algoritmos

Abstract

The use of technology in the classroom provides support to accelerate learning in students, as well as the identification of knowledge weaknesses for their strengthening thanks to technological tools. Quispe (2014) determines that the evolution of technology has developed new didactic tools to strengthen education, among them are the Intelligent Tutor Systems (ITS) which are instructional tools that make use of Artificial Intelligence (AI) to make decisions in what should be taught to each student and the most suitable strategies to help to understand clearly the knowledge. With innovative technologies, personalized education is possible and through ITS the learning of students can be determined and an alternative to the knowledge of particular subjects can be offered.

One of the main characteristics of ITS is their ability to become experts in the area, make decisions and learn from the cases that are presented to them, likewise they constantly evaluate the student to be able to decide what content should be taught and what is the best way to show it to him in a particular way (ECURED, 2019).

The project object of this research is the analysis of the impact of the implementation of an ITS that provides support in the learning of computational algorithms in the students of the 2020 generation of the Polytechnic University of Victoria (UPV) to improve academic performance and reduce the failure rate and low learning about software development.

Keywords: Intelligent tutor, learning, algorithms

Capítulo I

Introducción

Contexto de la Investigación

El constante movimiento económico y social del mundo refleja una creciente demanda de expertos en el área de las Tecnologías de la Información y Comunicaciones (TICS), particularmente el desarrollo de software. Toribio (2019) en su publicación afirma que, en un país como México de 128 millones de habitantes, enfrenta un déficit de 20,000 ingenieros y profesionales para este perfil, en los próximos 5 años.

Regil (2019) en su publicación indica que la industria 4.0 antecede que las transformaciones económicas, sociales y tecnológicas no están limitadas a una sola entidad; son globales y por tal motivo necesitan respuestas completas que permitan una innovación en las diferentes actividades, con un sentido de creatividad y coordinación. Para lograr estos cambios y generar recursos humanos competentes es necesario integrar en los ambientes educativos un sentido de transformación e innovación que genere un ambiente de motivación para el conocimiento. Para Innovar es necesario introducir un cambio en los procesos educativos y tecnológicos integrando novedades significativas de mejora.

Como se puede leer en la tesis doctoral de Gallego (2011):

La innovación, es el cambio educativo y la reforma han sido conceptos muy recurrentes desde mediados del siglo XX hasta la actualidad. Varios investigadores han tratado la innovación como un proceso que produce el cambio. La innovación cobra relevancia dentro de las organizaciones escolares y de los procesos, como algo a generar desde el propio centro. (p.57)

La transformación tecnológica y educativa es considerada parte del procedimiento innovador de las tareas afines a llevar a cabo actividades de mejor calidad. En recientes décadas con la integración de las TICs al hablar de innovar es significado de hacer cambios de mejora en el sistema educativo para obtener así educandos de calidad en el aula.

La industria 4.0 contribuye a una solución empresarial integral que se centra en la habilitación de la estrategia empresarial, la integración de sistemas y datos, facilitando la optimización empresarial en tiempo real, extiende una investigación en simulaciones de negocios alineados con procesos de negocio con el objetivo de asegurar datos operativos para crear un sistema cibernético (Telukdariea, A. et al., 2018).

Aún y cuando organizaciones como la ANUIES plantean reducir la brecha digital a través del enfoque de los aprendizajes de los estudiantes que culminan sus planes de estudio académicos a través de los requerimientos que soliciten las compañías privadas y entidades gubernamentales, la motivación del egresado profesional se ve reducida frente a numerosas y variadas habilidades que debe dominar para programar computadoras a través de diferentes lenguajes de programación. Un estudiante de nuevo ingreso, que toma un primer curso de algoritmos y programación computacional, tiene carencias de conocimiento asociadas con elementos relacionados con múltiples aspectos de su historia académica y que pueden ser variados como la capacidad de entender un texto, analizar y resolver un problema matemático o plantear una estrategia de solución para llevar a cabo la implementación de un algoritmo a una problemática presentada. El éxito de superar estos obstáculos está relacionado con dos elementos principales en el proceso de enseñanza – aprendizaje: el estudiante y el profesor. El primero al tener tenacidad e iniciativa de aprender, entendiendo el valor que tiene el aprendizaje para el éxito de emplearse como experto en la

industria del software y el segundo al identificar las fortalezas y debilidades del alumno para plantearle un plan de acción que le permita superarse con la frustración nula.

Incorporar nuevas políticas educativas en el proceso de enseñanza-aprendizaje requiere adaptar los planes de estudio, manuales de asignatura y métodos.

En las actividades docentes es muy común el uso de la tecnología a través de las Tecnologías de la Información y Comunicaciones (TIC), entre ellas destaca el uso de la computadora la cual ha desempeñado una función sobresaliente y significativa por sus grandes características de procesamiento. En la medida que avanza la innovación tecnológica, surgen cambios en el proceso educativo en busca de métodos más efectivos. Se asume en la transformación de las ciencias computacionales se asocia una didáctica que apoya a la experiencia educativa.

Con el uso de las TIC, surge el software educativo para la enseñanza de acuerdo con tres paradigmas de trabajo: la enseñanza asistida por computadora (CAI), integrando la inteligencia artificial (ICAI) y como herramienta, los Sistemas de Tutoría Inteligente (ITS) (Self, 1999). En el ámbito del desarrollo del software las clasificaciones centran el interés en las actividades de propuestas de solución del alumno para algoritmos y obtener un óptimo rendimiento académico respecto a conceptos de algoritmos y programación computacional.

Los sistemas de tutoría inteligentes (ITS) son sistemas diseñados para replicar la efectividad de la tutoría humana en herramientas digitales, en el tema del desarrollo de software y algoritmos computacionales, los ITS ayudan a los estudiantes a comprender las características de los lenguajes de programación específicos y desarrollan sus habilidades generales de resolución de problemas.

En esta tesis se propone una metodología para analizar las habilidades de los estudiantes de nuevo ingreso de la UPV relacionadas al desarrollo de algoritmos, posteriormente se evalúa el impacto de los ITS que proporcione soporte en el aprendizaje de algoritmos computacionales y finalmente se plantea una investigación de tipo experimental de diseño cuasiexperimental teniendo la exploración del grupo experimental no aleatorio para analizar el comportamiento de las variables dependientes respecto a los conocimientos adquiridos.

La investigación se centra a los estudiantes de la carrera de ingeniería en tecnologías de la información (ITI) de la Universidad Politécnica de Victoria (UPV) ya que se ha detectado un alto índice de reprobación en asignaturas relacionadas al desarrollo de software y programación según lo muestran los resultados de evaluación generados en el Sistema Integral de Información de la Universidad Politécnica de Victoria (SIIUPV) presentados en la sección planteamiento del problema.

Antecedentes de la Investigación

Las indagaciones que se han realizado en recientes décadas en torno a las ciencias educativas han cambiado la manera que los docentes y estudiantes llevan a cabo las actividades educativas, se pasó de tener un componente donde el alumno solo era un testigo, dentro y fuera del aula de clase, a formar parte de la transformación educativa en donde maestro y estudiante colaboran en conjunto (Rodríguez, 2021).

La indagación en el campo del aprendizaje es extensa por lo que es necesario buscar los factores que influyen en el aprendizaje de la programabilidad de computadoras para obtener procedimientos que ofrezcan de manera efectiva soluciones a los problemas que encuentran los alumnos para mejorar el dominio en el proceso educativo (Quiroga, 2016).

Los procedimientos de aprendizaje forman parte de una colección de metodologías coordinadas con el objetivo de orientar el conocimiento del estudiante con destino a el aprendizaje significativo que se puede observar desde el enfoque constructivista en el cual propone que cada individuo construye su propio aprendizaje, dicho esto se afirma que los métodos de enseñanza y aprendizaje se basan en la relación entre el estudiante y el profesor (Mena, 1997).

La instrucción asistida por computadora (CAI) a través de Sistemas de Tutoría Inteligente (ITS), nace en la década de los 60s, siendo investigada y desarrollada en centros y universidades, tomando mayor auge después de la aparición de las microcomputadoras. Los ITS son programas que enfocan una sesión de enseñanza como un proceso de cooperación entre el tutor y el alumno con objetivos de enseñar y aprender determinados conceptos; su diseño y construcción es la base de la psicología cognitiva, investigación educativa e inteligencia artificial (Quispe, 2014).

Con relación al desarrollo de ITS es un tema relacionado con el nacimiento de la Inteligencia Artificial (IA) y la documentación del uso de algoritmos como las redes bayesianas, redes neuronales o lógica difusa, las cuales son alternativas para la construcción de una herramienta cada vez más inteligente y empática para el apoyo en el aprendizaje del desarrollo de software.

Los ITS comenzaron a desarrollarse en los años 80 y fueron diseñados con la idea de impartir conocimiento con base en alguna forma de inteligencia para guiar al estudiante en el proceso de aprendizaje (Urretavizcaya, 2001; Sancho, 2002). Su propósito es exhibir un comportamiento similar al de un tutor humano, que se adapte al comportamiento del estudiante, identificando la forma en que el mismo resuelve un problema y brindarle ayuda cuando cometa errores. Un tutor inteligente, por lo tanto: “es un sistema de software que utiliza técnicas de

inteligencia artificial (IA) para representar el conocimiento e interactúa con los estudiantes para enseñárselo” (VanLehn, 1988). Wolf (1984) define los ITS como:

Sistemas que modelan la enseñanza, el aprendizaje, la comunicación y el dominio del conocimiento del especialista y el entendimiento del estudiante sobre ese dominio (*p.* 78).

Así mismo Giraffa (1997) delimita los ITS como sistemas que contienen metodologías de IA para generar un entorno que contempla estilos cognitivos de los estudiantes que hacen uso del sistema.

Entre los ITS desarrollados siguiendo las ideas de Carbonell (1970) y con base en paradigmas de programación convencional (no a través del paradigma de agentes inteligentes) se pueden destacar:

- Scholar (Carbonell, 1970).
- Why (Stevens et al., 1977),
- Sophie (Brown et al., 1989),
- Guidon (Clancey et al., 1991),
- West (Burton et al., 1981),
- Buggy (Brown y Burton, 1978),
- Debuggy (Brown et al., 1989),
- Steamer (Stevens et al., 1977),
- Meno (Wolf, 1984),
- Proust (Johnson et al., 1984),
- Sierra (VanLehn, 1988).

Luego, surgen: Andes (Gertner et al., 1998; Gertner y Van Lehn, et al., 2000) en el Pittsburgh Science of Learning Center's LearnLab, que su consorcio con miembros de Carnegie

Mellon University, University of Pittsburgh y Carnegie Learning. Metutor es un tutor de medios fines del Department of Computer Science, U.S. Naval Postgraduate School, Monterey (Galvin, 1994; Rowe, 1998).

El ITS CircSim, fue desarrollado en conjunto por el Departamento de Ciencias de la Computación del Illinois Institute of Technology y el Departamento de Fisiología del Rush College of Medicine. Este tutor es el más avanzado actualmente en su tipo, y se lo utiliza en el Rush College of Medicine para complementar las clases teóricas sobre problemas cardiovasculares (Kim, 1989; Kim, 2000; Cho, 2000; Hume et al. 1992, 1996; Shah, 1997).

Los ITS ayudan a los estudiantes en el aula y fuera de ella a aprender diferentes temas, hace la función del profesor para ofrecer una alternativa de estudio y aprendizaje de los diferentes tópicos.

Planteamiento del Problema

El país mexicano es una de las naciones con alto índice de egresados en distintas áreas de ingeniería, sin embargo existe evidencia que lo aprendido en el aula no esta alineado a lo demandado por la industria. La formación del alumno no es dependiente únicamente de lo adquirido en el salón de clase y por tal motivo es necesario implementar metodologías para mejorar el desenvolvimiento lógico matemático para que tengan un dominio de conocimiento crítico en problemas de la vida cotidiana (Vázquez, 2012).

Como se puede leer en la investigación de Reséndiz (2008):

La formación completa del estudiante de ingeniería no es sólo trabajo de las escuelas, requieren forzosamente de la práctica tutelada por ingenieros maduros para que tenga vivencias personales en todos y cada uno de los procesos intelectuales de la ingeniería. (p.12)

Esta investigación se enfoca en la carrera de ITI de la UPV la cual prepara a los estudiantes para tener una capacidad de análisis, diseño, desarrollo e implementación de sistemas computacionales, además de prepararlos en el ámbito de redes para la administración y configuración de dispositivos de telecomunicaciones, con ésta visión de preparación de la institución es necesario implementar una herramienta que ayude a comprender, asimilar y entender con mayor facilidad los temas que se pretendan enseñar por parte del profesor respecto a la construcción de algoritmos computacionales.

Para poder analizar la población que estará sujeta al estudio de investigación, la Tabla 1 muestra la población estudiantil de agosto del 2020 a diciembre 2021 de la Universidad Politécnica de Victoria (UPV):

Tabla 1.

Población estudiantil de la Universidad Politécnica de Victoria de agosto 2020 a diciembre 2021.

Carrera	Cantidad de alumnos	Cantidad de alumnos
	Agosto 2020	a diciembre 2021
Ing. en Tecnologías de la información	281	305
Ing. Mecatrónica	366	338
Ing. en Tecnologías de Manufactura	191	176
Ing. en Sistemas Automotrices	277	225
Lic. Administración y Gestión Empresarial	146	253
Lic. Administración en Gestión de PyMES	159	82
Profesional Asociado en Ing. Industrial	3	1
Inmersión a los estudios superiores	479	481
Total de alumnos:	1902	1861

Nota: Esta tabla representa la población de alumnos en la Universidad Politécnica de Victoria en los periodos 2020 y 2021 de acuerdo con el Sistema Integral de Información de la UPV (SIUPV 2020).

La población objetivo para aplicar una metodología de un ITS son los alumnos de nuevo ingreso de la carrera de Ing. en Tecnologías de la Información, para agosto del 2020 existe un ingreso de 94 alumnos los cuales tienen diferentes perfiles de bachillerato y representan el 6.11% de la población estudiantil de la universidad de ese periodo.

En la experiencia docente de trabajar con estudiantes matriculados en el primer cuatrimestre en cursos básicos de programación de computadoras, sin duda implica muchas dificultades que han ocurrido debido al comienzo tardío sobre el aprendizaje de algoritmos. Algunas veces los profesores se cuestionan sobre el lenguaje de programación utilizar como punto de partida, pero la elección del lenguaje no es el punto más importante, lo más relevante es el inicio temprano en el aprendizaje algoritmos computacionales el cual puede ser durante la educación primaria.

Los problemas de aprendizaje en el desarrollo de algoritmos computacionales a nivel licenciatura se reflejan en el bajo rendimiento académico, originando con ello la deserción escolar

por no acreditar asignaturas relacionadas con la programación de computadoras. Un elevado número de estudiantes que no cumplen con los requerimientos mínimos de conocimiento para solución de ejercicios algorítmicos, generan un incremento en el índice de reprobación ocasionando en el estudiante un nivel de desmotivación y frustración que truncaría su carrera. La tabla 2 y 3 muestran los detalles de las materias que tienen mayor índice de reprobación en el desarrollo de software lo que provoca un nivel de desinterés en continuar en la carrera de ITI.

En la UPV las materias de algoritmos, lógica computacional y programación estructurada, son materias del primer ciclo de formación de la carrera de Ing. en Tecnologías de la Información, las cuales tienen un alto índice de reprobación al ser materias base en la iniciación en la programación computacional, una causal es que los alumnos de nuevo ingreso provienen de bachilleratos no relacionados con el desarrollo de software, lo que conlleva a que no tienen las bases necesarias para la generación de algoritmos informáticos y por consecuencia, no les sea sencillo aprender a hacer reaccionar una computadora a través de programación.

Para ayudar a los estudiantes durante las actividades de resolución de problemas y tener un buen seguimiento académico de aprendizaje, se pretende diseñar e implementar un modelo de ITS en el cual ayude con estas actividades. Un ITS debe estar equipado con conocimiento de dominio que pueda respaldar los servicios de tutoría apropiados (Viger, P., et al., 2013, p. 364).

Lamentablemente México es un país en vías de desarrollo en donde la introducción de estas tecnologías de ITS en la educación, en comparación de otros países en vías de desarrollo, ha sido lenta y aún más a diferencia de países desarrollados. Todo lo anterior es debido a la falta de una metodología para implementarlas (Rodríguez, 2021).

En el marco de este trabajo está limitado a los estudiantes de la carrera de ITI de la UPV ya que se han detectado asignaturas relacionadas al desarrollo de software y programación que

tienen 20% o más de reprobación lo cual para la UPV es una señal de alerta para atender a los alumnos vulnerables a través de asesorías académicas y con ello mejorar la eficiencia terminal. Los resultados de las evaluaciones son obtenidos del sistema integral de información de la Universidad Politécnica de Victoria (SIUPV), es por ello la motivación de desarrollar una investigación científica que soporte el desarrollo y experimentación de un sistema de tutoría inteligente para que sirva de ayuda en el aprendizaje de algoritmos computacionales.

Con el análisis del comportamiento del rendimiento académico en el periodo enero 2019 – agosto 2020, se identifican las asignaturas con mayor índice de reprobación en el ámbito del desarrollo de software en donde se aprecia claramente que existe un déficit en las estrategias implementadas en el proceso de enseñanza – aprendizaje con la educación tradicional.

La tabla 2 muestra las estadísticas de no acreditación de asignaturas periodo de enero del 2019 a agosto del 2020; en donde es importante señalar que las materias identificadas son las que tienen relación en el desarrollo de software y que a su vez se han impartido una o más ocasiones dentro del periodo señalado.

Tabla 2.

Estadísticas de no acreditación de asignaturas periodo de enero del 2019 a agosto del 2020 de la Ing. en TI de la UPV.

<i>Materia</i>	<i>Enero - Abril 2019</i>	<i>Mayo - Agosto 2019</i>	<i>Septiembre - Diciembre 2019</i>	<i>Enero - Abril 2020</i>	<i>Mayo - Agosto 2020</i>
Programación orientada a objetos					83%
Graficación por computadora avanzada				71%	
Fundamentos de programación orientada a objetos				67%	
Minería de datos aplicada	62%				
Cómputo en dispositivos móviles		44%			43%
Introducción a la graficación por computadora	25%		59%	35%	
Estructura de datos	42%	27%	39%		
Tecnología y aplicaciones web		36%	49%		32%
Matemáticas básicas para computación	33%	34%		44%	
Probabilidad y estadística		34%	24%		
Introducción a la programación			36%	21%	
Negocios electrónicos	39%			20%	32%
Introducción a las tecnologías de información	29%				
Programación		32%	24%		
Base de datos					28%
Programación web	27%	28%			
Integración de tecnologías de la información		25%			
Introducción a las bases de datos	20%				

Nota: Esta tabla representa los resultados de reprobación de las materias relacionadas al desarrollo de software en el periodo comprendido entre enero 2019 y agosto 2020, en ella se aprecia los cuatrimestres en que las asignaturas se impartieron y el nivel de reprobación que arrojaron los resultados en cada uno de ellos, de acuerdo con el Sistema Integral de Información de la UPV (SIIUPV 2020).

En consecuencia con los resultados analizados de manera individual en cada uno de los cuatrimestres de enero 2019 a agosto 2020, se descartan las asignaturas: Integración de tecnologías de la información e Introducción a las bases de datos, ya que solo presentaron alto índice de reprobación en un cuatrimestre, cabe mencionar que las materias de: Programación orientada a

objetos y Graficación por computadora, se encuentran en las mismas circunstancias pero al ser materias con alto nivel de aplicación del conocimiento relacionados al desarrollo de software, es necesario considerarlas para la estadística del nivel de reprobación y aprobación de los estudiantes.

Finalmente, la tabla 3 muestra el concentrado general de materias con mayor índice de reprobación de asignaturas relacionadas al desarrollo de software periodo enero 2019 - agosto 2020.

Tabla 3.

Concentrado general de materias con mayor índice de reprobación del periodo enero 2019 – agosto 2020 del programa académico (PE) de ITI de la UPV.

<i>Materia</i>	<i>% Reprobación</i>
Programación orientada a objetos	83%
Graficación por computadora avanzada	71%
Fundamentos de programación orientada a objetos	67%
Minería de datos aplicada	62%
Cómputo en dispositivos móviles	47%
Introducción a la graficación por computadora	46%
Estructura de datos	39%
Tecnología y aplicaciones web	37%
Matemáticas básicas para computación	36%
Probabilidad y estadística	32%
Introducción a la programación	31%
Negocios electrónicos	31%
Introducción a las tecnologías de información	29%
Programación	29%
Base de datos	28%
Programación web	28%

Nota: Esta tabla representa el concentrado general de materias relacionadas al desarrollo de software con mayor índice de reprobación entre el periodo enero 2019 – agosto 2020 en el PE de ITI de la UPV, de acuerdo con el Sistema Integral de Información de la UPV (SIIUPV 2020).

Con los hallazgos de los altos índices de reprobación en las materias relacionadas en el desarrollo de software y algoritmos computacionales, se hace evidente la falta de estrategias educativas y tecnológicas para solucionar la problemática respecto a la carencia de aprendizajes en el ámbito del desarrollo de software a través de algoritmos computacionales y con ello aumentar la eficiencia terminal de la carrera de Ingeniería en Tecnologías de la Información. En esta investigación se pretende implementar una estrategia de innovación educativa que propicie una mejora para el aprendizaje en el desarrollo de software. Con base a lo anterior se formulan las siguientes preguntas de investigación.

Preguntas de Investigación

Pregunta General

¿Cuál es el impacto de la implementación de un Sistema de Tutoría Inteligente (ITS) de apoyo en el aprendizaje de algoritmos computaciones en los estudiantes de primer cuatrimestre del PE de ITI de UPV?

Preguntas Específicas

- a) ¿Cuáles son las habilidades de los alumnos del primer cuatrimestre en desarrollo de algoritmos computacionales?
- b) ¿Cuál es el nivel de conocimiento de los estudiantes que hacen uso de un ITS respecto a la construcción de algoritmos?

Implementar estrategias innovadoras que contribuyan al enriquecimiento del proceso enseñanza – aprendizaje colaboran en la generación del conocimiento y mejora la educación a nivel Internacional, repercutiendo en la generación de personas competentes en la industria del software. Los ITS son herramientas que proporcionan un nuevo enfoque en el proceso de aprendizaje de los estudiantes. Los avances de las ciencias de la computación y educación apoyan

de manera conjunta en el proceso de aprendizaje significativo, esta investigación aplicará la teoría y metodologías de un ITS en el campo de la IA.

Objetivos

Objetivo General

Analizar el impacto de la implementación de Sistema de Tutoría Inteligente (ITS) bajo la teoría de la educación de la psicología cognitiva que proporcione soporte en el proceso de aprendizaje de algoritmos computacionales en estudiantes del primer ciclo de formación de la carrera de ITI de la UPV, reduciendo la reprobación y bajo aprendizaje en el momento del desarrollo de software.

Objetivos Específicos

- Analizar las habilidades de los estudiantes de nuevo ingreso de la UPV relacionadas en el desarrollo de algoritmos computacionales.
- Implementar un modelo de ITS que integre elementos tecnológicos y educativos para desarrollar conocimientos y habilidades en el desarrollo de algoritmos computacionales, para reducir el índice de reprobación y bajo aprendizaje de algoritmos computacionales
- Evaluar el aprendizaje de los estudiantes respecto a la construcción de algoritmos computacionales a través de diagramas de flujo.

Definición de Variables

Variable independiente: ITS de apoyo en el proceso de aprendizaje de algoritmos computacionales.

Variable dependiente: Aprendizaje de algoritmos computacionales.

Con base a lo anterior, la tabla 4 muestra la definición de variables de investigación.

Tabla 4.
Definición de variables de investigación.

Variable	Tipo	Definición conceptual	Definición operacional
Sistema de tutoría inteligente (ITS) de soporte y apoyo en el proceso de aprendizaje de algoritmos computacionales.	Variable independiente (I)	Software computacional de apoyo y soporte para la construcción de algoritmos computacionales.	Medir el Grado de aprendizaje y aprovechamiento en la construcción de algoritmos computacionales.
Aprendizaje de algoritmos computacionales.	Variable dependiente (D)	Proceso en el cual los estudiantes adquieren o modifican habilidades, destrezas y conocimientos relacionados en la construcción de algoritmos a través de diagramas de flujo.	Evaluación de la manera de implementar soluciones algorítmicas a problemas propuestos a los estudiantes. Caracterización previa de conocimientos (pre – test) y verificación de los logros obtenidos (post – test)

Nota: Esta tabla representa la definición conceptual y operacional de las variables de investigación.

Justificación de la Investigación

En el ámbito del aprendizaje de algoritmos computacionales, la implementación de un ITS en la asignatura de introducción a la programación de la carrera de ITI beneficiará en el aprendizaje de algoritmos computacionales ya que gracias al ITS fungirá como profesor para gestionar ejercicios y ayudarle a construir algoritmos, el uso del sistema apoyará en el aprendizaje y bajar el índice de reprobación que se tiene en la asignatura; y como se afirmó previamente con base a diferentes autores, los ITS ayudan a mejorar el aprendizaje significativo en diferentes áreas de especialización.

Esta investigación tiene un alto sentido de innovación educativa y pertinencia ya que la implementación de una herramienta tecnológica que mida el nivel de conocimiento de los estudiantes en un área en particular, en este caso para conocer el nivel de dominio en el ámbito del

desarrollo de software, es la única existente en una Universidad dependiente de la Dirección General de Universidades Tecnológicas y Politécnicas de México (Rodríguez, 2021).

Con el uso de los tutores inteligentes se puede lograr que la eficiencia terminal se incremente en la UPV, pero más importante que se genere un alto sentido de motivación para que los estudiantes tengan un entorno disponible para aplicar sus aprendizajes.

Capítulo II

Marco teórico

En este apartado se definen los conceptos involucrados en la definición del problema, en donde se justifica la implementación de un ITS para fortalecer el aprendizaje de algoritmos computacionales, para lo cual se definen conceptos del área de las ciencias computacionales y educación, así mismo se incorpora la teoría de la psicología cognitiva como base en el estudio de factores de motivación y análisis de dimensiones del conocimiento a través de la taxonomía de Marzano y Kendall (2007).

Proceso Enseñanza y Aprendizaje

El proceso de enseñanza aprendizaje es una fase en la educación en donde intervienen dos actores principales: los alumnos y profesores, Alvarado *et al.* (2018) lo define como:

El proceso de enseñanza aprendizaje se concibe como el espacio en el cual el principal protagonista es el alumno y el profesor cumple con una función de facilitador de los procesos de aprendizaje. Son los alumnos quienes construyen el conocimiento a partir de leer, de aportar sus experiencias y reflexionar sobre ellas, de intercambiar sus puntos de vista con sus compañeros y el profesor. En este espacio, se pretende que el alumno disfrute el aprendizaje y se comprometa con un aprendizaje de por vida (Alvarado et al., 2018, p. 611).

Hilgard y Bower (1980) define el aprendizaje como "el proceso mediante el cual se origina o se modifica una actividad respondiendo a una situación siempre que los cambios no puedan ser atribuidos al crecimiento o al estado temporal del organismo" (p. 718).

Tomando como referencia las definiciones de Hilgard y Bower (1980) y Feldman (2005), se identifica que el aprendizaje es un proceso que se encuentra en constante movimiento ya que contiene atributos que aportan conocimiento y ofrece a las personas la capacidad de aprender sobre un tema en específico proporcionando un nivel de experiencia. Para ello es indispensable hacer un análisis sobre los estilos aprendizaje para conocer las formas en que se adquiere y procesa la información para construir conocimiento.

Estilos de Aprendizaje

Los estilos de aprendizaje indican cómo los estudiantes perciben y procesan la información para construir su propio conocimiento y proporcionan pautas sobre cómo interactuar con la realidad.

Rodríguez (2002) afirma que los estilos de aprendizaje muestran disparidades cuantitativas en el bienestar mental debido a la integración de aspectos cognoscitivo y afectivo-motivacionales del desempeño propio, ya que definen cómo los aprendices perciben, recuerdan y piensan, cómo son las cosas generalmente.

En investigaciones analizadas en Saarikoski *et al.* (2001) se tienen hallazgos sobre la presentación de información a través de variadas estrategias pedagógicas, se benefician los estilos de aprendizaje asegurando la superación académica.

Los estudiantes aprenden de diferentes maneras; tienen particularidades y estilos individuales de la forma en que observan y procesan la información. Estas preferencias individuales son llamadas estilos de aprendizaje (Grasha y Hruska-Riechmann, 1992).

Varios modelos se han desarrollado para entender y evaluar los estilos individuales de aprendizaje y sus dimensiones, así como una variedad de inventarios de estilos de aprendizaje, la tabla 5 muestra los modelos cognitivos de estilos de aprendizaje y sus características principales:

Tabla 5.

Concentrado general de los modelos cognitivos de estilos de aprendizaje.

Modelo:	Descripción:
Modelo de Kagan (1980)	Destaca la capacidad de reflexión del estudiante frente a su impulsividad.
Modelo de Kolb (1984)	Fundado en la práctica; a) experiencia-concreta; b) conceptualización- abstracción; c) experimentación activa; y d) observación-reflexión.
Modelo de Felder-Silverman (1988)	Habla de cinco estilos de aprendizaje distintos: a). Sensorial frente a intuitivo; b). Visual frente a verbal; c). Inductivo frente a deductivo; d). Activo frente a reflexivo; e). Secuencial frente a global.
Modelo de Myers-Briggs (1980)	Los dos autores plantean un conjunto de estilos de aprendizaje que se combinan en 16 tipos diferentes; los estilos básicos son: a). Extrovertidos/introvertidos; b). Sensoriales/intuitivos; c). Pensativos/emocionales; y d). Juzgadores/receptores.
Modelo de Howard Gardner (1983)	(Teoría de las múltiples inteligencias): El autor estableció originalmente siete estilos: a) inteligencia verbal/lingüística; b) inteligencia lógica/matemática; c) inteligencia visual/espacial; d) inteligencia corporal/kinestésica; e) inteligencia musical/rítmica; f) inteligencia interpersonal; y g) inteligencia intrapersonal.

Nota: Esta tabla representa el concentrado de los modelos cognitivos de estilos de aprendizaje.

La tabla 6 muestra los modelos de estilos de aprendizaje de orientación psicológica:

Tabla 6.

Concentrado general de los modelos de estilos de aprendizaje de orientación psicológica.

Modelo:	Descripción:
Modelo de A. Grasha y S. Hruska-Riechmann (1992)	Plantea la existencia de: a). El estudiante independiente; b). El estudiante dependiente; c). El estudiante competitivo; d). El estudiante colaborativo; e). El estudiante reticente; y f). El estudiante participativo.
Modelo de Margaret Martínez (1999)	Tiene una plena orientación psicológica –sobre emociones e intenciones–, y señala la existencia de varios estilos de aprendizaje: a). Estudiante en transformación; b). Estudiante ejecutor; c). Estudiante conformista; y d). Estudiante que se resiste.

Nota: Esta tabla representa el concentrado de los modelos de estilos de aprendizaje de orientación psicológica.

Los estilos de aprendizaje son fortalezas y preferencias distintivas que permiten trabajar tareas intelectuales y psicológicas de modos específicos y diversos. De acuerdo con Schmeck *et al.* (1991) esto significa que los estudiantes reciben y procesan información de diferentes maneras, preferentemente centrándose en diferentes tipos de información, tiende a operar y percibir la información de manera diferente, y lograr la comprensión a diferentes niveles.

La tabla 7 muestra los factores para el aprendizaje los cuales son importantes a considerar en la implementación de ITS enfocados en el desarrollo de algoritmos computacionales (Parra *et al.*, 2015):

Tabla 7.

Concentrado general de factores para el aprendizaje.

Factores de aprendizaje:	Descripción:
Motivación.	La motivación es el “querer aprender” por tanto es fundamental que el estudiante desee aprender. Aunque la motivación se encuentra limitada por la personalidad y fuerza de voluntad de cada persona.
Experiencia.	La experiencia es el “saber aprender”, ya que el aprendizaje requiere determinadas técnicas básicas tales como: a). Técnicas de comprensión (vocabulario), b). Conceptuales (organizar, seleccionar, etc.), c). Repetitivas (recitar, copiar, etc.) y d). Exploratorias (experimentación).
Inteligencia.	Para poder aprender, el individuo debe estar en condiciones de hacerlo, es decir, tiene que disponer de las capacidades cognitivas para construir los nuevos conocimientos.
Conocimientos previos.	Relacionado con la experiencia, los conocimientos previos pueden ser la base para generar nuevo conocimiento.

Nota: Esta tabla representa el concentrado de factores para el aprendizaje necesarios ser considerados en el desarrollo de algoritmos computacionales según Parra, et al., (2015).

Los estudiantes al hacer sus actividades realizan múltiples operaciones cognitivas que logran que sus mentes se desarrollen fácilmente. La tabla 8 muestra las operaciones cognitivas que el estudiante desarrolla para llevar a cabo diferentes operaciones (Klimenko, 2009):

Tabla 8.

Concentrado general de las operaciones cognitivas que desarrollan los estudiantes.

Operaciones cognitivas:	Descripción:
Recepción de datos.	Supone un reconocimiento y una elaboración semántico- sintáctica de los elementos del mensaje (palabras, iconos, sonido) donde cada sistema simbólico exige la puesta en acción de distintas actividades mentales. Los textos activan las competencias lingüísticas, las imágenes las competencias perceptivas y espaciales, etc.
Compresión de la información.	Recibida por el estudiante, que a partir de sus conocimientos anteriores, sus intereses y sus habilidades cognitivas analizan, organizan y transforman la información recibida para elaborar conocimientos.
Retención a largo plazo.	Retener la información obtenida y los conocimientos asociados a este.
Transferencia.	Del conocimiento a nuevas situaciones para resolver con su concurso las preguntas y problemas que se planteen.

Nota: Esta tabla representa el concentrado de operaciones cognitivas que desarrollan los estudiantes considerados en el desarrollo de algoritmos computacionales según Klimenko, (2009).

Teorías del Aprendizaje

Las Teorías del aprendizaje tienen una función referencial, a modo de señales que indican un recorrido. De hecho, las teorías sobre la educación reflejan un sustrato cultural e ideológico que no escapa del concepto de educación que adoptamos.

Para fines de esta investigación se analizó la teoría de la psicología cognitiva como base para la construcción de un modelo de Sistema de Tutoría Inteligente para el apoyo de algoritmos computacionales, a través de esta teoría se construirá una red neuronal implementada en el módulo de dominio del tutor para que pueda actuar de manera inteligente y soportar al estudiante.

Teoría de la Psicología Cognitiva

Hablar de la psicología cognitiva es hablar de una de las ramas que la psicología ocupa para a través de procesos, el ser humano adquiere sabiduría y es consciente de su medio. Ahora bien, las estrategias de aproximación cognitiva involucran un cambio de percepción de la situación por parte del alumno. Estas estrategias son las más efectivas para reducir el aburrimiento (Pinto, 2003).

Las estrategias de aproximación conductual refieren al intento por cambiar la situación real que genera aburrimiento. Una de las estrategias más comunes consiste en pedirle al profesor tareas más interesantes, proponer alternativas a la tarea encomendada, o simplemente informar la situación. Las estrategias de evitación conductual refieren a las acciones que el estudiante realiza para evadirse del aburrimiento, buscando alguna distracción, como, por ejemplo, charlar con un compañero o jugar con el celular (Pinto, 2003).

De acuerdo con las computadoras son sistemas que logran captar, transformar y elaborar información; la psicología cognitiva ha encontrado en éste un modelo para referirse a la actividad mental (Pinto, 2003).

La teoría de la psicología cognitiva Mayer (1985) la define como un análisis sólido de los procesos y estructuras mentales con el objetivo de entender la conducta del ser humano. Bajo esta premisa el análisis científico hace hincapié que la psicología cognitiva está constituida sobre el desempeño de los sujetos respecto a la comprensión de la conducta humana en contextos reales. En este sentido Mayer (1985) en su investigación indica que la psicología cognitiva tiene como objetivo principal la comprensión de la mente humana y cuyas características es el estudio del comportamiento humano sobre el actuar a sus representaciones mentales. El estudio del comportamiento humano hizo posible la conceptualización de nuevas herramientas y sistemas informáticos.

Características de la Psicología Cognitiva

Dentro de las características más importantes de la teoría educativa de la psicología cognitiva, Riviere (1987) las ordena de la siguiente manera:

- a. Existe un grupo de orientaciones científicas y teóricas, evidencia de conocimiento previo e integrado en el ser humano.
- b. Se redefine el concepto de conducta relacionado a las acciones afectivas del ser humano, el enfoque es hacia la observación de los cambios afectivos y emocionales de las personas, así como de sus características cognitivas y sociales.
- c. Las áreas que sustentan a la teoría son la comunicación, computación, epistemología y lingüística (Gomes-Pena, 1984).

- d. Motiva el sentido de regresión en procesos mentales hacia la psicología experimental permitiendo la generación de memoria, concentración, motivación, imaginación y procesamiento de la información.
- e. Los estudios de la conducta participan no psicólogos, por ejemplo: Piaget, Chomsky, Pribram y Newell.
- f. Su estudio se centra en el estudio de la mente y de todo el ambiente que sucede a su alrededor.
- g. Se ocupa de la actividad de las personas respecto a los aspectos afectivos (interés y propósitos) así como de los aspectos cognitivos (conocimientos, memorización, concentración e imaginación).
- h. Analiza la conducta de las personas al responder a ciertos estímulos para cumplir sus propósitos e intenciones.
- i. Considera que la actividad de las personas no puede ser analizada de manera científica sin verificar las representaciones o constructos mentales que antecedan el comportamiento humano.
- j. Se centra en que la mente es un estado del funcionamiento del cerebro.
- k. Estudia la mente humana para procesar símbolos con la finalidad de interactuar con el exterior.
- l. Afirma que la capacidad de información de la mente humana tiene límites, existiendo así la memoria a corto plazo surgiendo un sistema de capacidad limitada para resguardar datos.

La Psicología Cognitiva en la Investigación

Dentro del rubro de investigación, la teoría de la psicología cognitiva está enfocada a todos los métodos que requieren la memorización, adquisición y uso del saber para extraer, procesar y almacenar la información que concibe el ser humano Borja (2005). La información la teoría la comprende como un estimulante que comienza a ser procesado por las personas. La tabla 9 muestra la representación de la taxonomía de eventos cognitivos.

Tabla 9.
Taxonomía de los eventos cognitivos

Categoría	Función	Componentes
Procesos cognitivos	Recepción, registro y almacenamiento de información	Atención, percepción y memoria.
Procesos cognitivos superiores	Permite la comunicación, búsqueda de solución de problemas	Lenguaje y pensamiento
Funciones cognitivas	Permite la adaptación mediante recursos algorítmicos heurísticos	Inteligencia y creatividad
Procesos cognitivos complejos	Permiten cambios en los desempeños cognitivos	Aprendizaje y desarrollo cognitivo

Nota: La tabla representa la taxonomía de los eventos cognitivos, tomado de Borja (2005).

Los eventos cognitivos básicos existen en los animales y personas y permiten la recepción, almacenamiento y registro de datos, los eventos cognitivos superiores permiten la manipulación de información en forma de símbolos y es una peculiaridad de los seres humanos. Considerando los eventos cognitivos básicos y superiores como un sistema, la inteligencia y la creatividad son elementos que controlan dicho sistema (Borja, 2005).

La inteligencia y la creatividad son funciones que permiten la adaptación al ambiente lo que permite considerarse como funciones cognitivas. Estos elementos son adaptables a los cambios constantes del sistema educativo tomando en cuenta variables internas y externas; para esta adaptación se habla de procesos cognitivos complejos relacionados con el aprendizaje y desarrollo.

Analizando la taxonomía de los eventos cognitivos, primeramente, los seres humanos se encuentran rodeados de diferentes cambios en el medio ambiente, lo que representa los estímulos, para ser atendidos. A ese grupo de estímulos se le denomina caudal de información la cual está representada por luces, sonidos, olores, etc., e incurrir en los registros sensoriales (vista, oído, olfato, etc.) y se almacena en la memoria sensorial la cual al almacenar información se convierte en memoria en corto plazo la cual tiene la facultad de resguardar la información por un periodo corto de tiempo de alrededor de 30 a 60 segundos (Borja, 2005).

Posteriormente los datos se trasladan a una memoria permanente denominada memoria a largo plazo. Esta memoria es el soporte del sistema cognitivo ya que concentra la base del conocimiento para un buen funcionamiento del sistema cuyos elementos son el pensamiento, percepción, comprensión y lenguaje (Borja, 2005).

El pensamiento es el proceso cognitivo superior que se presenta en las personas en diferentes circunstancias específicas:

- a. Cuando se formulan conceptos.
- b. En la resolución de problemas.
- c. Cuando se utiliza el razonamiento.
- d. En la toma de decisiones.

La percepción es un proceso de extracción e interpretación de información que opera a través de patrones almacenados en la memoria a largo plazo para permitir el reconocimiento. La percepción se concibe siguiendo datos de manera secuencial lo que significa que es un procesamiento guiado.

El lenguaje es la capacidad que permite la manipulación y entendimiento de símbolos con el objetivo de comunicar. Los aspectos psicológicos del lenguaje están relacionados con la comprensión y producción de disertaciones (Borja, 2005).

Dicho lo anterior, la psicología cognitiva es la ciencia que brinda soporte en los métodos de adquisición de conocimientos que al ser procesados se convierten en información. Dicha información estimula el proceso del saber el cual se representa a través de diferentes eventos taxonómicos cognitivos que se utilizan como almacenamiento de datos.

Para llevar a cabo la adquisición de información en las personas, el lenguaje proporciona las herramientas de comprensión a través de símbolos con la finalidad de transmitir conocimiento y por ello se analizarán las tendencias de la psicología cognitiva.

Tendencias de la Psicología Cognitiva

Dentro de la educación la teoría de la psicología cognitiva va incrementando su aplicación a partir de distintas fuentes de interés: Las ciencias computacionales, las teorías de la información y la epistemología genética (Mayer, 1985).

En ese orden, a continuación, se describe brevemente las tendencias de la teoría:

1. La evolución de las ciencias computacionales: Los aspectos de las ciencias computacionales han realizado un impacto en la psicología relacionado con el conjunto de ideas de procesos simbólicos y un medio para expresar teorías a través del uso del sistema computacional. A través de los recursos informáticos se permite el almacenamiento de información, desarrolla la imaginación y aporta elementos de recuperación de datos.

2. Teorías de la información: La codificación de datos es un conjunto de reglas y procesos en las cuales los mensajes o señales se transforman en una representación de información. El transporte de datos se lleva a cabo a través de un canal de comunicación vinculado con la eficiencia de la transmisión y el ruido influye en la detección de señales. A partir de ello, las personas se han definido como un canal activo de comunicación con capacidad limitada.
3. Epistemología genética: Estudio del sujeto epistémico. Dimensión cognitiva del comportamiento, está relacionada con el procesamiento de la información del ser humano.

Respecto a la orientación funcional de la teoría, Vigotsky fundador de la línea de la psicología cognitiva, tuvo principal interés en el estudio de la actividad como entidad operadora de medios, haciendo énfasis en lo siguiente:

- a. Análisis del proceso, no del objeto. Todo proceso psicológico sufre cambios, bajo esa premisa resulta factible dividir el desarrollo en capas. Se hace hincapié en el análisis del proceso de cada una de las capas.
- b. Enfoque en la definición mayor que la representación. La explicación descubre relaciones dinámicas que causan a los fenómenos.
- c. Preferencia por la psicología de las funciones superiores. El interés está centrado en los conceptos científicos y el pensamiento con el lenguaje.

Vigotsky fue un personaje mentor de la teoría de la psicología cognitiva, dentro de sus principales contribuciones se distinguen:

1. Aportaciones constructivistas.
2. Concepción de los procesos mentales superiores como el pensamiento y lenguaje.

3. Concepción que el conocimiento es el resultado de la interoperabilidad entre el sujeto que conoce y el objeto del saber.
4. Relacionar el lenguaje con el desarrollo cognitivo general del ser humano.
5. Concepción dinámica de los procesos psicológicos.
6. Preocupación por generar un estímulo de motivación en las personas para aprender.

Justificación de la Teoría Utilizada en la Investigación

Esta investigación estará sustentada bajo la teoría de la educación psicología cognitiva la cual se basa en los procesos mediante los cuales el ser humano adquiere conocimientos. Derivado que uno de sus principales objetivos es el estudio de la percepción, memoria, razonamiento y resolución de problemas.

Considerando la división del conocimiento que hace la teoría de psicología cognitiva para la división en etapas del desarrollo cognitivo, es la indicada para la implementación de un ITS en la UPV ya que gracias a sus aportaciones teóricas se puede identificar el conocimiento que obtienen los alumnos a través del entorno que lo rodea categorizando el conocimiento con base a lo indicado en la taxonomía de Marzano y Kendall (2017) lo que permite identificar el avance de aprendizaje de los estudiantes a través de los niveles de dominio en los que se divide.

En el caso del aprendizaje del desarrollo de software un punto importante a considerar es asegurarse que alumnos tengan los recursos necesarios para construir algoritmos como lo son: las plataformas de apoyo, las asesorías y tutorías académicas. Por otra parte, sobresale en esta teoría que cuando un alumno se encuentra importuno y desmotivado al momento de estudiar por falta de entendimiento de los tópicos, llega el momento en que recuerde la importancia de prestar atención para aprobar una evaluación o buscar la aplicabilidad o interés de aquello que desea aprender.

Aprender a construir algoritmos requiere además de su entorno, un nivel de motivación y comprensión para generar software, los grandes beneficios que ello conlleva y con ello hacer entender a los estudiantes que no solo se aprueba una materia, si no más bien, se aprende de ella y se aplica en la vida profesional.

Al hacer uso de la teoría cognitiva para la construcción de un ITS en el aprendizaje de algoritmos computacionales a nivel superior se puede evadir el aburrimiento y frustración al momento de desarrollar software ya que una de las estrategias del ITS es motivar al estudiante mediante la resolución de ejercicios de complejidad gradual lo que permitirá un avance significativo al momento de aprender.

Sistemas de Tutoría Inteligente (ITS)

Contexto de los ITS

En el campo de la informática existen diferentes herramientas tecnológicas que hacen uso de la IA y de las TICs con el objetivo de mejorar el aprendizaje en el aula, dentro las mas destacadas sobresalen los ITS los cuales a través de la IA puede emular a un profesor para determinar que, cuando y como enseñar, imitando a un profesor real (Rodríguez, 2021).

Los ITS son sistemas computacionales que están diseñados para integrar técnicas de IA para proporcionar al tutor cuál conocimiento debe enseñar y cómo debe hacerlo, esto produce un comportamiento computacional parecido al ser humano al momento de enseñar.

Ovalle (2007), define al sistema tutorial inteligente como una herramienta cognitiva computarizada que busca mejorar el proceso de enseñanza - aprendizaje de los estudiantes a partir de la interacción entre varios módulos: dominio, pedagógico y estudiante.

Los sistemas de tutoría inteligente (ITS), como lo asevera Suarez Granados, et al. (2016) en su investigación, son sistemas diseñados con la idea de impartir conocimiento para guiar al estudiante en el proceso de aprendizaje a través de una forma de inteligencia.

En el diseño de los Sistemas de Tutoría Inteligente interviene la Inteligencia Artificial (AI) la cual es la disciplina que estudia la creación y diseño de entidades capaces de razonar por si mismas (Cataldi & Lage, 2009), la visión por generar formas que imiten el comportamiento de la inteligencia humana y la creciente demanda del sistema educativo para medir el rendimiento del estudiante, la AI de la mano de las tecnologías de la información y la comunicación (TIC) crearon la Instrucción Asistida por Computadora (CAI), el cual es un sistema de instrucciones computacionales y posee dos características importantes: el estudiante y la computadora que es un vehículo para las instrucciones, no un método de enseñanza (Huapaya, 2009). Más tarde este sistema evolucionó a los Sistemas Tutorías inteligentes.

Un sistema tutorial inteligente (ITS), se define como: sistemas computacionales diseñados para impartir instrucción y apoyar inteligentemente los procesos de enseñanza aprendizaje mediante la interacción con el alumno (Arias, Jiménez, & Ovalle, 2009).

Desde esta perspectiva un ITS es un tipo de ambiente interactivo que está diseñado para el aprendizaje individual y se distingue de los otros tipos por su capacidad para modelar el estado cognitivo del usuario, permitiendo brindar consejos sensibles al contexto y retroalimentar en todos los pasos de un proceso de aprendizaje (Graesser, Chipman, Haynes, & Olney, 2005). Su objetivo principal, además de dominar un área de conocimiento en específico, es desarrollar una metodología que se adapte al alumno e interactúe dinámicamente con el mismo. (Hernández, et al., 2015).

Modelos de Sistemas de Tutoría Inteligente

En la recopilación de información relacionada con los Sistemas de Tutoría Inteligente se plantean diferentes metodologías para el diseño, construcción y selección adecuada de los modelos de la arquitectura de los ITS (Sánchez, Cabrera y Martínez, 2015).

De acuerdo con Arias, Jiménez y Ovalle (2009) proponen un modelo de planificación instruccional en sistemas tutoriales inteligentes que se basa en el nivel de conocimientos de estudiantes, en la teoría de la planificación de la Inteligencia Artificial (AI) y en la estructura de cursos aplicada en el en el Sistema Tutorial Inteligente (Cursos Inteligentes Adaptativos). Este modelo admite generar planes de actividades adaptados a las características de los estudiantes, además a medida que los estudiantes avanzan en un curso, van adquiriendo nuevo conocimiento que les permite a su vez habilitar nuevos temas. De esta manera se puede lograr la planificación instruccional basada en el secuenciamiento del currículo (Sánchez, et al., 2015).

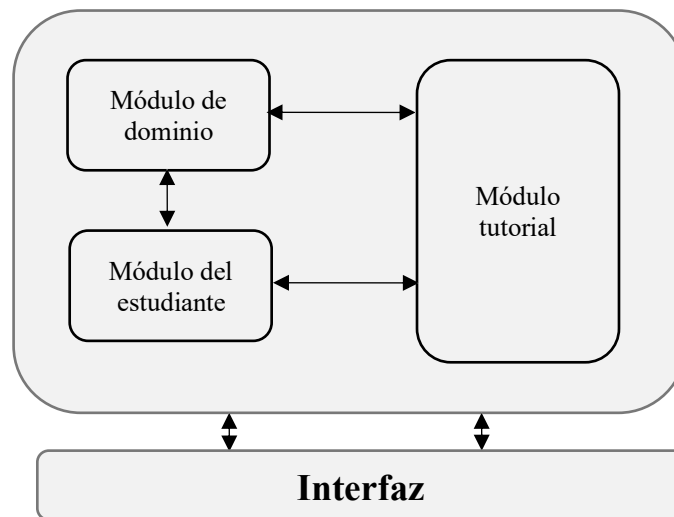
En la investigación de Ovalle y Jiménez (2006) se muestra una investigación en donde se pone en evidencia un ambiente inteligente distribuido de aprendizaje donde se integran los ITS y los ambientes colaborativos de aprendizaje, se implementan técnicas de IA específicamente en el área del aprendizaje automático conocido como razonamiento basado en casos logrando una educación individualizada y colaborativa (Sánchez, et al., 2015).

En Cataldi et al. (2009) se expone una arquitectura básica para los ITS, donde se muestra el módulo del tutor. Esta arquitectura presenta una forma más efectiva de comunicación entre el usuario y el sistema, ya que considera el estilo de aprendizaje del estudiante y propone integrar los agentes inteligentes para el desarrollo de los sistemas tutores inteligentes.

En la figura 1 se muestra la arquitectura de un ITS en la cual se puede observar cada una de las partes que lo conforman: 1) El módulo de dominio o experto representa el conocimiento, es

aquí donde intervienen metodologías de IA como las redes neuronales de clasificación y aprendizaje profundo a través del uso herramientas como lo son Tensorflow, 2) El módulo del estudiante en donde su principal función es capturar el aprendizaje desde el módulo de dominio; 3) El módulo tutorial es quien contiene las estrategias, metodologías e instrucciones que se ajustan a las necesidades del estudiante sin la intervención del ser humano. El objetivo de este elemento es reducir al mínimo la diferencia del conocimiento entre el experto y estudiante (Sánchez, et al., 2015). El último módulo del ITS se denomina 4) Módulo de entorno; el cual es el gestor de interacción entre todos los componentes del sistema y controla la interfaz de usuario entre la computadora y el ser humano mostrando herramientas de usabilidad y experiencia de usuario para el uso adecuado del ITS (Sánchez, et al., 2015)

Figura 1.
Arquitectura de un ITS.



Nota: El gráfico representa la arquitectura de un Sistema de Tutoría Inteligente (ITS). Tomado de *Salgueiro, et. al., 2005.*

Otro modelo de ITS a resaltar es el expuesto por Guzmán y Conejo (2004) en el cual se da a conocer un modelo de evaluación cognitiva para ser aplicado a un ITS como módulo de diagnóstico del conocimiento del alumno. Este modelo integra test adaptativos computarizados y

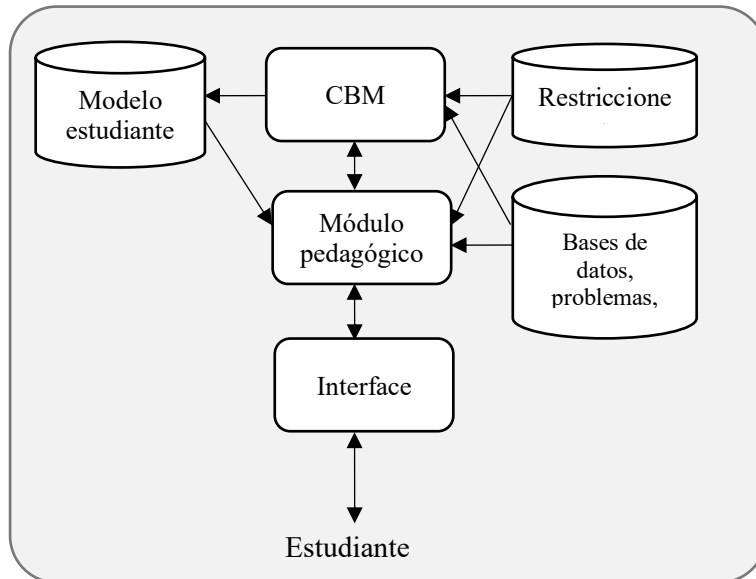
un modelo de respuesta basado en la teoría de respuesta al ítem, esto con el fin de estimar el conocimiento del alumno para decidir que preguntas se mostrarán en la prueba (Sánchez, et al., 2015). Este modelo se puede utilizar para aprender cualquier tipo de temáticas en diferentes áreas, como lo son las matemáticas y la programación computacional.

Sistemas de Tutoría Inteligente en el contexto Educativo

Existe la relación entre proyectos y documentos que aplican ITS en el ámbito educativo identificando su aplicación en diferentes áreas del conocimiento. En la investigación de Rodríguez, Castillo y Lira (2013) se muestra un desarrollo de un ITS con características reactivas y el componente interactivo de comunicación para el aprendizaje de las matemáticas, dicho sistema se denominó Malvi y los investigadores lo utilizaron para identificar el nivel de aprendizaje que se encontraban algunos estudiantes de licenciaturas, luego de hacer este estudio se identificó que los estudiantes de licenciatura en educación para la salud y la licenciatura en comercio internacional manifiestan un conocimiento general muy bajo de matemáticas básicas y los supera los alumnos de ingeniería con un 13% de su desempeño (Sánchez, et al., 2015).

Así mismo Mitrovic, et al., (2013) plantea en su investigación un ITS denominado SQL – TUTOR el cual sirve de guía en la realización de consultas a los estudiantes, ayudando a mejorar el rendimiento en el área de base de datos. El sistema ofrece un problema al estudiante el cual debe ser solucionado para posteriormente sea enviado para su análisis, una vez analizado, el ITS proporciona una respuesta donde se indica si dicha consulta se ha realizado correctamente o no. La figura 2 muestra la arquitectura del SQL-TUTOR.

Figura 2.
Arquitectura SQL-TUTOR.



Nota: El gráfico representa la arquitectura del ITS denominado “SQL-TUTOR” propuesto por *Mitrovic y Ohlsson*, 1999.

El sistema SQL-TUTOR consiste en una interfaz de interacción con el estudiante, una base de conocimiento para el experto, un modelo de estudiante que determina el contenido de las acciones pedagógicas (Mitrovic & Ohlsson, 1999).

Por otro lado, Arevalillo-Herráez, Arnau, y Marco- Giménez (2012) describen un sistema tutorial inteligente que se centra en la etapa de traducción de problemas verbales en notación simbólica puesto que uno de los pasos más difíciles de aprender álgebra es la traducción de un problema verbal a notación algebraica. El ITS está basado en un lenguaje de descripción en hipergrafos capaz de representar simultáneamente soluciones algebraicas a un problema dado, realiza el seguimiento de cada una de las acciones del alumno y a su vez construye un modelo del estudiante. Es un sistema en el cual contiene una interfaz de usuario capaz de al alumno a seguir la secuencia de pasos que describe el método cartesiano. El sistema ha sido evaluado en un entorno experimental al que se le asignaron varios ejercicios algebraicos para resolver, el grupo de

estudiantes con ayuda del ITS logró, luego de varias sesiones de uso, finalizar los ejercicios de manera correcta ya que el sistema permite al alumno tomar alguna ruta válida que produce una solución correcta. La experimentación del ITS basado en hipergrafos arroja como resultados la mejora de las competencias del alumno en la resolución de problemas (Sánchez, et al., 2015).

En el desarrollo de esta investigación, se analizaron los diferentes ITS que han sido aplicados y experimentados en la educación y se hace evidente los buenos resultados que estos han generado, así mismo se encontró que los ITS han sido utilizados en otras áreas de la ciencia sin perder su filosofía. Por ejemplo, el trabajo investigativo de Cabada, Barrón, y Olivares (2014) evidencia un ITS para la enseñanza / aprendizaje de la matemática, esto con el fin de identificar el estado emocional del estudiante. El software lleva a cabo un análisis emocional a través de una red neuronal artificial, donde se tiene en cuenta el tiempo y errores dentro de un problema matemático e imágenes de reconocimiento facial del estudiante. Se concluyó en el ITS que el reconocimiento de emociones de los alumnos juega un papel importante en el proceso de aprendizaje.

En la investigación de Oulhaci, Tranvouez, Espinasse y Fournier (2013) se propone un ITS denominado SIMFOR el cual mediante la gamificación entrena a no profesionales sobre la gestión de riesgos. El juego consta de módulos de creación de escenarios que simulan un acontecimiento con factor de riesgo para la interacción con el usuario. Es en este proceso formativo donde el usuario adquiere conocimiento sobre el riesgo y como debe enfrentarlo.

Se asume que el aprender a programar computadoras, el estudiante desarrolla ciertas habilidades, pero los profesores encuentran que no todos los estudiantes las desarrollan con la misma eficiencia al tener deficiencias en aptitud y actitud (Quiroga, 2016).

La enseñanza de programación dentro de la Universidad Politécnica de Victoria toma el grupo de estudiantes como una entidad dinámica, es decir, pasa de un tema tras otro, pero es en

ese trayecto que el estudiante no alcanza el dominio del tema que le de las herramientas suficientes para abordar el siguiente, lo que causa que progresivamente el alumno se vea superado por las actividades de la materia y termine por darse por vencido y alejarse de la programación.

En el área de la programación computacional el alumno debe desarrollar habilidades de inteligencia emocional para generar algoritmos con el menor número de procedimientos para la resolución de problemas, esto se logra gracias a la lógica que se implemente en cada uno de los algoritmos (Quiroga, 2016)

Ejemplos de Sistemas de Tutoría Inteligente

Como lo comenta Zatarain (2018), los ITS son sistemas de instrucción basados en computadoras que cuentan con modelos de contenidos que especifican qué enseñar, así como las estrategias de enseñanza que describen cómo se debe enseñar (Sharma, Ghorpade, Sahni y Saluja, 2014). Algunos ejemplos de ITS de programación son ITS PHP (Weragama, 2013) y CSTutor (Buchanan y LaViola Jr, 2012). ITS PHP se desarrolló para enseñar a los estudiantes a programar en el lenguaje de PHP. Por otra parte, CSTutor ayuda en el aprendizaje del lenguaje C#, incorporando un modelo de aprendizaje donde las actividades de enseñanza y aprendizaje se diseñan en un entorno de la vida real.

Como lo indica Machin (2018) en su investigación, los ITS para la programación orientada a objetos se integran técnicas y una base de habilidades para el desarrollo de software. Existen algunos principios que sustentan los ITS y que la autora hace referencia en su investigación:

Principio 1: La estandarización para hacer mas eficiente la reutilización de recursos.

Principio 2: La interoperabilidad para permitir que las aplicaciones sean compatibles y reutilizables sin importar las plataformas tecnológicas.

Principio 3: La flexibilidad para contener componentes funcionales genéricos capaces de adaptarse a las aplicaciones existentes.

Principio 4: La congruencia en el desarrollo.

Principio 5: La independencia de características de desarrollo.

En la construcción de los ITS destaca el módulo de dominio, es donde el tutor actúa como un experto en un área determinada para poder interactuar con el módulo del estudiante el cuál es en este último donde se interactúa con el software inteligente para evaluar y clasificar el aprendizaje con base a resultados de rúbricas de evaluación, el ITS alcanza un nivel de experto gracias a las redes neuronales las cuales son indispensables para la clasificación del aprendizaje de estudiante a partir de una colección de datos (datasets) los cuales se analizan a partir de la construcción de la red de neuronal (Machin, 2018).

Los ITS están contruidos con base AI la cual es una rama que desarrolla hardware y software con inteligencia parecida a la del ser humano.

Desde hace ya varias décadas se ha introducido el término de AI como la capacidad de que una máquina emule el comportamiento humano como técnica para la toma de decisiones. La AI fue vista por diversos autores como algo imposible, algo que se encontraba lejos de la realidad, nos relataban que en un futuro se tendría la posibilidad de convivir con máquinas las cuales iban a poseer una inteligencia similar o superior a la de los seres humanos. Hoy en día existe la aplicación de la IA en sistemas o hardware que se encuentran presentes en la vida cotidiana que si bien es cierto están muy lejos de llegar a tener todas las características del razonamiento humano, pero si emulan alguna característica en específico que puede ayudar con la toma de decisiones (Machin, 2018).

Según Hashimoto, Rosman, Rus y Meireles (2018), la AI puede definirse en términos generales como el estudio de los algoritmos que confieren a las máquinas la capacidad de razonar y realizar funciones cognitivas como la resolución de problemas.

Nilsson (2014) a su vez define la inteligencia artificial como una subparte de la ciencia de la informática, preocupada por cómo dar a las computadoras la sofisticación para actuar de manera inteligente, y hacerlo en ámbitos cada vez más amplios.

La IA es un campo que se encuentra presente en las tecnologías informáticas con el fin de brindar la posibilidad de que una máquina tome decisiones o haga predicciones con base a una fuente de datos.

Metodologías para el Desarrollo de ITS

Al igual que existen varias arquitecturas para desarrollar un sistema tutor inteligente son múltiples las metodologías que se emplean con el mismo objetivo, una de ellas es elaborada por Salgueiro, Costa, Cataldi, Lage, García-Martínez (2005), la cual propone un nuevo enfoque, sin alejarse de la estructura clásica, hace énfasis sobre todo en el módulo del tutor, aboga sobre la idea de que mientras más conocimiento instruccional se posea, mejor explicado será el contenido y las actividades, traduciéndose esto en mayor conocimiento para el alumno.

Como contrapartida, Cataldi y Lage (2009), proponen una metodología con un enfoque más hacia el módulo del alumno. Reconocen que un ITS que posea un módulo del alumno muy detallado garantizaría en su totalidad el calificativo ‘inteligente’ del sistema, pues se acoplaría aún más a las particularidades de sus alumnos.

Por otra parte, existen metodologías que persiguen la integración de un sistema gestor de aprendizaje (SGA) con un ITS (Tarongí, 2010), dotando así a estos últimos con un enfoque basado en la Web.

En la integración de los componentes del ITS se establecen las relaciones indicando cuál es la equivalencia entre los datos de las distintas tablas de la base de datos.

La primera relación que se encuentra en el módulo del alumno es el estilo de aprendizaje que determina el método pedagógico del módulo del tutor. A partir de los distintos tipos de métodos pedagógicos se han establecido las equivalencias con las diferentes dimensiones de los estilos de aprendizaje, se marca el estilo que predomina en cada método, según las características del alumno, según sus carencias; estas se toman en cuenta con el objetivo de instruirlo de la mejor forma posible.

Otras de las relaciones que existen entre el módulo del alumno y el módulo del tutor son la base para definir las actividades que realizará el alumno, los cuales vienen condicionados por el componente estilo de aprendizaje y están marcados por el componente nivel de conocimiento. El componente estilo de aprendizaje condiciona el formato en que se presentarán los objetos, mientras que el nivel de conocimiento marca la complejidad de estos.

Por otra parte, la interfaz viene modelada por el estilo de aprendizaje, aunque esta relación no es muy vinculante, ya que el alumno tiene libertad para modificar y adaptar el entorno de trabajo o interfaz a su gusto, se guardan los cambios que realice y se mantienen para las siguientes sesiones de trabajo que establezca.

A modo de inicialización de la aplicación ITS se pueden tomar en consideración las pautas establecidas donde se han analizado las características que presentan los diferentes tipos de herramientas y su relación con las dimensiones del estilo de aprendizaje.

El objetivo principal del Sistema de Tutoría Inteligente es identificar el nivel de aprendizaje sobre el área del desarrollo del software a través de un algoritmo de inteligencia artificial el cual permitirá detectar de manera temprana el nivel de dominio de los temas relacionados con la programación computacional para ofrecer una alternativa de solución a manera de ejercicios para fortalecer el aprendizaje significativo.

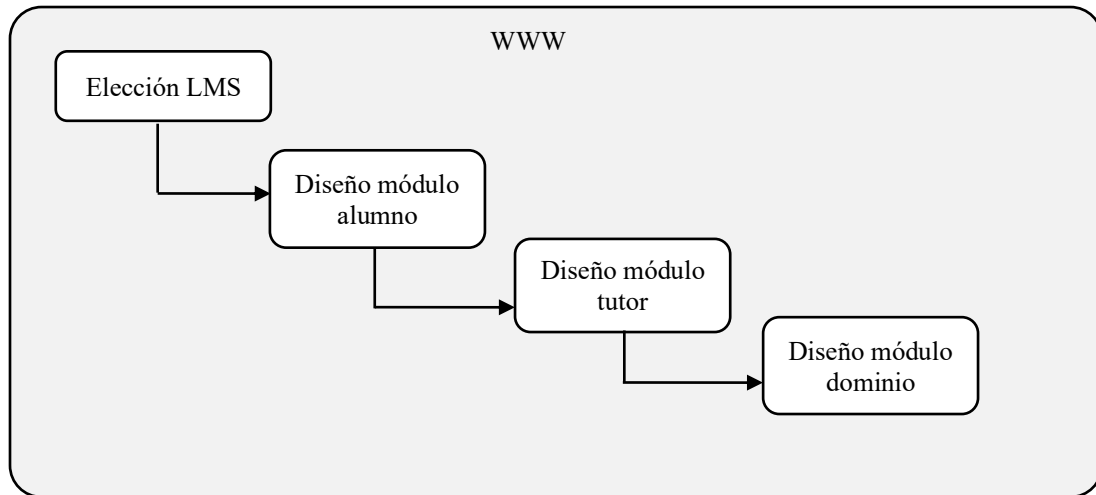
El sistema permitirá al estudiante ofrecer soluciones para aprender a desarrollar software y fortalecer así los conocimientos sobre programación computacional.

Existen diferentes librerías Inteligencia Artificial que permiten el desarrollo de algoritmos para la creación de Sistemas de Tutores Inteligentes, Tensorflow es una biblioteca de código abierto para aprendizaje automático a través de un rango de tareas, y desarrollado por Google para satisfacer sus necesidades de sistemas capaces de construir y entrenar redes neuronales para detectar y descifrar patrones y correlaciones, análogos al aprendizaje y razonamiento usados por los humanos.

El uso de herramientas de inteligencia artificial como lo es el caso de Tensorflow para la construcción de Sistemas de Tutoría Inteligente, admite generar enseñanzas en tiempo real permitiendo la discusión libre entre las herramientas tecnológicas y el alumno.

En la figura 3 se presenta gráficamente la metodología que se sigue como desarrollo de un ITS basado en la web consta de 4 etapas.

Figura 3.
Metodología para la construcción de un ITS.



Nota: El gráfico representa la Metodología para la construcción de un ITS basado en la web.

Selección de la Plataforma Gestor de Aprendizaje (SGA)

Existen diferentes plataformas SGA que apoyan el proceso de enseñanza – aprendizaje, en el desarrollo de esta investigación se presenta un modelo de ITS para que sea aplicado al grupo experimental en el apoyo en el aprendizaje de algoritmos computacionales, se utilizó el LMS MOODLE por ser de código abierto y completamente modificable, es una opción para integrar el ITS por su estructura modular, lo cual ofrece la opción de agregar o eliminar código sin afectar la operación del LMS, así mismo integra gestores de base de datos diversos como lo son PostgreSQL y MySQL que son de fácil instalación.

En este apartado se establecen las relaciones entre los principales componentes que forman parte del ITS; la primera relación que se encuentra en el módulo del alumno es el estilo de aprendizaje que determina el método pedagógico del módulo de tutor. A partir de los distintos tipos de métodos pedagógicos se han establecido las equivalencias con las diferentes dimensiones de los estilos de aprendizaje, se marca el estilo que predomina en cada método, según las

características del alumno, según sus carencias; estas se toman en cuenta con el objetivo de instruirlo de la mejor forma posible (Suárez, et al., 2016).

Diseño de Módulo del Alumno

El diseño del módulo del alumno sucede cuando se analiza las debilidades y fortalezas del alumno en el momento de asimilar los contenidos de una asignatura. Proporciona información que puede ser retroalimentada a través del ITS y tiene estrecha vinculación con los contenidos y el método pedagógico del módulo del tutor, con los que se instruyen al estudiante, así como con la interfaz de usuario en la que se presentan los contenidos por el módulo de dominio (Suárez, et. al., 2016).

Para tener una idea del nivel de conocimiento que posee el alumno previo a la interacción con el ITS, se analiza una minería de datos sobre el promedio de las notas de la asignatura de **Introducción a la programación computacional** que toman actualmente en la Universidad Politécnica de Victoria; así mismo se presenta un pre-test para que el sistema analice su nivel de conocimiento en el desarrollo de algoritmos. A través de una red neuronal de entrenamiento profundo construida con herramientas de Inteligencia Artificial, se clasifica al alumno según el conjunto de términos lingüísticos en competente, básico avanzado, básico y no competente.

Diseño de Módulo de Tutor

El módulo del tutor es el motor de ejecución del sistema adaptativo, pues codifica los métodos de enseñanza que son apropiados para el dominio objetivo y el estudiante. El mismo selecciona la intervención educativa más adecuada en función del conocimiento y estilos de aprendizaje de los alumnos (Suárez, et. al., 2016).

El diseño del módulo del tutor consta de codificar los métodos de enseñanza que son apropiados para el dominio objetivo de los estudiantes, selecciona la intervención educativa

adecuada en función del conocimiento y estilos de aprendizaje de los estudiantes. Las características del alumno determinan el formato en que desean que se les presente los materiales junto con la complejidad de los contenidos que debe aprender; esto permite planificar una ruta de aprendizaje a través de una red neuronal para que se presente una propuesta de trabajo en el módulo del alumno.

Diseño de Módulo de Dominio

Finalmente, en el módulo de dominio consiste en integrar una red neuronal de aprendizaje profundo a través de librerías de Inteligencia Artificial que permitirá comparar las acciones y elecciones del alumno en un sistema experto con el objetivo de evaluar lo que el usuario conoce y desconoce.

El módulo del dominio está compuesto por la ruta de aprendizaje que viene definida por el módulo del tutor. Esta ruta de aprendizaje contiene la secuenciación a aplicar para definir y estructurar el tema, que se debe materializar en una serie de actividades (tareas, cuestionarios, encuestas, etc.) que vienen diseñados y condicionados por el curso y que se ofrecen al alumno a través de una interfaz de usuario, adaptada a las características del estudiante (Suárez, et. al., 2016).

La interacción del alumno con las actividades en la interfaz produce una serie de informes, que se almacenan en una base datos para verificar el progreso del alumno. Dicho análisis se obtiene del promedio de las calificaciones obtenidas producto de la interacción del estudiante con el ITS en la plataforma web MOODLE, con ello a través de una red neuronal se logra clasificar el aprendizaje del alumno en un determinado tema.

Rutas de Aprendizaje en el Diseño de Algoritmos Computacionales

Para la construcción de un ITS de apoyo en el aprendizaje de algoritmos computacionales a través de diagramas de flujo, se identifican en la literatura las habilidades cognitivas que el estudiante debe conocer respecto a la lógica computacional para la solución de problemas informáticos, dichas habilidades se describen a continuación.

Habilidades para el Desarrollo de Algoritmos

Álvarez (1999) define la habilidad como:

La dimensión del contenido que muestra el comportamiento del hombre en una rama del saber propio de la cultura de la humanidad. Es, desde el punto de vista psicológico, el sistema de acciones y operaciones dominado por el sujeto que responde a un objetivo.

Específicamente la habilidad “implementar algoritmos” (p.5).

Proviene de la habilidad matemática “algorítmica” y se adecua a la informática como disciplina. Un algoritmo como entidad matemática abstracta, independiente de restricciones tecnológicas, puede estudiarse en dos categorías: el análisis de algoritmos y el diseño de algoritmos. Para el análisis y el diseño de un algoritmo pueden emplearse múltiples vías de representación. Sin embargo, la implementación, como aplicación o ejecución del procedimiento, *requiere necesariamente* en el campo de las Ciencias Informáticas la vinculación a un lenguaje de programación con soporte tecnológico.

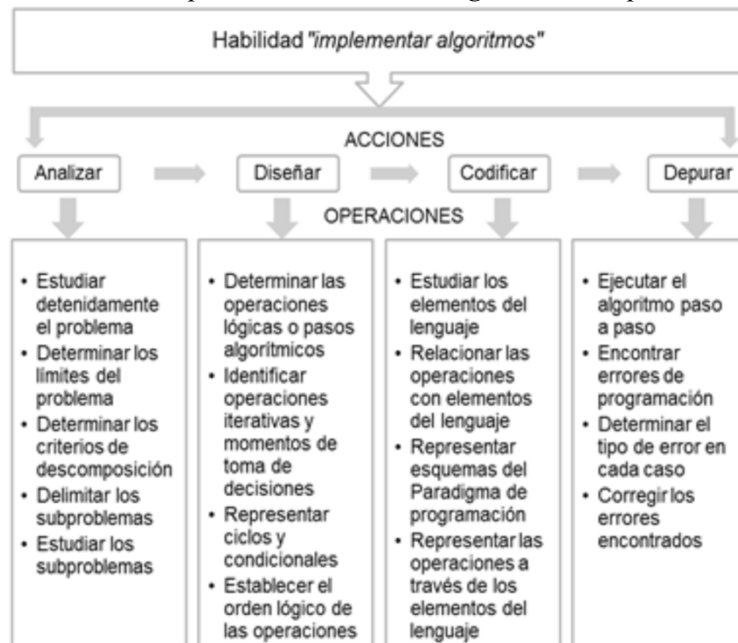
La formación y *desarrollo de cualquier habilidad*, mediante actividades o tareas docentes, requiere conocer su estructura interna, especialmente el sistema de acciones necesarias para regular dicha actividad. De acuerdo con Barreras et. al (2003), toda habilidad tiene como componentes estructurales a:

1. Los conocimientos, como base gnoseológica.
2. Las acciones y operaciones, como componentes ejecutores.
3. Los motivos y objetivos, como componentes inductores.

La figura 4 muestra los componentes relacionados a las habilidades que se deben tener para el desarrollo de algoritmos propuestos por De la Caridad Saez, et.al., (2015)

Figura 4.

Habilidades ejecutores necesarias para el desarrollo de algoritmos computacionales.



Nota: El gráfico representa las habilidades que se requieren para el desarrollo de algoritmos. Tomado por *De la Caridad Saez, et.al., (2015).*

Las habilidades que puede destacar dentro de un estudiante en el momento de desarrollar algoritmos computacionales son:

- Establecer un plan detallado de solución del problema.
- Dominio de un lenguaje de programación.
- Hacer pruebas y depuración de código.

De forma específica, las materias de desarrollo de software dentro de la UPV buscan que el estudiante sea capaz de abstraer de la realidad los elementos de un problema que impliquen la

automatización de las tareas para el manejo de información, analizar, moldear y describir componentes necesarios para la solución de problemas con la computadora.

Para el apoyo del aprendizaje dentro del aula en temas relacionados con el desarrollo de software, existen los Sistemas de Tutoría Inteligente (ITS) que a través de Inteligencia Artificial (AI) sirven de ayuda para detectar el aprendizaje de los estudiantes de manera oportuna y brindar una alternativa de conocimiento (Rodríguez, 2021).

Los Sistemas de Tutoría Inteligente son sistemas diseñados para replicar la efectividad de la tutoría humana en herramientas digitales. La efectividad de la tutoría individual en la instrucción en grupos grandes se ha establecido con experimentos con tutores humanos.

La tutoría es una forma de atención educativa donde el profesor apoya a un estudiante o a un grupo de estudiantes de una manera sistemática, por medio de la estructuración de objetivos, programas, organización por áreas, técnicas de enseñanza apropiadas e integración de grupos conforme a ciertos criterios y mecanismos de monitoreo y control, entre otros (Crow, T, et al., 2018).

Especificaciones para la Construcción de Algoritmos Computacionales

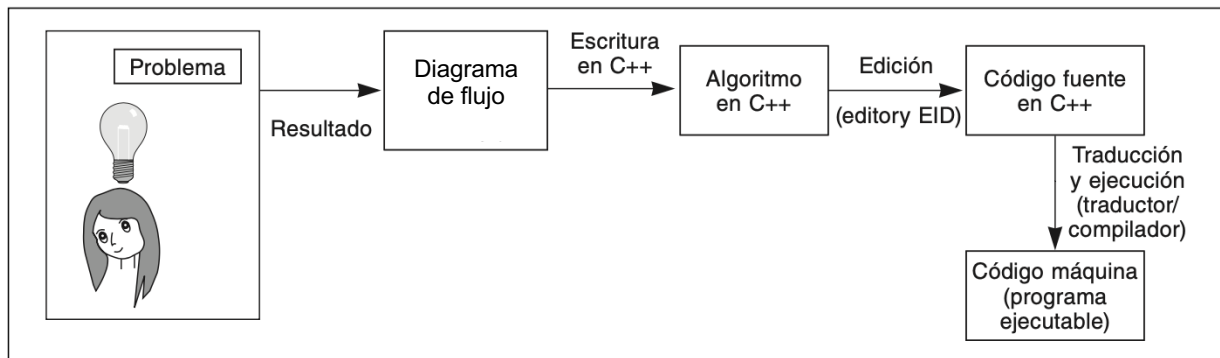
Para que el procesador de una computadora realice un proceso se le debe suministrar un algoritmo adecuado finito. El procesador debe ser capaz de procesar el algoritmo lo que significa: comprender las instrucciones de cada paso y realizar las operaciones correspondientes (Joyanes, 2008).

Un programa computacional se escribe en un lenguaje de programación y las operaciones que conducen a expresar un algoritmo en forma de programa se llaman *programación*. El proceso de traducir un algoritmo en diagrama de flujo a un lenguaje de programación se denomina

codificación, y el algoritmo escrito en un lenguaje de programación se denomina *código fuente* (Rodríguez, 2021).

El proceso de conversión de un algoritmo escrito en diagrama de flujo hasta un programa ejecutable comprensible por la máquina se muestra en la Figura 5:

Figura 5.
Proceso de transformación de algoritmo a programa ejecutable.



Nota: El gráfico representa el proceso de transformación de un algoritmo en diagrama de flujo a un programa ejecutable propuesto por **Joyanes** (2008).

Joyanes (2008) en su publicación afirma que existen 6 fases para la construcción de algoritmos computacionales y que son esenciales para el desarrollo de software, las cuales se muestran en la tabla 10:

Tabla 10.

Fases para la construcción de algoritmos computacionales.

No.	Fase.	Descripción.
1	<i>Análisis del problema.</i>	El problema se analiza teniendo en cuenta los requerimientos funcionales y no funcionales para la construcción del algoritmo.
2	<i>Diseño del algoritmo.</i>	Una vez analizado el problema, se diseña una solución que conducirá al algoritmo que resuelva el problema.
3	<i>Codificación.</i>	La solución se escribe en la sintaxis de un lenguaje de programación y se obtiene un código fuente que se compila a continuación.
4	<i>Compilación, ejecución, verificación y depuración.</i>	El programa se ejecuta y se comprueba su funcionalidad y se eliminan todos los errores puedan manifestarse.
5	<i>Mantenimiento.</i>	El programa se modifica y se actualiza cada vez que sea necesario.
6	<i>Documentación.</i>	Escritura de las diferentes fases del ciclo de vida del software en manuales de usuario y técnico.

Nota: Esta tabla representa las fases para la construcción de algoritmos y muestra los conocimientos necesarios previos a la construcción de software de acuerdo Joyanes (2008).

El desarrollo de esta investigación se enfoca en la fase *diseño del algoritmo* a través de los diagramas de flujo ya que es una de las fases iniciales en el desarrollo de software (Joyanes, 2008), se analizan los principales elementos los cuales son:

- a). Declaración de variables.
- b). Usos y aplicaciones de condicionantes (Si, Si – No)
- c). Usos y aplicaciones de funciones cíclicas (para, desde y mientras).

Capítulo III

Metodología de la investigación

Diseño de Investigación

El diseño de la investigación esta enfocado a las estrategias y procedimientos que se deben abordar en el momento de llevar a cabo una investigación científica. Niño (2011) plantea que el diseño de la investigación tiene que ver con la planificación general del proceso de investigación, iniciando desde la selección del tema hasta las técnicas, instrumentos de recolección de datos y criterios de análisis de resultados.

Para la realización de la investigación se adoptó una metodología cuantitativa como principal paradigma de investigación. Hernández, et al. (2014) manifiesta en su publicación que la investigación cuantitativa se centra en el estudio de un conjunto de procesos secuenciales probatorios. Cada una de las fases son consecutivas con un orden riguroso, se genera a partir de una idea principal derivándose los objetivos, preguntas de investigación y revisión del estado del arte para la construcción de una visión teórica. De las preguntas de investigación se derivan las hipótesis y se determinan variables, posteriormente se diseña la investigación, se define la muestra, se recolectan datos, se hace un análisis exhaustivo para finalmente generar un reporte de resultados (Hernández, et al., 2014).

Según Kerlinger y Lee (2002) los elementos importantes en el planteamiento del problema de investigación cuantitativa son:

- El problema debe expresar una relación entre dos o más conceptos o variables (características o atributos de personas, fenómenos, organismos, materiales, eventos, hechos, sistemas, etc., que pueden ser medidos con puntuaciones numéricas).

- El problema debe estar formulado como pregunta, claramente y sin ambigüedades; por ejemplo: ¿qué efecto?, ¿en qué condiciones...?, ¿cuál es la probabilidad de...?, ¿cómo se relaciona... con...?

- El planteamiento debe implicar la posibilidad de realizar una prueba empírica, es decir, la factibilidad de observarse en la “realidad objetiva”.

Hernández, et al. (2014) indica que es necesario que en una investigación se debe cuestionar acerca de las consecuencias del estudio, es decir, se debe tener establecido claramente el objetivo de la investigación seleccionando la metodología correcta.

El enfoque cuantitativo de esta investigación generó datos numéricos los cuales a través de sistema estadístico de los resultados se comprobó los supuestos de las variables Sistema de Tutoría Inteligente y aprendizaje de algoritmos computacionales. Hernández, et al. (2014) manifiestan que este enfoque usa la obtención de la información con el objetivo de verificar la hipótesis teniendo en consideración el uso de los números y la disciplina estadística con el fin de fijar aspectos comportamentales de comprobación de enfoques teóricos, es decir, hace una medición del comportamiento de las variables y la comprobación de las teorías a partir de la generación de la base de datos.

El nivel de estudio fue exploratorio ya que se analizó la innovación tecnológica aplicada en la educación para ver el comportamiento del conocimiento de los alumnos posterior de haber usado el modelo ALGO-ITS. Hernández, et al. (2014) establecen que este modelo de investigación tiene el objetivo de analizar el estudio de temas innovadores para el análisis de fenómenos desconocidos. exploratorio

El tipo de investigación que se llevó a cabo es descriptivo ya que detalla la relación entre las variables Sistema de Tutoría Inteligente y aprendizaje de algoritmos computacionales.

Hernández, et al. (2014) al respecto establecen que la investigación descriptiva pretende analizar las características y propiedades de las variables sujetas a estudio.

El diseño de la investigación fue cuasiexperimental dado que no se manipularon las variables independiente y dependiente, solo se obtuvo los resultados de verificación de relación de las variables respecto al aprendizaje de algoritmos computacionales.

Hernández, et al. (2014) afirma en su publicación que los diseños de investigación cuantitativa cuasiexperimental recolectan datos en diferentes momentos para hacer deducciones respecto a los cambios que se presenten e identificar sus consecuencias. Los diseños longitudinales recolectan información en diferentes puntos para identificar las causas y efectos.

Con base al objetivo de analizar el impacto de la implementación de un ITS bajo la teoría de la educación de la psicología cognitiva, que proporcione soporte en el proceso de aprendizaje de algoritmos computacionales en estudiantes del primer ciclo de formación de la carrera de ITI de la UPV, se presenta una metodología a través de la implementación de un ITS que sirva de ayuda en el aprendizaje de algoritmos computacionales dentro de la materia de **Introducción a la Programación de la carrera** de ITI en la UPV.

En el diseño de la investigación el propósito principal del post-test es identificar la relación causal entre la variable dependiente correspondiente al sistema informático ITS de soporte en el aprendizaje de algoritmos computacionales y la variable independiente relacionada con el aprendizaje significativo de algoritmos. El diseño presentado en la investigación está enfocado a analizar estadísticamente los indicadores sobre el conocimiento previo y posterior al uso de ITS de los estudiantes que forman parte del grupo experimental, el diseño cuantitativo ayudará a validar la hipótesis de investigación y a cumplir los objetivos.

Dentro de ese orden de ideas, la investigación se divide en tres vertientes: la caracterización de conocimientos previos relacionados al desarrollo de algoritmos computacionales a través de diagramas de flujo, la implementación de un modelo de ITS desarrollado con IA haciendo uso de las redes neuronales de aprendizaje profundo para el mejor entrenamiento del ITS y que pueda ser capaz de identificar los conocimientos y habilidades de cada uno de los estudiantes del grupo experimental y finalmente la evaluación del conocimiento de los estudiantes que hicieron uso del ITS para que la herramienta tecnológica ITS pueda de manera personalizada ofrecer un apoyo en el desarrollo de algoritmos y generar un ambiente de motivación para implementar la lógica computacional a través de diagramas de flujo. Esta investigación sigue el modelo de tres etapas: concepto, desarrollo e impacto (Nunamaker Jr. et al., 1990).

El ITS desarrollado sirve como una prueba del concepto de la investigación fundamental, así como una herramienta para la investigación continua. Por lo tanto, esta investigación siguió de cerca los conceptos de la metodología de investigación de Desarrollo de Sistemas (Nunamaker Jr. et al., 1990).

Sujetos de Investigación

El diseño consiste en una encuesta transversal al grupo experimental correspondiente a los 94 alumnos de nuevo ingreso periodo 2020 a la carrera de ITI de la UPV procedentes de bachilleratos como el Colegio de Bachilleres de Tamaulipas (COBAT), Instituto Tamaulipeco de Capacitación para el Empleo (ITACE), preparatorias, Centro de Bachillerato Tecnológico industrial y de servicios (CBTis), y de Centro de Estudios Tecnológicos industrial y de servicios (CETis) quienes formaron parte del grupo experimental. Para ello se contó con el consentimiento

de la Universidad Politécnica de Victoria para la exploración de los conocimientos en el ámbito del desarrollo de software. El censo realizado es de conveniencia o incidental.

Se trabajó con una muestra experimental correspondiente al 100% a los alumnos de nuevo ingreso al primer ciclo de formación de la carrera de ITI de la UPV periodo 2020. La muestra fue no aleatoria a conveniencia considerando la totalidad de los alumnos que ingresaron a la UPV desde diferentes perfiles de bachilleratos, esto con la intención de identificar de una manera eficaz el tratamiento de los resultados del post- test que afecten a la variable dependiente correspondiente al aprendizaje de algoritmos computacionales a través de diagramas de flujo (Cook y Campbell, 1986).

Los estudiantes de la carrera de Ingeniería en Tecnologías de la Información completaron de manera voluntaria las encuestas de caracterización del conocimiento y evaluación de conocimientos posterior al uso del modelo del ITS, tras haber sido informados del propósito de la investigación. No se recogieron en ningún momento datos confidenciales de los estudiantes identificándose cada uno de ellos por la matrícula asignada por la Universidad.

Procedimiento

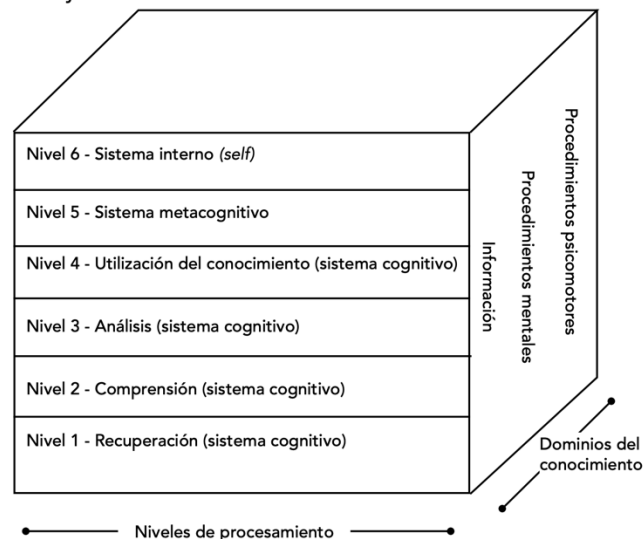
A los alumnos del grupo experimental se les aplicó, previamente al uso del ITS, un examen pre-test el cual su función era la identificación y caracterización de conocimientos previos en el ámbito del desarrollo de software. Es necesario caracterizar a un alumno para identificar su nivel de dominio en los temas relacionados a la construcción de algoritmos computacionales. Existen métodos pedagógicos para identificación del nivel de dominio de los estudiantes como lo es la taxonomía de Marzano y Kendall (2007) la cual es utilizada para identificar las dimensiones cognitivas sobre el conocimiento en las que están posicionados los alumnos respecto al nivel de

conocimiento en el desarrollo de algoritmos computacionales. Para el estudio de esta investigación se hizo uso de ella para categorizar los aprendizajes e identificar las dimensiones de conocimiento en las que están posicionados los alumnos respecto al nivel de dominio en el desarrollo de algoritmos computacionales. Con la identificación del conocimiento se puede implementar un modelo de ITS el cual permitirá con base a sus características mejorar el aprendizaje de los estudiantes respecto a la construcción de software.

Marzano y Kendall (2007) en su taxonomía hacen hincapié en la dificultad de ejecución de un proceso mental en donde dimensionan el nivel de aprendizaje del alumno respecto a su sistema de pensamiento y dominios de conocimiento. La taxonomía está construida mediante dos dimensiones principales: niveles de procesamiento mental y los dominios de conocimiento clasificados en: información, procedimientos mentales y procedimientos motores (Álvarez y Valdebenito, 2017).

La figura 6 muestra las dimensiones de la taxonomía de Marzano y Kendall (2007).

Figura 6.
Dimensiones del conocimiento de la taxonomía de Marzano y Kendall



Nota: El gráfico representa las dimensiones de conocimiento de la taxonomía de Marzano y Kendall. Tomado de *Marzano y Kendall (2017)*.

Técnicas

El modelo implementado para la recopilación de información en esta investigación fue a través de encuestas y la técnica cuestionario. De acuerdo con López-Roldán, et al. (2015) los cuestionarios son una herramienta utilizada para obtener información utilizando de manera sistemática la interrogación hacia los sujetos de investigación que conforman una muestra o censo con el objetivo de recopilar información sobre percepciones derivadas de un problema de investigación.

Por otro lado, Bernal (2000) dice que el cuestionario es un instrumento que es redactado a partir de un conjunto de interrogantes enfocadas a responder el contenido de las variables de una investigación. Cuando se redactan deben ser breves y comprensibles.

A partir de la taxonomía de Marzano y Kendall (2007) para la clasificación e identificación de aprendizajes, en la presente investigación se diseñaron instrumentos de recolección de datos aplicando un pre-test y un post-test a través de cuestionarios en Google Forms en donde se muestran los resultados de las evidencias de conocimientos de los alumnos respecto a los aprendizajes de algoritmos computacionales, el nivel de dominio al momento de desarrollar software se clasifica según las dimensiones del conocimiento de la taxonomía. Los instrumentos incluyen ítems que concentran información de aprendizajes que permite clasificarlo según las habilidades de los estudiantes encuestados.

El pre-test aplicado contiene ítems relacionados con la caracterización y nivel de conocimientos en el desarrollo de algoritmos, previa al uso de un ITS, abarca preguntas acerca a la experiencia y dominio de las tecnologías de la información y programación computacional, se hace un reconocimiento de la contextualización del conocimiento previo de los alumnos sobre esos temas. Así mismo el pre-test específicamente contiene ítems de exploración del nivel dominio en el desarrollo de algoritmos y pseudocódigo en donde los alumnos resuelven una serie de ejercicios básicos de programación computacional tomados de la literatura de Joyanes (2008) y con base a sus resultados se hace una clasificación a través de la taxonomía de Marzano y Kendall (2007) para categorizar el nivel de dominio. Esta información es el punto de partida para el ITS para tener el conocimiento personalizado específico de cada uno de los estudiantes del grupo experimental.

Posteriormente se aplicó un cuestionario post-test diseñado para identificar el nivel de conocimiento una vez utilizado el ITS propuesto en donde se hará un análisis del nivel de dominio en el desarrollo de diagramas de flujo. Para este instrumento se aplicaron ítems de resolución de problemas algorítmicos tomados de Joyanes (2008) en donde partiendo de la caracterización de conocimientos del pre-test, el ITS hace una identificación personalizada del nivel de dominio de

cada alumno del grupo experimental para que el ITS proporcione el soporte correspondiente para generar un ambiente de motivación y conocimiento en la resolución de problemas algorítmicos.

Validez de los Instrumentos

La validez de acuerdo con Taherdoost (2016) está relacionado con un juicio que proporciona un experto con relación a una construcción. La validez tiene que ver el nivel de relación de un juicio del experto y un constructo. Es decir, que un instrumento contiene una validez aparente cuando el contenido es relevante para obtener la información que se requiere en la investigación.

Para esta investigación se realizó una validez de dos cuestionarios a través del juicio de tres expertos con conocimientos y amplio dominio de competencias relacionadas a la innovación educativa y tecnológica.

La tabla 11 muestra los resultados de la validación con los expertos en el área de la innovación educativa aplicados en esta investigación.

Tabla 11.

Validez de los instrumentos de recolección de datos.

No.	Experto	Categoría	Indicador
1	Dr. Said Polanco Martagón.	Suficiencia	Los ítems son suficientes para medir la dimensión.
		Claridad	Los ítems son claros, tienen semántica y sintaxis adecuada.
		Coherencia	Los ítems se encuentran completamente relacionados con la dimensión que se está midiendo.
2	Dr. Jorge Arturo Hernández Almazán.	Relevancia	Los ítems son relevantes y deben ser incluidos.
		Suficiencia	Los ítems son suficientes para medir la dimensión.
		Claridad	Los ítems son claros, tienen semántica y sintaxis adecuada.
3	Dr. Hiram Herrera Rivas.	Coherencia	Los ítems se encuentran completamente relacionados con la dimensión que se está midiendo.
		Relevancia	Los ítems son relevantes y deben ser incluidos.
		Suficiencia	Los ítems son suficientes para medir la dimensión.
		Claridad	Los ítems son claros, tienen semántica y sintaxis adecuada.
		Coherencia	Los ítems se encuentran completamente relacionados con la dimensión que se está midiendo.
		Relevancia	Los ítems son relevantes y deben ser incluidos.

Nota: Esta tabla representa validación de los instrumentos de expertos que miden las variables de estudio.

Confiabilidad

La confiabilidad está relacionada con la consistencia que tiene el instrumento de recolección de datos que, al medir en diferentes aplicaciones, los resultados deben coincidir respecto al nivel de confianza. Una aproximación fiable de la confiabilidad se obtiene mediante tres parámetros: La homogeneidad, estabilidad y equivalencia (Heale y Twycross, 2015).

Oviedo y Campo-Arias (2005) expresan en su investigación que la confiabilidad indica la consistencia de un instrumento e índice de asociación que toma valores de 0 y 1. El valor 0 indica que no existe relación y el valor 1 muestra relación perfecta. De otro modo George y Mallery (2003) manifiestan que existen criterios que permiten evaluar, a través del coeficiente del Alfa de Cronbach, la correlación entre las interrogantes de los ítems de los instrumentos; esto son:

- Coeficiente alfa $>.9$ es excelente
- Coeficiente alfa $>.8$ es bueno
- Coeficiente alfa $>.7$ es aceptable
- Coeficiente alfa $>.6$ es cuestionable
- Coeficiente alfa $>.5$ es pobre
- Coeficiente alfa $<.5$ es inaceptable

Para el caso de esta investigación la confiabilidad se realizó a través del Alfa de Cronbach obtenida de un censo de 94 alumnos pertenecientes al grupo experimental, el análisis de confianza se llevó a cabo al pre-test de caracterización y el pos-test de conocimientos posterior al uso del modelo ITS. Los resultados de la medición del nivel de confianza de los instrumentos de recolección de datos se muestran en la sección de análisis de resultados de la investigación.

Aspectos Éticos

Para la realización de esta investigación se tomó en cuenta la privacidad de la información de los integrantes del grupo experimental. En ningún momento se manipuló de manera deliberada los resultados de los instrumentos de recolección de datos ya que hubieran afectado los resultados de la investigación. Por otro lado, la información relevante está resguardada en una base de datos de manera segura en un servidor de la Universidad.

Al respecto el Steneck (2007) manifiesta que la ética en la investigación está relacionada con la forma de ser un investigador honesto, objetivo, integro, cuidadoso y sincero; teniendo un alto sentido de respeto hacia la autoría, actuando con responsabilidad sin discriminación; ser competente, legal y asumir siempre el rol de protección hacia las personas.

Planificación de la metodología

El diseño de la investigación consta de 3 fases. La fase 1 considera el análisis y caracterización del aprendizaje del grupo experimental sobre el desarrollo de algoritmos computacionales, se considera una duración de 6 meses. La fase 2 la cual consta de la implementación del modelo de ITS de soporte en el aprendizaje de algoritmos computacionales, inicia en el mes 21 del desarrollo de la investigación. Finalmente, la fase 3 que comprende la evaluación del conocimiento del grupo experimental respecto a la construcción de algoritmos la cual inicia en el mes 33. La tabla 12 muestra la planificación de la metodología de la investigación.

Tabla 12.
Planificación de la metodología de investigación.

	Mes													
	3	6	9	12	15	18	21	24	27	30	33	36	39	42
Fase 1														
Fase 2														
Fase 3														

Nota: Esta tabla representa la planificación de la metodología para el desarrollo de la investigación.

Para la implementación del ITS se basó en la técnica de IA de redes neuronales de aprendizaje profundo mediante el uso de la librería Tensorflow para diseñar un módulo de dominio para representar el conocimiento y tener como resultado una herramienta tecnológica educativa apta para actuar con inteligencia a través de su entrenamiento y tener la capacidad de ofrecer soporte a los estudiantes en el desarrollo de software.

Respecto al desarrollo del ITS se considera para su construcción teorías de sistemas con experimentación y observación para validar la hipótesis. El proceso de investigación consta de 5 pasos principales: Construcción de un marco conceptual, desarrollo de una arquitectura de sistemas, análisis y diseño del sistema, construcción del sistema y observación y evaluación del sistema (Nunamaker Jr. et al., 1990). Este proyecto de investigación se divide en tres fases para

incorporar estos cinco pasos principales de la metodología de investigación del desarrollo de algoritmos y sistemas.

Caracterización del Conocimiento

La primera fase comprendió acerca de la obtención de la caracterización del conocimiento de los alumnos respecto a la construcción de algoritmos computacionales y uso de las tecnologías de la información, dicho conocimiento permitirá definir el módulo de dominio del ITS para el entrenamiento y soporte oportuno al momento de ser utilizado, en esta fase se definieron los requisitos del ITS para construir un sistema robusto y que brinde apoyo en la construcción de diagramas de flujo.

La Inteligencia Artificial (IA) desempeña un papel importante en el diseño de los ITS, este tema también se estudió con cierta extensión. Los ITS existentes y otros sistemas de enseñanza computarizados que enseñan programación fueron estudiados para identificar los requisitos del sistema. La información obtenida de esta literatura también se utilizó para desarrollar una arquitectura de sistemas basada en la arquitectura estándar de un Sistema de Tutoría Inteligente.

En la fase 1 se hace una descripción de la metodología utilizada para la generación de evidencias sobre conocimiento en el desarrollo de algoritmos computacionales, se hace una exploración al grupo experimental sobre los conocimientos en el desarrollo de algoritmos aplicando un instrumento de levantamiento de datos que permite identificar las habilidades previas en el ámbito del desarrollo de software.

Dada la importancia que tiene el desarrollo tecnológico para la generación de algoritmos computacionales, es importante hacer la pregunta de investigación: *¿Cuál es el impacto de la implementación de un Sistema de Tutoría Inteligente (ITS) de apoyo en el aprendizaje de*

algoritmos computaciones en los estudiantes de primer cuatrimestre del PE de ITI de UPV? Para responder a la pregunta se deben identificar los conocimientos previos en el desarrollo de algoritmos computacionales bajo los siguientes rubros:

- a) *Datos generales del alumno,*
- b) *Conocimientos en el desarrollo de software,*
- c) *Prácticas de programación,*
- d) *Recursos y materiales didácticos orientados a los alumnos,*
- e) *Estudios, conocimientos y habilidades y*
- f) *Tutorías y asesorías.*

El diseño de investigación es cuasi experimental con un censo en donde se aplicó un instrumento de caracterización a los 94 alumnos de la generación 2020 quienes representan el 100% de la población de nuevo ingreso.

Actividades

Basado en la taxonomía de Marzano y Kendall (2007) para identificar el nivel de dominio de los estudiantes al momento de la construcción de algoritmos, se implementó un instrumento de recolección de datos de caracterización de aprendizajes, la tabla 13 muestra las características e ítems de las actividades desarrolladas por los estudiantes respecto en la construcción de algoritmos previo al uso y experimentación a través de una herramienta tecnológica educativa como lo es el ITS. La implementación del instrumento fue a través de la plataforma Google Forms en donde se tiene integrados los ejercicios a resolver graduales en complejidad.

Tabla 13.

Nivel de conocimiento de algoritmos a partir de la taxonomía de Marzano Kendall (2007).

Nivel cognoscitivo	Actividad / ítems	Objetivo
Recuperación	<ol style="list-style-type: none">1. De la siguiente lista, ¿Cuáles son las estructuras de control básicas de un algoritmo?2. ¿Cuál es la estructura de datos que permite múltiples valores organizados de forma indexada?3. Escoge la opción correcta que enlista cada tipo de representación de un algoritmo en este mismo orden.	Definir el nivel de memorización de bloques de diagramas de flujo.
Comprensión	<ol style="list-style-type: none">1. Identifica el tipo de dato que admite valores con punto decimal de 16 bits.2. De acuerdo con tus conocimientos, ¿Cuáles son los pasos que se consideran en un diagrama de flujo?3. De acuerdo con las siguientes respuestas, ¿Cuál es el valor que puede cambiar durante la ejecución de un algoritmo?	Identificar el nivel de interpretación y asociación en la construcción de diagramas de flujo
Análisis	<ol style="list-style-type: none">1. ¿Cuál de estos ejemplos no es un arreglo?2. Analiza el siguiente arreglo: [2, 4, 5, 1, 5, 7] ¿Cuál es la salida del arreglo anterior?3. De acuerdo con el siguiente algoritmo, ¿Cuál es la salida del siguiente ciclo?	Identificar la capacidad de análisis para la construcción de algoritmos computacionales a través de diagramas de flujo.
Aplicación	<ol style="list-style-type: none">1. ¿Cuál es la salida de la siguiente instrucción?2. ¿En qué parte del algoritmo se debe cambiar para que 'n' sea 'igual' a 45?3. De acuerdo con la estructura de declaración de variables, ¿Cuál es la manera correcta de declaración?	Identificar el nivel de aplicación y modificación de algoritmos para proponer una alternativa de solución.
Metacognición	<ol style="list-style-type: none">1.Cuál es la manera correcta de declarar las variables a través del siguiente algoritmo.2. Define el orden correcto conforme la estructura de un algoritmo.3. Escoge la opción que optimice el siguiente algoritmo para realizar una suma de dos números.	Identificar la solución a problemas reales para proponer una automatización a través de un algoritmo computacional programable en algún lenguaje. Se analiza el nivel de construcción de algoritmos a través de la eficacia.

Nota: La tabla representa los ejercicios graduales en complejidad propuestos en el instrumento de recolección de datos 1: Caracterización de los aprendizajes, con el objetivo de identificar el nivel de dominio de los estudiantes al momento de la construcción de algoritmos previo a utilizar un ITS.

Implementación de un Modelo ITS

La segunda fase del diseño de la investigación corresponde al diseño e implementación de un modelo de ITS que integra elementos tecnológicos y educativos para desarrollar conocimientos y habilidades en el desarrollo de algoritmos computacionales que fue utilizado por el grupo experimental para el soporte en el aprendizaje de algoritmos computacionales. Esta fase incluye el

diseño de esquemas y bases de conocimiento necesarias para el sistema, los métodos para modelar los diferentes componentes de la arquitectura de los ITS fueron analizados detalladamente.

Construir un ITS es una tarea compleja que lleva tiempo. Se ha estimado que toma de 200 a 300 horas para que brinde una hora de instrucción (Aleven et al., 2006). El número de horas de instrucción necesarias para enseñar algoritmos y programación computacional es muy grande. Por lo tanto, es necesario limitar los temas enseñados por el ITS. Se decidió que el sistema atendería a programadores de nivel principiante sin conocimientos previos de programación.

Posteriormente en el proceso de diseño fue encontrar representaciones adecuadas para el tema seleccionado. Era necesario representar este tema de una manera que permitiera analizar las respuestas de los estudiantes e identificar diferentes soluciones a un problema dado. Las consultas en la literatura se utilizaron para estudiar los diferentes métodos que se habían utilizado previamente para representar el conocimiento de los sujetos. De los muchos métodos que se habían utilizado anteriormente, era necesario encontrar un método que fuera lo suficientemente flexible como para manejar la multitud de variaciones posibles al escribir algoritmos computacionales. También era necesario que el método seleccionado facilitara el proceso de proporcionar comentarios apropiados basados en errores particulares cometidos por los estudiantes. En consecuencia, se decidió utilizar la lógica de predicado de primer orden “FOPL” (por sus siglas en inglés) para modelar la base de conocimiento. Aunque esta representación se había utilizado previamente para los ITS, no se había utilizado para representar el tema para analizar programas de computadora.

El tema seleccionado para incluir en el sistema se estudió en detalle para ver cómo podría modelarse utilizando FOPL. Este fue un proceso iterativo. En cada paso, se seleccionó un tipo de construcción de programación y se propuso un modelo adecuado. Luego, se consideró un conjunto

de ejemplos que demuestran el uso de esta construcción. Estos ejemplos se rastrearon manualmente para garantizar que el modelo propuesto pudiera usarse para analizar estos programas. Si se encontraron problemas, el modelo se perfeccionó y el proceso se repitió. Esto se hizo para todas las construcciones que se iban a incluir en el sistema implementado.

Respecto a la confiabilidad del ITS implementado en la UPV es necesario que contenga bien definidos los módulos de dominio, conocimiento y estudiante para su óptimo funcionamiento. El núcleo principal del modelo es el módulo de dominio el cual utiliza técnicas de IA para construir una red neuronal de aprendizaje profundo a través de Machine Learning en donde es imposible tutorizar adecuadamente sin tener una buena representación del tema. La representación del conocimiento del alumno depende de cómo se representa el tema, por lo tanto, el enfoque principal de esta investigación es aplicar una metodología cuantitativa para que a través de las estadísticas se identifique el nivel de conocimiento y desempeño de los estudiantes en el desarrollo de algoritmos, asimismo el método descriptivo permitirá identificar y de analizar las variables independiente y dependiente de la investigación para medir la correlación entre las habilidades y destrezas del grupo experimental para el desarrollo de algoritmos y el diseño e implementación de un modelo de Sistema de Tutoría Inteligente de apoyo en el conocimiento.

En los ITS el módulo del alumno ajusta las características de un alumno que utiliza el sistema. Los estudiantes son seres humanos que difieren en muchas peculiaridades, como el conocimiento de la materia, el nivel de educación, el estilo de aprendizaje, la motivación y la edad. Modelar todos estos rasgos es un problema complicado, por lo tanto, esta investigación se centró solo en modelar el conocimiento actual de cada estudiante sobre el tema que enseña el sistema.

Como se describió previamente, diferentes métodos de modelado de estudiantes se han utilizado de manera exitosa en distintos ITS. Estas representaciones fueron estudiadas a través de

encuestas de literatura, sobre esos métodos, se seleccionó el modelo de estudiante de redes neuronales de aprendizaje profundo como el método más adecuado para el ITS de aprendizaje de algoritmos y programación.

El módulo de enseñanza se concentra en métodos adaptados por el sistema. Los diferentes métodos de enseñanza utilizados por distintos autores en la construcción de ITS se analizaron con base a revisiones de literatura. De los métodos descritos previamente, se seleccionaron los más idóneos para el sistema.

Evaluación de los Aprendizajes del Grupo Experimental

La tercera fase de la investigación corresponde a la evaluación del aprendizaje sobre la construcción de algoritmos al grupo experimental (alumnos que utilizaron el ITS) en una población de estudiantes del primer cuatrimestre de la carrera de ITI de la UPV. Los participantes fueron participantes del grupo experimental corresponden a la generación 2020 de la carrera de ITI de la UPV con la ventaja adicional de identificar sus debilidades en el desarrollo de software en el momento de estudiar algoritmos y programación computacional gracias al uso del ITS. Los estudiantes que participaron en la investigación recibieron calificaciones las cuales no contaron para su promedio final de la materia de Fundamentos de Programación, esto por cuestión de ética. Por lo tanto, a todos los alumnos de la asignatura hicieron uso del ITS denominado ALGO-ITS.

Durante el proceso de evaluación, se recopilarán datos de los estudiantes que se utilizarán en el sistema. Hay varios aspectos que la evaluación de un ITS necesita considerar. Un punto importante es si los estudiantes realmente adquirieron conocimiento al usar el sistema, los resultados previos y posteriores a la prueba fueron la principal forma de datos utilizada para este propósito. También fue necesario evaluar la validez del modelo de estudiante (Mark y Greer,

1993). Los detalles de las interacciones de los estudiantes con el sistema se registraron y analizaron junto con los resultados previos y posteriores a la prueba pre-test, la usabilidad del sistema también fue una consideración importante que se evaluará. Esto se analizará utilizando respuestas cuantitativas de un cuestionario.

Actividades

Para la fase 3 se aplicó un post-test al grupo experimental para identificar el nivel de conocimiento en el momento de construcción de algoritmos computacionales a través de diagramas de flujo posterior al uso del modelo de ITS. El instrumento de recolección de datos se aplicó a través de la plataforma Google Forms en donde los alumnos solucionaban ejercicios graduales de complejidad una vez que cada uno de ellos hizo uso del modelo ALGO-ITS.

Igualmente que en la implementación del pre-test, se hizo uso de la taxonomía de Marzano y Kendall (2007) para la aplicación del post-test para conocer el nivel de conocimiento de los estudiantes para desarrollar software, el grupo experimental hizo uso durante dos cuatrimestres del ITS el cual ofreció soporte y ayuda en la construcción de algoritmos generando un ambiente de motivación, a través de la teoría de la psicología cognitiva, a cada uno de los estudiantes identificando el nivel de avance y dimensionando sus aprendizajes mediante la taxonomía de Marzano y Kendall (2007).

Respecto a la validación del instrumento de recolección de datos implementado al grupo experimental, se hizo uso de la técnica por juicio de expertos en el área de programación computacional lo cual da la certeza de que los datos recolectados proporciona evidencia del avance de los aprendizajes en la construcción de algoritmos mediante diagramas de flujo y pseudocódigo.

La tabla 14 muestra los ítems considerados en el post-test clasificados por las dimensiones de la taxonomía de Marzano y Kendall (2007) para recolectar las evidencias de conocimiento del grupo experimental.

Tabla 14.

Ítems de post-test clasificados en la taxonomía de Marzano Kendall (2007).

Nivel cognoscitivo	Ítems	Objetivo
Recuperación	<ol style="list-style-type: none"> 1. ¿Cuáles son las estructuras de control básicas de un algoritmo? 2. De la siguiente lista, ¿cuál es la estructura de datos que permite múltiples valores organizados de forma indexada? 	Definir el nivel de memorización de bloques de diagramas de flujo.
Comprensión	<ol style="list-style-type: none"> 1. Identifica el concepto que corresponda a la siguiente descripción: Dentro de los componentes de un pseudocódigo es el nombre que se le asigna el algoritmo. 2. Con base al diagrama que se presenta responde a la siguiente pregunta: ¿Qué error detectas en el siguiente diagrama el cual debe sumar 2 números? 	Identificar el nivel de interpretación y asociación en la construcción de diagramas de flujo
Análisis	<ol style="list-style-type: none"> 1. Con base al diagrama que se presenta responde a la siguiente pregunta: ¿qué error puedes detectar en el siguiente diagrama de flujo, el cual debe solicitar 3 calificaciones, realizar su promedio y mostrar si el alumno aprobó o no la materia siendo 70 la mínima de calificación? 	Identificar la capacidad de análisis para la construcción de algoritmos computacionales a través de diagramas de flujo.
Aplicación	<ol style="list-style-type: none"> 1. Con base al diagrama que se presenta responde a la siguiente pregunta: ¿qué error puedes detectar en el siguiente diagrama de flujo, 	Identificar el nivel de aplicación y modificación de algoritmos para proponer una alternativa de solución.
Metacognición	<ol style="list-style-type: none"> 1. De acuerdo con tus conocimientos ¿Qué pseudocódigo corresponde a un programa que pida por teclado dos números enteros y muestre su suma, resta, multiplicación, división y el resto de la división? Si la operación no es conmutativa, también se mostrará el resultado invirtiendo los operadores. 	Identificar la solución a problemas reales para proponer una automatización a través de un algoritmo computacional programable en algún lenguaje. Se analiza el nivel de construcción de algoritmos a través de la eficacia.

Nota: La tabla representa los ejercicios graduales en complejidad propuestos en el instrumento de recolección de datos post-test posterior al uso del modelo ALGO-ITS.

Capítulo IV

Análisis de Resultados

Para el análisis de los resultados de la investigación se diseñaron dos instrumentos de recolección de datos los cuales fueron creados en coordinación con un grupo de expertos en docencia, investigación, Sistemas de Tutoría Inteligente e Inteligencia Artificial, con el fin de generar certidumbre en la caracterización de los aprendizajes y en la evaluación del conocimiento del grupo experimental, por ello es necesario certificar los instrumentos para tener un grado de pertinencia, estabilidad y seguridad en cada uno de los ítems.

Para validar los instrumentos de recolección de datos se implementó el coeficiente del Alfa de Cronbach ya que proporciona una metodología para conocer el grado de confianza y consistencia de los ítems de cada reactivo lo cual permite medir una cualidad no observable, como lo es la inteligencia, en una población de sujetos (Belén et al., 2015).

Para determinar la confiabilidad y consistencia de los instrumentos, se utilizó el software especializado IBM SPSS versión 28, en donde se hizo el cálculo el coeficiente de Alfa de Cronbach, se pudo estimar el nivel de confianza interna de los instrumentos. La medida de confianza a través del Alfa de Cronbach establece que los ítems miden un mismo constructo y están altamente relacionados. Cuanto más cercano se encuentra el valor del Alfa de Cronbach al valor 1, mayor es el nivel de consistencia y confianza interna de los ítems analizados.

Con base a los instrumentos aplicados al grupo experimental, en primera instancia el primero permitió identificar el conocimiento y habilidades de inicio acerca del desarrollo de software en dónde a través de la taxonomía de Marzano y Kendall (2007) se hace una categorización de los aprendizajes. Por otro lado, el segundo evaluó los conocimientos de los estudiantes del grupo experimental relacionado a la construcción de algoritmos después de haber

hecho uso del Sistema de Tutoría Inteligente ALGO-ITS en la Universidad Politécnica de Victoria (UPV), finalmente se hizo una comparación de los resultados de ambos instrumentos respecto al conocimiento en el desarrollo de algoritmos computacionales para identificar el grado de pertinencia de la implementación del sistema de tutoría inteligente en la UPV.

Validación del Instrumento de Caracterización del Conocimiento

La tabla 15 muestra el coeficiente Alfa de Cronbach total del instrumento de caracterización del conocimiento el cual tuvo un valor de **0.802** considerado en la escala de George y Mallery (2003) como bueno y confiable. El numero de elementos corresponde la cantidad de preguntas consideradas en el instrumento.

Tabla 15.

Estadísticas de fiabilidad

Alfa de Cronbach	N de elementos
0.802	30

Nota: La tabla representa el nivel de confianza y fiabilidad a través del Alfa de Cronbach analizado mediante SPSS versión 28.

El valor de confianza del Alfa de Cronbach representa la correlación de las respuestas de los ítems que contienen la misma unidad de medida, la interpretación se realiza con una consistencia interna del valor del Alfa de Cronbach que utiliza la correlación de los ítems.

Por otro lado, la tabla 16 muestra las estadísticas descriptivas de los elementos en donde la primera columna indica el ítem de conocimiento, la segunda columna muestra la media lo cual indica el índice de dificultad de cada pregunta respecto a las diferentes respuestas disponibles. La desviación estándar mide el grado de dispersión de las observaciones individuales alrededor de su media, y finalmente la columna N indica la población del grupo experimental lo cual es de 94 alumnos.

Tabla 16.

Estadísticas del elemento.

Ítem	Media	Desviación estándar	N
Experiencia_desarrollo	.90	.296	94
Nivel_experiencia	.66	.476	94
Desarrollo_diagramas	.82	.387	94
Nivel_desarrollo_diagramas	.87	.335	94
Dominio_lenguaje_computacional	.52	.502	94
N1_Recuperacion_P1	.88	.323	94
N1_Recuperacion_P2	.44	.499	94
N1_Recuperacion_P3	.65	.480	94
N2_Comprension_P1	.80	.404	94
N2_Comprension_P2	.81	.396	94
N2_Comprension_P3	.77	.426	94
N3_Analisis_P1	.26	.438	94
N3_Analisis_P2	.34	.476	94
N3_Analisis_P3	.60	.493	94
N4_Aplicacion_P1	.61	.491	94
N4_Aplicacion_P2	.34	.476	94
N4_Aplicacion_P3	.94	.246	94
N5_Metacognitivo_P1	.53	.502	94
N5_Metacognitivo_P2	.06	.246	94
N5_Metacognitivo_P3	.48	.502	94
Recursos_motivacion	.84	.368	94
Diagnostico_inicio_materia	.76	.432	94
Algoritmos_graduales_complejidad	.90	.296	94
Calificaciones_destrezas	.84	.368	94
Competencias_metodologia	.72	.450	94
Competencias_soluciones_programacion	.74	.438	94
Competencias_Sistemas_Inf	.74	.438	94
Conoces_motivo_tutor	.76	.432	94
Interaccion_tutor	.86	.347	94
Ayudaron_asesorias	.70	.460	94

Nota: La tabla representa las estadísticas del elemento analizado mediante SPSS versión 28.

La “**desviación estándar**” representa el nivel de dispersión de los datos respecto a la media, mientras mas amplia significa que los valores se extienden sobre un rango amplio. Dicho lo anterior se puede observar que los ítems Experiencia_desarrollo, N4_Aplicacion_P3 y Algoritmos_graduales_complejidad muestran valores bajos lo que representa que los valores esperados en el instrumento son lo ideales para evaluar el conocimiento y caracterización de los alumnos, por otro lado, los ítems Dominio_lenguaje_computacional, N1_Recuperacion_P2, N3_Analisis_P3, N5_Metacognitivo_P1 y N5_Metacognitivo_P3; muestran valores altos lo que

significa que los rangos de solución de los ejercicios del ítem se extienden por una categoría de dispersiones alto, es decir, las opciones de solución de los ejercicios de los ítems son mas amplias y complejas lo que requiere que el alumno implemente los conocimientos categorizados en la taxonomía de Marzano y Kendall (2007).

Por otro lado, la columna “**Media**” de la tabla 16 representa el índice de dificultad de cada uno de los ítems del instrumento en donde los valores cercanos a 1 representan mayor índice de dificultad, dicho lo anterior, la tabla 17 refleja los ítems que mostraron mayor índice de dificultad.

Tabla 17.
Ítems identificados con mayor índice de dificultad.

Ítem	Media
Experiencia_desarrollo	.90
Desarrollo_diagramas	.82
Nivel_desarrollo_diagramas	.87
N1_Recuperacion_P1	.88
N2_Comprension_P1	.80
N2_Comprension_P2	.81
N4_Aplicacion_P3	.94
Recursos_motivacion	.84
Algoritmos_graduales_complejidad	.90
Calificaciones_destrezas	.84
Interaccion_tutor	.86

Nota: La tabla representa los ítems del instrumento con mayor índice de dificultad.

Con base al análisis de las estadísticas de los elementos llevados a cabo a través del software SPSS versión 28, se identifica que los ítems que reflejan mayor índice de dificultad forman parte de los niveles avanzados de la taxonomía de Marzano y Kendall (2007) en donde el nivel de aplicación requiere que el alumno analice, memorice, tenga un sentido de motivación e implemente soluciones algorítmicas, dicho lo anterior, los ítems de los niveles avanzados de la taxonomía requieren mayor esfuerzo por los estudiantes para la solución de los algoritmos.

La tabla 18 refleja el resumen de las estadísticas proporcionadas por el software SPSS donde se muestran las medias de los elementos el cual es el promedio de los ítems de la tabla 15. Se observa la **varianza** que es la medida de dispersión que representa la movilidad de los datos

arrojados por los ítems, la **covarianza** representa el grado de variación de ítems aleatorios respecto a la media, y finalmente se muestra la correlación que es la asociación de los ítems evaluando las tendencias creciente y decreciente de los resultados de los ítems.

Tabla 18.

Estadísticas de elemento resumen.

	Media	Mínimo	Máximo	Rango	Máximo / Mínimo	Varianza	N de elementos
Medias de elemento	.671	.064	.936	.872	14.667	.047	30
Varianzas de elemento	.177	.060	.252	.192	4.176	.004	30
Covarianzas entre elementos	.021	-.059	.204	.264	-3.450	.001	30
Correlaciones entre elementos	.112	-.292	.967	1.259	-3.309	.033	30

Nota: La tabla representa las estadísticas del elemento resumen analizado mediante SPSS versión 28.

La tabla 19 muestra las estadísticas totales del elemento en donde se muestra la columna **“Alfa de Cronbach si el elemento se ha suprimido”** la cual permite identificar los ítems que se pueden eliminar para incrementar el nivel de confianza del instrumento, se observa el ítem **“N4_Aplicacion_P2”** el cual si es eliminado el nivel de confianza se eleva a .811 pero se tomó la decisión de no hacerlo ya que el ítem forma parte de la evaluación del cuarto nivel de la taxonomía de Marzano y Kendall (2007) de aplicación para obtener el conocimiento de los alumnos y es necesario analizar sus resultados para medir el nivel de dominio de los estudiantes en el momento de desarrollar algoritmos.

Tabla 19.

Estadísticas del total de elementos.

	Media de escala si el elemento se ha suprimido	Varianza de escala si el elemento se ha suprimido	Correlación total de elementos de corregida	Alfa de Cronbach si el elemento se ha suprimido
Experiencia_desarrollo	19.23	22.740	.290	.797
Nivel_experiencia	19.48	22.059	.304	.797
Desarrollo_diagramas	19.32	22.413	.296	.797
Nivel_desarrollo_diagramas	19.27	22.799	.229	.799
Dominio_lenguaje_computacional	19.62	22.497	.189	.802
N1_Recuperacion_P1	19.26	23.117	.137	.802
N1_Recuperacion_P2	19.70	20.147	.727	.775
N1_Recuperacion_P3	19.49	20.059	.781	.773
N2_Comprension_P1	19.34	20.894	.702	.780
N2_Comprension_P2	19.33	20.804	.745	.779
N2_Comprension_P3	19.37	22.688	.192	.801
N3_Analisis_P1	19.88	21.911	.376	.793
N3_Analisis_P2	19.80	20.443	.691	.778
N3_Analisis_P3	19.54	22.487	.196	.802
N4_Aplicacion_P1	19.53	20.187	.729	.775
N4_Aplicacion_P2	19.80	23.432	.002	.811
N4_Aplicacion_P3	19.20	23.389	.083	.803
N5_Metacognitivo_P1	19.61	22.263	.239	.800
N5_Metacognitivo_P2	20.07	23.425	.068	.803
N5_Metacognitivo_P3	19.66	22.184	.256	.799
Recursos_motivacion	19.30	23.093	.118	.803
Diagnostico_inicio_materia	19.38	22.927	.129	.804
Algoritmos_graduales_complejidad	19.23	22.697	.306	.797
Calificaciones_destrezas	19.30	23.416	.027	.806
Competecias_metodologia	19.41	22.761	.159	.803
Competencias_soluciones_programacion	19.39	22.220	.299	.797
Competencias_Sistemas_Inf	19.39	22.155	.315	.796
Conoces_motivo_tutor	19.38	22.282	.289	.797
Interaccion_tutor	19.28	23.256	.081	.804
Ayudaron_asesorias	19.44	22.700	.168	.803

Nota: La tabla representa las estadísticas del total de elementos analizado mediante SPSS versión 28.

La columna de “**correlación total de elementos corregida**” muestra valores numéricos positivos lo que representa que cada uno de los ítems del instrumento se plantearon y estructuraron de manera correcta para recabar información de los estudiantes al momento de desarrollar algoritmos. Como lo indica George y Mallery (2003), cuando existen números negativos en este rubro se hace presente la inconsistencia de algún ítem. Los valores negativos representan que el

ítem se planteó de manera incorrecta y no tiene relación con los demás ítems del instrumento. Las preguntas de este instrumento contienen respuestas incorrectas representadas con 0 y respuestas correctas representadas con 1, es decir, a mayor acercamiento a valor 1 en la correlación de elementos mayor es el nivel de confianza y coherencia entre los ítems planteados en el instrumento. Esta columna permite identificar los ítems con valores bajos e incrementar el coeficiente del Alfa de Cronbach.

Finalmente, la tabla 20 muestra los estadísticos de la prueba en conjunto, mostrando la **Media** de las respuestas a los ítems con respuestas valor afirmativo. Respecto al valor de la **desviación estándar** es de sobresalir que el valor que se obtuvo fue inferior a 5, lo que representa un valor bajo de la dispersión de los valores de correlación de los ítems dando la pauta a un buen diseño del instrumento ya que los ítems categorizados en las diferentes dimensiones de la taxonomía de Marzano y Kendall (2007) proporcionaron valores con alto sentido de consistencia. La desviación estándar midió la variabilidad de las respuestas entre el censo poblacional.

Tabla 20.

Estadísticas de la escala.

Media	Varianza	Desviación estándar	N de elementos
20.14	23.647	4.863	30

Nota: La tabla representa las estadísticas de la escala analizado mediante SPSS versión 28.

Resultados Caracterización del Conocimiento

A continuación, se hace un análisis de los resultados obtenidos del instrumento de recolección de datos para la caracterización del conocimiento del grupo experimental respecto al desarrollo al desarrollo de algoritmos computacionales.

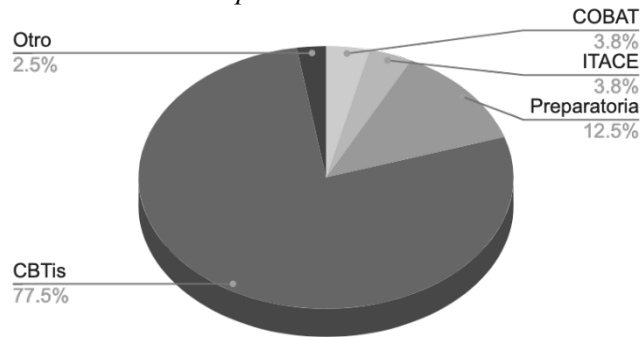
Los resultados se clasificaron de la siguiente manera:

a) *Datos generales de los estudiantes.*

En la muestra experimental a la que se aplicó el instrumento de caracterización de conocimientos de algoritmos y programación se observa que la edad promedio de los alumnos encuestados es de 19 años, respecto al estado civil el 94% manifiesta ser solteros mientras que el 6% comentó que tiene otro estado civil, respecto al sexo el 70% son hombres mientras que el 30% de la población son mujeres.

La figura 7 muestra los resultados del bachillerato estudiado por los estudiantes universitarios en donde se observa que el 77.5% de los encuestados estudiaron en un Centro de Bachillerato Tecnológico industrial y de servicios (CBTis), el 12.5% en una preparatoria, el 3.8% en el Colegio de Bachilleres de Tamaulipas (COBAT), el mismo porcentaje en el Instituto Tamaulipeco de Capacitación para el Empleo (ITACE), finalmente el 2.5% revela haber estudiado en otro bachillerato.

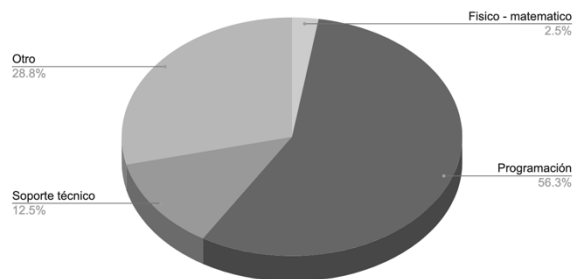
Figura 7.
Bachillerato estudiado por los estudiantes entrevistados.



Nota: El gráfico representa el bachillerato estudiado por los estudiantes entrevistados.

Respecto a la especialidad estudiada por los estudiantes encuestados, la figura 8 muestra que el 56.3% de los alumnos tienen especialidad de programación, el 28.8% tienen otro tipo de especialidad, el 12.5% estudiaron soporte técnico, mientras que el 2.5% manifiesta haber estudiado la especialidad físico – matemático.

Figura 8.
Especialidad estudiada por los estudiantes entrevistados.

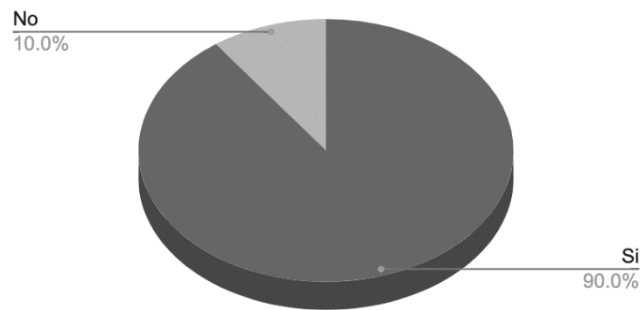


Nota: El gráfico representa la *especialidad estudiada* por los estudiantes entrevistados.

b) *Conocimientos en el desarrollo de software.*

En el ámbito de experiencia en el desarrollo de algoritmos computacionales, la figura 9 muestra que 90% tiene experiencia desarrollando algoritmos, mientras que el 10% manifiesta que no la tiene.

Figura 9.
Experiencia en el desarrollo de algoritmos.



Nota: El gráfico representa la experiencia de los estudiantes entrevistados sobre el desarrollo de algoritmos.

c) *Prácticas de programación.*

Respecto al análisis prácticas y programación, para validar el nivel de conocimiento adquirido por los entrevistados respecto al desarrollo de algoritmos y programación, se aplicó la taxonomía de Marzano y Kendall (2007) la cual clasifica en diferentes categorías el conocimiento de las personas y en esta investigación se utilizó para identificar el nivel de conocimiento de los estudiantes en la construcción de algoritmos computacionales a través de diagramas de flujo. Álvarez y Valdebenito (2017) identifican en su investigación los niveles de conocimiento de la taxonomía de Marzano las cuales son las siguientes:

1. Recuperación
2. Comprensión

3. Análisis
4. Aplicación
5. Metacognitivo
6. Evaluación

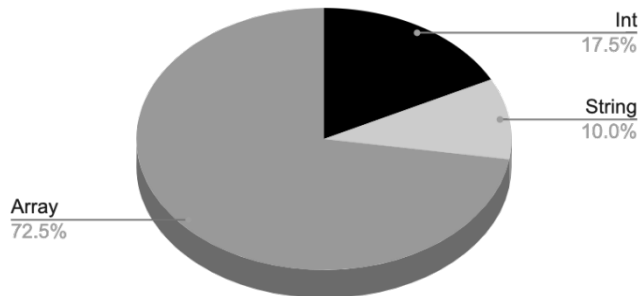
Para validar el nivel de conocimiento de los alumnos encuestados respecto al desarrollo de software, se diseñó un instrumento de recolección de datos validado por un grupo de expertos en el área de la programación computacional en donde se integraron ítems que identifican la capacidad de comprensión en el desarrollo de algoritmos computacionales y se clasificaron a partir de las categorías de la taxonomía de Marzano y Kendall (2007) en donde los resultados se muestran a continuación:

1. Desarrollo de software: Nivel de recuperación.

Para identificar el nivel de recuperación respecto al conocimiento, se determina la capacidad de registro, memorización e identificación de las estructuras de control al momento de diseñar algoritmos computacionales a través de diagramas de flujo, se cuestionó a los alumnos sobre *¿Cuál es la estructura de datos que permite múltiples valores organizados de forma indexada?* en donde el 72.5% respondió que es un vector (array), el 17.5% comentó que es un entero (int) y el 10% aseveró que es una cadena de texto (string). La figura 10 muestra los resultados obtenidos:

Figura 10.

¿Cuál es la estructura de datos que permite múltiples valores organizados de forma indexada?

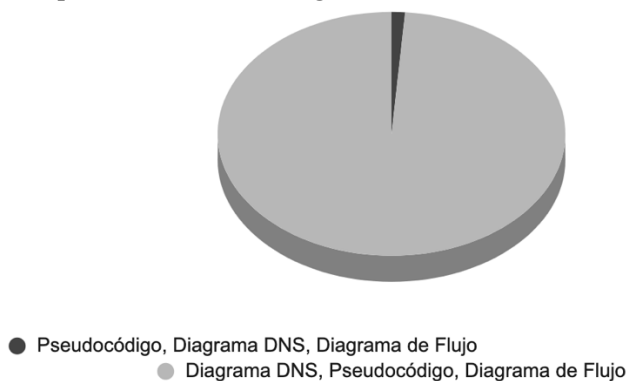


Nota: El gráfico representa el nivel de memorización de las estructuras de datos.

Por otro lado, se le presentó al alumno encuestado un diagrama de flujo donde se muestra un algoritmo y se solicitó *elegir la representación correcta en la construcción de algoritmos*, el 98.8% respondió diagrama DNS, pseudocódigo y diagrama de flujo y el 1.2% respondió pseudocódigo, diagrama DNS y diagrama de flujo. La figura 11 muestra los resultados del ítem:

Figura 11.

Elección de la opción correcta que enlista cada tipo de representación de un algoritmo.



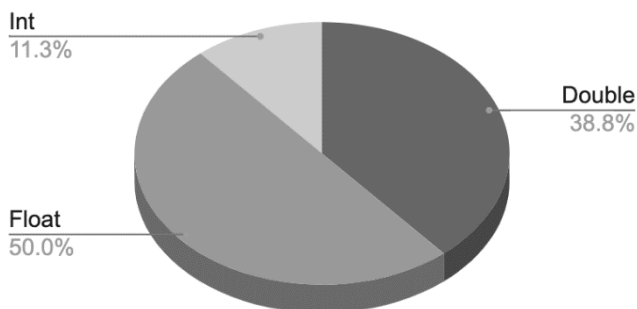
Nota: El gráfico representa el nivel de memorización de la representación de algoritmos.

Con los datos recabados en el *nivel de recuperación de la taxonomía de Marzano y Kendall (2007)* se muestra evidencia que el **85%** de los estudiantes encuestados dominan el registro, memorización e identificación de las estructuras de control al momento de diseñar algoritmos mediante diagramas de flujo y con ello se identifica que la mayoría de los encuestados se encuentran o superan la primera categoría de la taxonomía.

2. *Desarrollo de software: Nivel de comprensión.*

Para validar el nivel de comprensión de la taxonomía de Marzano y Kendall (2007) respecto a la interpretación, traducción, ordenación e ilustración de las estructuras de control al momento de diseñar algoritmos computacionales a través de diagramas de flujo, se solicitó a los estudiantes *identificar el tipo de dato que admite valores con punto decimal de 16 bits* en donde el 50% respondió doble (double), el 38.8% respondió flotante (float) y el 11.2% respondió entero (int). La figura 12 muestra los resultados del ítem:

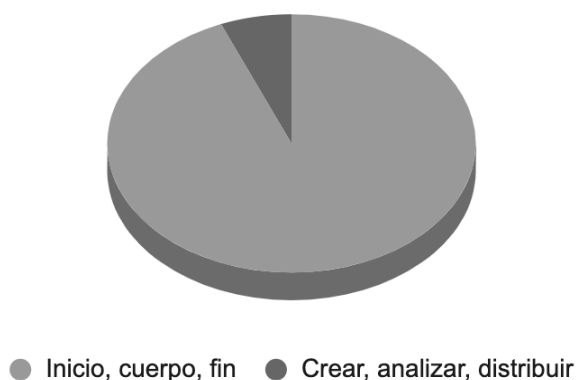
Figura 12.
Identificación el tipo de dato que admite valores con punto decimal de 16 bits.



Nota: El gráfico representa la identificación del tipo de dato que admite valores decimales de 16 bits.

Así mismo se les preguntó a los alumnos encuestados de acuerdo con sus conocimientos, *¿Cuáles es el procedimiento que se considera sobresaliente para la construcción de diagramas de flujo?*, el 93.7% respondió: Inicio, cuerpo y fin; mientras que el 6.3% respondió: crear, analizar y distribuir. La figura 13 muestra los resultados del ítem:

Figura 13.
Procedimiento sobresaliente para la construcción de diagramas de flujo.



Nota: El gráfico representa el procedimiento sobresaliente para la construcción de diagramas de flujo.

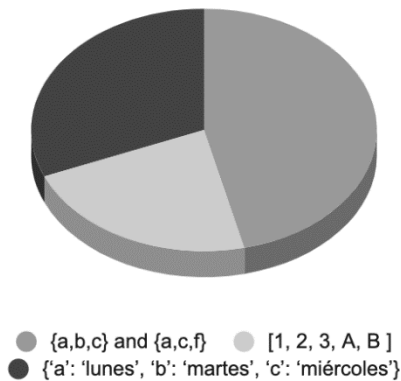
Con los datos recabados en el *nivel de comprensión de la taxonomía de Marzano y Kendall (2007)* se muestra evidencia que el **62%** de los estudiantes encuestados interpretan, traducen, describen y explican las estructuras de control al momento de diseñar algoritmos mediante diagramas de flujo y con ello se identifica que la mayoría de los encuestados se encuentran o superan la segunda categoría de la taxonomía.

3. Desarrollo de software: Nivel de análisis.

Para validar el nivel de análisis de la taxonomía de Marzano y Kendall (2007) respecto a la experimentación, comparación, discusión, ordenación y selección de las estructuras de control al momento de diseñar algoritmos computacionales a través de diagramas de flujo, se les mostró a

los estudiantes encuestados una serie de opciones de estructuras de control y se les cuestionó *¿Cuál de las opciones mostradas no es un arreglo (array)?*, en donde el 46.3% respondió: {a,b,c} and {a,c,f}, el 31.2% respondió: {'a': 'lunes', 'b': 'martes'} y el 22.5% respondió [1,2,3,A,B]. La figura 14 muestra los resultados del ítem:

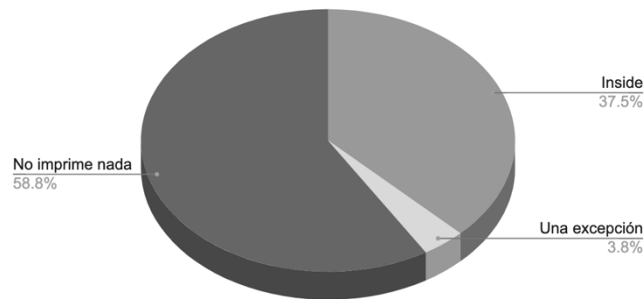
Figura 14.
Identificación de arreglos (array) .



Nota: El gráfico representa la identificación de los arreglos (array) en un bloque de opciones.

Así mismo se les presentó a los alumnos encuestados un diagrama de flujo y se les cuestionó *¿Cuál es la salida correcta del diagrama?*, el 58.8% respondió que no imprime nada, 37.5% seleccionó que imprime la cadena “inside” y finalmente el 3.8% respondió que la salida es una excepción. La figura 15 muestra los resultados del ítem:

Figura 15.
Identificación de la salida de un diagrama de flujo.



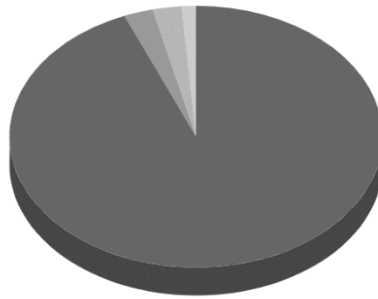
Nota: El gráfico representa la identificación de la salida de un diagrama de flujo.

Con los datos recabados en el *nivel de análisis de la taxonomía de Marzano y Kendall (2007)* se muestra evidencia que el **40.65%** de los estudiantes encuestados distinguen, analizan, diferencian, comparan, conectan y seleccionan las estructuras de control al momento de diseñar algoritmos mediante diagramas de flujo y con ello se identifica que menos de la mitad de la población encuestada se encuentran o superan la tercera categoría de la taxonomía. A partir de esta categoría el nivel de complejidad de los algoritmos sube y con ello se evidencia que los alumnos tienen carencias de los conocimientos al momento de desarrollar software.

4. Desarrollo de software: Nivel de aplicación.

Para validar el nivel de análisis de la taxonomía de Marzano y Kendall (2007) respecto a la aplicación, demostración, ilustración, transformación, comprobación y clasificación de las estructuras de control al momento de diseñar algoritmos computacionales a través de diagramas de flujo, se les cuestionó a los estudiantes encuestados de acuerdo con la estructura de declaración de variables ¿Cuál es la manera correcta de declaración?, a lo que el 93.7% respondió: entero $i = 10$, el 2.5% respondió: entero $10 = i+1$; con el mismo porcentaje respondieron: entero $i:10$, finalmente el 1.3% respondió: $\text{int } i = 1$. La figura 16 muestra los resultados del ítem:

Figura 16.
Identificación de la declaración correcta de variables.

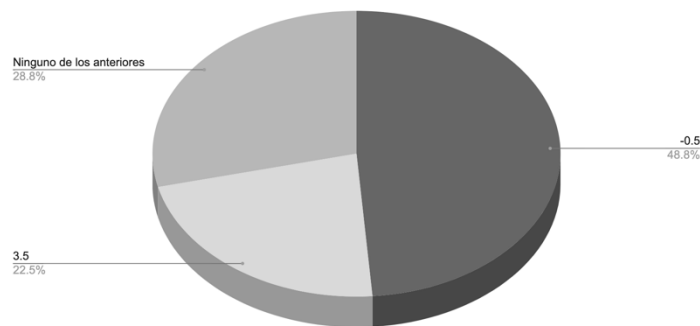


● entero i = 10 ● entero 10 = i+1 ● entero i:10 ● int i = 1;

Nota: El gráfico representa la declaración correcta de variables en el desarrollo de algoritmos.

Por otro lado, se les cuestionó a los alumnos de acuerdo con un algoritmo mostrado en el ítem *¿Cuál es la impresión de salida del algoritmo?*, el 48.8% respondió que imprime -0.5, el 28.7% respondió que ninguna de las opciones es correcta y finalmente el 22.5% respondió que la salida del algoritmo es 3.5. La figura 17 muestra los resultados del ítem:

Figura 17.
Identificación de la salida de un algoritmo.



Nota: El gráfico representa las salidas seleccionadas de un algoritmo.

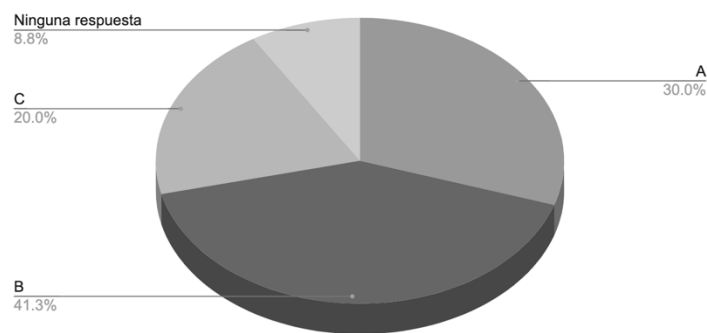
Con los datos recabados en el *nivel de aplicación de la taxonomía de Marzano y Kendall (2007)* se muestra evidencia que el **51.9%** de los estudiantes encuestados aplican, demuestran,

programan, transforman, resuelven y comprueban las estructuras de control al momento de diseñar algoritmos mediante diagramas de flujo y con ello se identifica que un poco mas de la mitad de la población encuestada se encuentran o superan la cuarta categoría de la taxonomía.

5. *Desarrollo de software: Nivel metacognitivo.*

Para validar el nivel metacognitivo de la taxonomía de Marzano y Kendall (2007) respecto a la planeación, diseño, construcción, integración, creación, desarrollo y modificación de las estructuras de control al momento de diseñar algoritmos computacionales a través de diagramas de flujo, se les mostró a los estudiantes encuestados una serie de opciones con distintos diagramas de flujo para posteriormente cuestionarles de acuerdo a su experiencia ¿Cuál es la manera correcta de sumar dos números en un diagrama de flujo? En donde el 41.2% seleccionó la opción B, el 30% seleccionó la opción A, el 20% seleccionó la opción C, mientras que el 8.8 % seleccionó que ninguna de las opciones es correcta. La figura 18 muestra los resultados del ítem:

Figura 18.
Identificación de la suma correcta de variables.



Nota: El gráfico representa la suma correcta de variables.

Con los datos recabados en el *nivel metacognitivo de la taxonomía de Marzano y Kendall (2007)* se muestra evidencia que el **46.9%** de los estudiantes encuestados planean, diseñan,

construyen, integran, crean, desarrollan y modifican estructuras de control al momento de diseñar algoritmos mediante diagramas de flujo y con ello se identifica que menos de la mitad de la población encuestada se encuentran o superan la quinta categoría de la taxonomía.

Haciendo una reflexión de los hallazgos encontrados al momento de aplicar el primer instrumento de recolección de datos en el cual se hace un reconocimiento y caracterización de los aprendizajes de los alumnos de nuevo ingreso de la UPV respecto al desarrollo de algoritmos a través de diagramas de flujo, se percibe que gran parte de la población no tiene los conocimientos básicos para la construcción de software, lo que marca la pauta para la implementación del modelo de ITS ALGO-ITS para el apoyo en el aprendizaje de algoritmos computacionales.

La tabla 21 muestra los aprendizajes de los estudiantes encuestados en donde se reflejan los conocimientos relacionados al desarrollo de software a través de la taxonomía de Marzano y Kendall (2007) categorizando el nivel de dominio en la programación computacional, se puede identificar que la gran parte de los estudiantes del grupo experimental se encuentran en la primera categoría denominada nivel de recuperación, por otro lado se aprecia claramente que conforme avanzan los niveles de complejidad de los algoritmos, menos de la mitad de los alumnos encuestados alcanzan a mantener el nivel de aprendizaje conforme el requerimiento sea mas alto.

Tabla 21.

Nivel de conocimiento de los estudiantes encuestados en cada una de las categorías de la taxonomía de Marzano y Kendall (2007).

Categoría de Marzano y Kendall: Desarrollo de software	Nivel de dominio
Primer nivel: Recuperación.	85%
Segundo nivel: Comprensión.	62%
Tercer nivel: Análisis.	40.65%
Cuarto nivel: Aplicación.	51.9%
Quinto nivel: Metacognitivo.	46.9%

Nota: La tabla representa el nivel de dominio de cada uno de los sujetos del grupo experimental en cada una de las categorías de la taxonomía de Marzano y Kendall (2007) respecto al desarrollo de software.

Tomando como referencia que el 90% de los alumnos encuestados respondieron que tienen experiencia desarrollando algoritmos computacionales, se identificó que no hay evidencia de nivel análisis y metacognitivo para generar soluciones algorítmicas. Los resultados se muestran a través de las categorías de la taxonomía de Marzano y Kendall (2007).

d) Recursos y materiales didácticos orientados a los alumnos.

El programa educativo de ITI tiene en sus asignaturas los programas de estudio y manuales de asignatura los cuales establecen de manera clara el desempeño que se espera obtener por parte de los alumnos en cada asignatura, por ello se les cuestionó a los estudiantes si los recursos didácticos y tecnológicos empleados en clase influenciaron la motivación e interés en la asignatura de algoritmos, a lo que el 83.8% respondió afirmativamente mientras que el 16.2% respondió de manera negativa.

Respecto al diagnóstico del estudiante al inicio de la asignatura, el uso de los recursos tecnológicos empleados por los profesores para impartir las asignaturas, la realización de prácticas graduales en complejidad y el reflejo de las calificaciones respecto a las habilidades y conocimiento obtenido, la tabla 22 muestra los resultados de los cuestionamientos del censo de los alumnos sujetos de la investigación.

Tabla 22.

Resultados de cuestionamientos de la muestra.

Cuestionamiento	Si	No
¿Los recursos didácticos y tecnológicos empleados en clase influenciaron tu motivación e interés por la misma?	84%	16%
Al inicio de cada materia, ¿se realizó un diagnóstico de tus conocimientos previos sobre cada una de ellas?	75%	25%
En la materia de algoritmos, ¿realizaste prácticas y ejercicios graduales en complejidad hasta llegar a un proyecto o un producto?	88.8%	11.2%
¿Las calificaciones que obtuviste son el reflejo de tus conocimientos, habilidades, destrezas y actitudes?	81.3%	18.7%

Nota: Esta tabla representa el concentrado del diagnóstico general aplicado a los estudiantes entrevistados al inicio de una asignatura.

e) *Estudios, conocimientos y habilidades de los alumnos.*

Sobre el uso de metodologías para implementar soluciones eficientes el 71.2% de los alumnos respondió que no usa mientras que el 28.8% manifestó que, si las lleva a cabo, la tabla 23 muestra los resultados de las técnicas y habilidades de los estudiantes encuestados que llevan a cabo al momento de estudiar una materia.

Tabla 23.

Resultados de los estudios, conocimientos y habilidades.

Cuestionamiento	Si	No
¿Aplicas metodologías para implementar soluciones eficientes?	71.2%	28.8%
¿Desarrollas soluciones a problemas de programación para diseñar soluciones eficientes?	73.8%	26.2%
¿Defines y diseñas sistemas de información de acuerdo con los requerimientos solicitados?	72.5%	27.5%

Nota: Esta tabla representa los resultados de las metodologías y propuestas de soluciones que los alumnos encuestados llevan a cabo al momento de desarrollar algoritmos computacionales.

f) Tutorías y asesorías.

La UPV al inicio de cada periodo asigna un tutor para el seguimiento académico y personal de cada uno de los tutorados asignados, bajo esta premisa se les cuestionó a los alumnos sobre este proceso y la tabla 24 muestra los resultados de los cuestionamientos:

Tabla 24.

Resultados de actividades de tutorías y asesorías.

Cuestionamiento	Si	No
¿Conoces el motivo de la asignación de un tutor?	75%	25%
¿Has interactuado con tu tutor académico?	83.3%	13.2%
¿Has tenido la necesidad de una asesoría académica?	40%	60%
Las asesorías que hallas necesitado, ¿te ayudaron a solucionar tus dudas académicas para mejorar el aprendizaje?	71.3%	28.7%

Nota: Esta tabla representa los resultados de los cuestionamientos a los alumnos encuestados sobre la asignación de tutores y necesidades de sesiones de tutorías y asesorías.

Respecto a los alumnos que necesitaron asesorías académicas para solucionar dudas en alguna asignatura y mejorar el aprendizaje, la tabla 25 muestra los resultados sobre las temáticas que los alumnos necesitaron mas asesorías académicas.

Tabla 25.

Temáticas que los alumnos encuestados solicitaron asesorías académicas.

Temática	Porcentaje de alumnos de solicitud de asesorías
Programación	77.5%
Metodologías de la programación	13.8%
Metodologías de desarrollo	6.2%
Algoritmos	2.5%

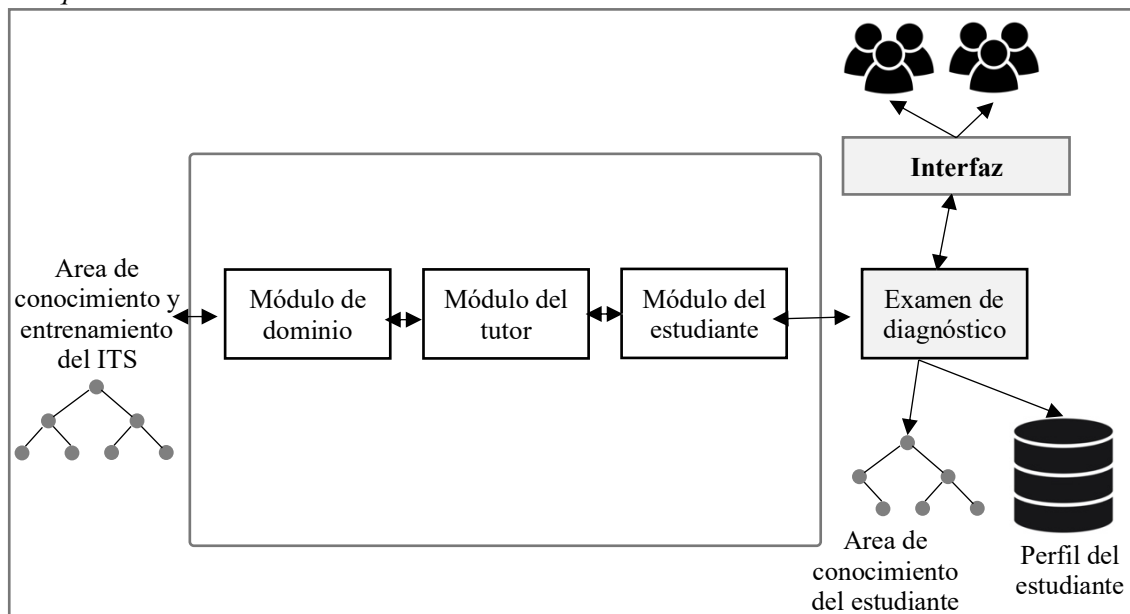
Nota: Esta tabla representa las temáticas en la que los estudiantes encuestados necesitaron asesorías académicas para mejorar el rendimiento académico.

Modelo Propuesto: ALGO-ITS

Como resultado de la fase 2 se implementó un modelo de ITS para el apoyo en la construcción de algoritmos computacionales el cual, a través de la IA y entrenamiento mediante redes neuronales de aprendizaje profundo, se logra obtener un ITS en la UPV denominado ALGO-ITS que permitirá dar soporte personalizado a cada uno de los estudiantes del grupo experimental.

La figura 19 muestra la arquitectura del modelo propuesto denominado ALGO-ITS el cual es un ITS desarrollado para soporte, ayuda y mejoramiento del conocimiento en la construcción de algoritmos computacionales. ALGO-ITS es un sistema informático innovador desarrollado para ser utilizado en la UPV, es una aplicación web integrada en un módulo de la plataforma institucional LMS MOODLE con acceso restringido al grupo experimental para la construcción de algoritmos computacionales. El sistema es único existente dentro del subsistema de universidades politécnicas y tecnológicas (Rodríguez, 2021).

Figura 19.
Arquitectura de ALGO-ITS.



Nota: El gráfico representa la arquitectura del ALGO-ITS implementado en el grupo experimental de alumnos de la Universidad Politécnica de Victoria.

Módulo de dominio: Representa el conocimiento relacionado en la construcción de algoritmos en forma de árbol de decisión en donde cada grafo representa capítulos y las aristas las rutas de aprendizaje que se van obteniendo.

Módulo de tutor: Representa el sistema adaptativo y el estilo de aprendizaje del estudiante basado en la teoría de la psicología cognitiva la cual selecciona una adecuada intervención en función del conocimiento, la caracterización del alumno permite a la red neuronal entrenada presentar el nivel de complejidad de los algoritmos computacionales a solucionar.

Módulo del estudiante: Es el responsable de evaluar el desempeño del estudiante para determinar habilidades cognitivas y de lógica al momento de construir algoritmos. El ITS identifica el conocimiento del alumno a través de un examen de diagnóstico inicial y almacena su evolución en un perfil del estudiante.

Se adoptó un enfoque iterativo para la evaluación y las mejoras del ITS. Después de implementar el sistema durante el cuatrimestre de enero – abril 2021, se realizaron mejoras basadas en las respuestas recibidas de los estudiantes que utilizaron el sistema. Este sistema mejorado se implementó después durante el cuatrimestre mayo – agosto 2021 para posteriormente ser evaluado utilizando los mismos métodos que en el primer cuatrimestre. El sistema se mejoró utilizando los comentarios recibidos de la segunda evaluación.

ALGO-ITS al final de su operación proporciona información del conocimiento de manera personalizada de cada uno de los estudiantes del grupo experimental para ello hace el cálculo del número de errores cometidos al momento de construir algoritmos, la cantidad de veces de soporte solicitado al ITS y el tiempo invertido al momento de construir y desarrollar algoritmos. Con estas variables (errores, soporte y tiempo) el ITS calcula el nivel de complejidad (bajo, medio o alto) de las siguientes construcciones de algoritmos a resolver. Las acciones del ITS están relacionadas con

un componente pedagógico que apunta a ofrecer la comprensión del algoritmo a través de la propuesta de la construcción de un diagrama de flujo eficaz.

Validación del Instrumento Uso del Modelo ALGO-ITS

La tabla 26 muestra el coeficiente Alfa de Cronbach total del instrumento de evaluación del conocimiento del grupo experimental posterior al uso del modelo ALGO-ITS, los resultados muestran que el coeficiente mostrado es de valor de **0.823** considerado en la escala de George y Mallery (2003) como bueno y confiable. El número de elementos corresponde la cantidad de preguntas consideradas en el instrumento.

Tabla 26.

Estadísticas de fiabilidad

Alfa de Cronbach	Alfa de Cronbach basada en elementos estandarizados	N de elementos
0.823	0.841	15

Nota: La tabla representa el nivel de confianza y fiabilidad a través del Alfa de Cronbach analizado mediante SPSS versión 28.

Por otro lado, la tabla 27 muestra los indicadores del comportamiento de cada uno de los ítems del instrumento de evaluación del conocimiento posterior al uso del modelo ALGO – ITS. La columna ítem enlista cada uno de los reactivos que se presentaron en forma de ejercicios a los alumnos encuestados del censo poblacional, la columna **Media** representa el índice de dificultad de cada ítem respecto a las posibles respuestas de opción múltiple. Finalmente la **desviación estándar** mide el grado de dispersión de los valores observables alrededor de la media lo que significa que al tener un valor alto la solución del ítem tienen un nivel de complejidad gradual en aumento.

Tabla 27.

Estadísticas del elemento.

Ítem	Media	Desviación estándar	N
N1_Recuperacion_P1	.95	.226	94
N1_Recuperacion_P2	.17	.378	94
N1_Recuperacion_P3	.51	.503	94
N2_Comprension_P1	.85	.358	94
N2_Comprension_P2	.93	.264	94
N2_Comprension_P3	.88	.323	94
N3_Analisis_P1	.83	.378	94
N3_Analisis_P2	.31	.464	94
N3_Analisis_P3	.57	.497	94
N4_Aplicacion_P1	.44	.499	94
N4_Aplicacion_P2	.31	.464	94
N4_Aplicacion_P3	.91	.281	94
N5_Metacognitivo_P1	.93	.264	94
N5_Metacognitivo_P2	.30	.460	94
N5_Metacognitivo_P3	.88	.323	94

Nota: La tabla representa las estadísticas del elemento analizado mediante SPSS versión 28.

Con base a los resultados de la tabla 27, la “**desviación estándar**” representa el nivel de dispersión de los datos respecto a la media, Mientras la “**Media**” representa el índice de dificultad de cada ítem. A mayor aproximación al valor 1, mayor es el nivel de complejidad. Los ítems: N1_Recuperacion_P3, N3_Analisis_P3 y N4_Aplicacion_P1, representan mayor desviación estándar lo cual significa que el nivel de dispersión en estos ítems es alto ya que sus múltiples soluciones a los ejercicios de algoritmos para estas entidades son múltiples y que de acuerdo con el conocimiento, nivel de experiencia, motivación y asesoría que tengan los alumnos estarán en condiciones de implementar una solución eficiente posterior al uso del modelo de ITS.

La tabla 28 refleja el resumen de las estadísticas proporcionadas por el software SPSS donde se muestran las medias de los elementos el cual es el promedio de los ítems de la tabla 26. Se observa la **varianza** que es la medida de dispersión que representa la movilidad de los datos arrojados por los ítems, la **covarianza** representa el grado de variación de ítems aleatorios respecto a la media, y finalmente se muestra la correlación que es la asociación de los ítems evaluando las tendencias creciente y decreciente de los resultados de los ítems.

Tabla 28.

Estadísticas de elemento resumen.

	Media	Mínimo	Máximo	Rango	Máximo / Mínimo	Varianza	N de elementos
Medias de elemento	.651	.170	.947	.777	5.563	.082	15
Varianzas de elemento	.152	-.051	.253	.202	4.962	.005	15
Covarianzas entre elementos	.036	-.039	.216	.255	-5.480	.002	15
Correlaciones entre elementos	.260	-.170	1.000	1.170	-5.866	.078	15

Nota: La tabla representa las estadísticas del elemento resumen analizado mediante SPSS versión 28.

La tabla 29 muestra las estadísticas totales del elemento en donde se muestra la columna “Alfa de Cronbach si el elemento se ha suprimido” la cual permite identificar los ítems que se pueden eliminar para incrementar el nivel de confianza del instrumento, si el ítem “N3_Analisis_P3” es eliminado el nivel de confianza se eleva a .848 pero no fue eliminado debido a que el ítem forma parte de la evaluación del cuarto nivel de la taxonomía de Marzano y Kendall (2007) de aplicación para obtener el conocimiento de los alumnos.

Tabla 29.

Estadísticas del total de elementos.

	Media de escala si el elemento se ha suprimido	Varianza de escala si el elemento se ha suprimido	Correlación total de elementos corregida	Alfa de Cronbach si el elemento se ha suprimido
N1_Recuperacion_P1	8.82	9.096	.507	.813
N1_Recuperacion_P2	9.60	8.846	.378	.816
N1_Recuperacion_P3	9.26	7.827	.625	.798
N2_Comprension_P1	8.91	8.229	.721	.795
N2_Comprension_P2	8.84	9.340	.265	.822
N2_Comprension_P3	8.88	9.309	.215	.825
N3_Analisis_P1	8.94	8.146	.718	.794
N3_Analisis_P2	9.46	7.993	.620	.798
N3_Analisis_P3	9.19	9.554	.012	.848
N4_Aplicacion_P1	9.33	9.213	.124	.839
N4_Aplicacion_P2	9.46	7.993	.620	.798
N4_Aplicacion_P3	8.85	9.289	.274	.821
N5_Metacognitivo_P1	8.84	8.845	.587	.808
N5_Metacognitivo_P2	9.47	8.015	.619	.799
N5_Metacognitivo_P3	8.88	8.470	.671	.800

Nota: La tabla representa las estadísticas del total de elementos analizado mediante SPSS versión 28.

Finalmente, la tabla 30 representa las estadísticas de la prueba en conjunto.

Tabla 30.
Estadísticas de la escala.

Media	Varianza	Desviación estándar	N de elementos
9.77	9.837	3.136	15

Nota: La tabla representa las estadísticas de la escala analizado mediante SPSS versión 28.

Las estadísticas de la tabla 29 reflejan la “**Media**” la cual sus valores bajaron en relación con el instrumento de caracterización, este valor representa el valor de las respuestas incorrectas a los ítems lo cual significa que el conocimiento se ha incrementado posterior al uso del modelo ALGO – ITS; por otro lado el valor de la “**Desviación estándar**” igualmente se decremento en comparación a los resultados de validación del primer instrumento ya que refleja que los valores de correlación de los ítems tienen un bajo margen de error y gran correlación entre las diferentes categorías de la taxonomía de Marzano y Kendall (2007) que es la utilizada para diseñar y clasificar los niveles de dominio de la solución de algoritmos presentados a los estudiantes en la encuesta.

Modelo ALGO-ITS: Evaluación del Conocimiento

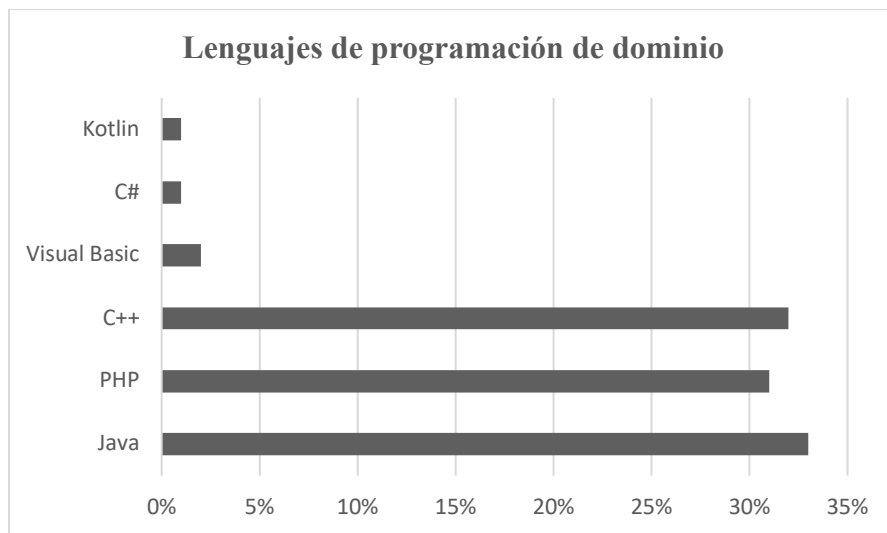
A continuación, se hace un análisis de los resultados obtenidos del instrumento de recolección de datos para la evaluación del conocimiento del grupo experimental respecto al desarrollo de algoritmos, posterior de haber utilizado el modelo ALGO-ITS de apoyo en el aprendizaje en el desarrollo de software.

La finalidad del instrumento es evaluar el aprendizaje a la construcción de algoritmos computacionales a través de diagramas de flujo, los resultados se clasificaron de la siguiente manera:

a. *Conocimientos sobre el desarrollo de software.*

En el ámbito de experiencia en el desarrollo de algoritmos computacionales, se les cuestionó a los alumnos sobre su experiencia y nivel de dominio de lenguajes de programación utilizados para el desarrollo de software, la mayoría de los alumnos manifiesta gran dominio de los lenguajes Java, C++ y PHP, la figura 20 muestra los resultados del ítem.

Figura 20.
Ítem relacionado a los conocimientos de lenguajes de programación



Nota: El gráfico representa los conocimientos de dominio de lenguajes de programación de parte del grupo experimental.

Para el análisis de prácticas y programación para validar el nivel de conocimiento adquirido respecto al desarrollo de algoritmos y programación, al igual que al primer instrumento de recolección de datos, se aplicó la taxonomía de Marzano y Kendall (2007) para clasificar las categorías del conocimiento en el desarrollo de software a través de algoritmos computacionales e identificar el nivel de dominio

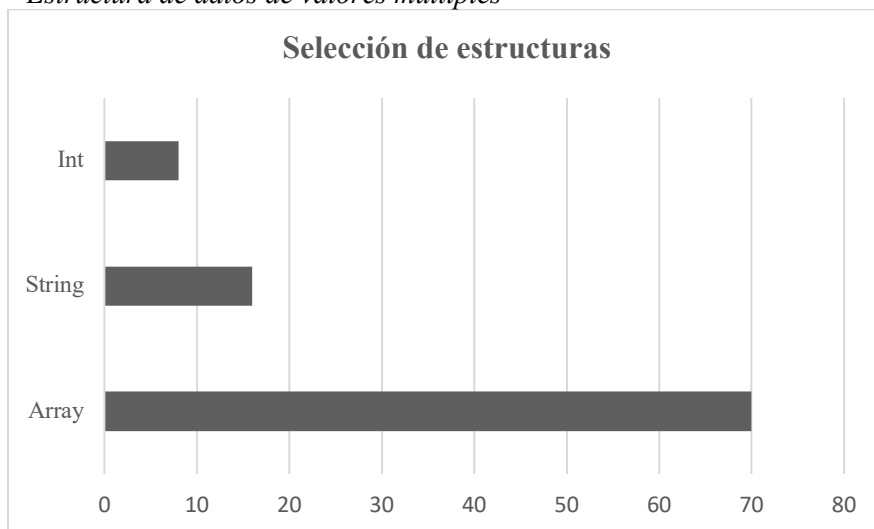
El grupo experimental una vez utilizado el modelo ALGO-ITS se analizó su nivel de dominio en el desarrollo de software. Para certificar el nivel de conocimiento de los alumnos encuestados respecto al desarrollo de software, se implementó un instrumento validado por un

grupo de expertos en el área de programación computacional en donde los ítems tienen la capacidad de comprensión en el desarrollo de algoritmos computacionales y se clasificaron a partir de las categorías de la taxonomía de Marzano y Kendall (2007).

b. Desarrollo de software: Nivel de recuperación.

Para identificar el nivel de recuperación respecto al conocimiento, se determina la capacidad de registro, memorización e identificación de las estructuras de control al momento de diseñar algoritmos computacionales a través de diagramas de flujo, se cuestionó a los alumnos sobre *¿Cuál es la estructura de datos que permite múltiples valores organizados de forma indexada?* en donde el 70 alumnos (75%) respondieron vector (array), 16 estudiantes representando (17%) aseveró que es una cadena de texto (string) mientras que 8 alumnos (8%) comentó que es un entero (int). La figura 21 muestra los resultados obtenidos:

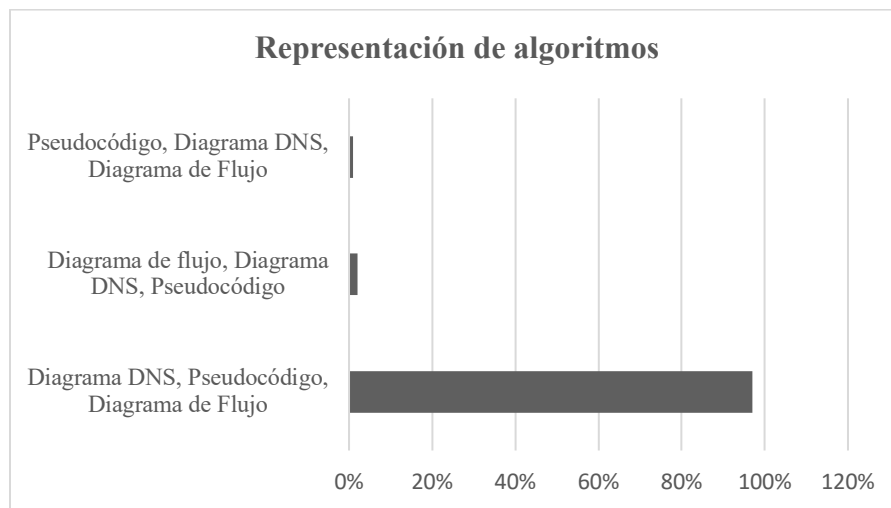
Figura 21.
Estructura de datos de valores múltiples



Nota: El gráfico representa la estructura de datos de valores múltiples de algoritmos.

Por otro lado, se le presentó al alumno encuestado un diagrama de flujo donde se muestra un algoritmo y se solicitó *elegir la representación correcta en la construcción de algoritmos*, el 98.8% respondió diagrama DNS, pseudocódigo y diagrama de flujo y el 1.2% respondió pseudocódigo, diagrama DNS y diagrama de flujo. La figura 22 muestra los resultados del ítem:

Figura 22.
Representación gráfica de algoritmos



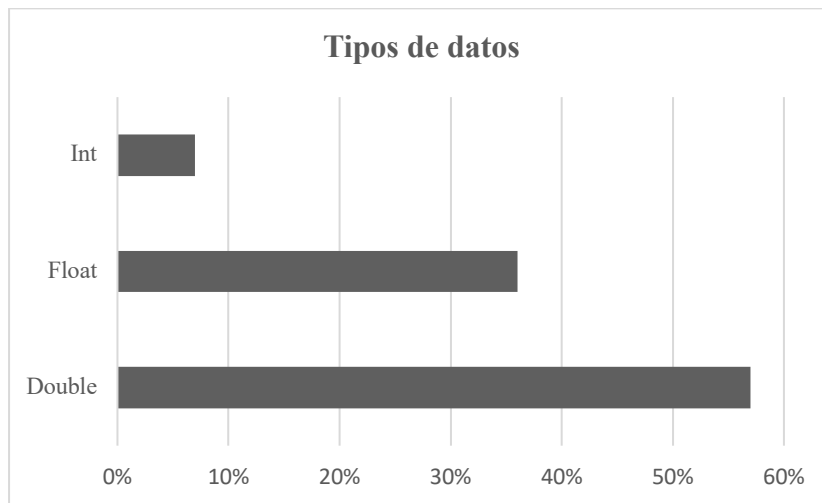
Nota: El gráfico representa la representación gráfica en la construcción de algoritmo.

Con los datos recabados en el *nivel de recuperación de la taxonomía de Marzano y Kendall (2007)* se muestra evidencia que el **86%** de los estudiantes encuestados dominan el registro, memorización e identificación de las estructuras de control al momento de diseñar algoritmos mediante diagramas de flujo y con ello se identifica que la mayoría de los encuestados se encuentran o superan la primera categoría de la taxonomía.

c. *Desarrollo de software: Nivel de comprensión.*

Para validar el nivel de comprensión de la taxonomía de Marzano y Kendall (2007) respecto a la interpretación, traducción, ordenación e ilustración de las estructuras de control al momento de diseñar algoritmos computacionales a través de diagramas de flujo, se solicitó a los alumnos *identificar el tipo de dato que admite valores con punto decimal de 16 bits* en donde el 57% respondió doble (double), el 36% respondió flotante (float) y el 7% respondió entero (int). La figura 23 muestra los resultados del ítem:

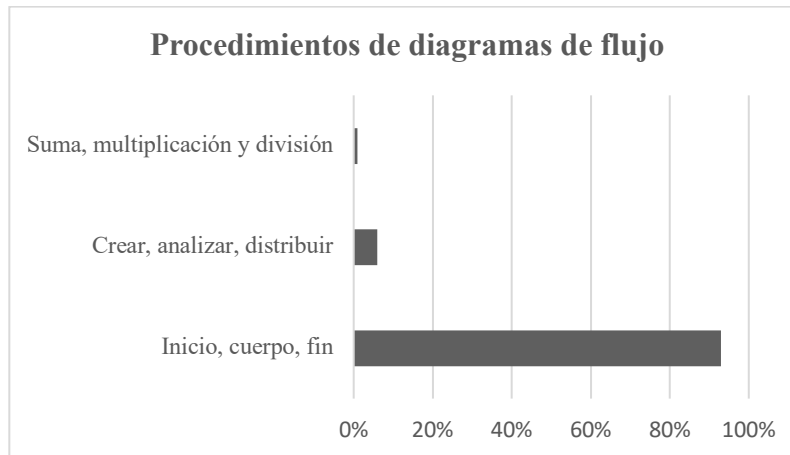
Figura 23.
Tipos de datos en el desarrollo de algoritmos.



Nota: El gráfico representa el nivel de dominio en la identificación de tipos de datos.

Así mismo se les preguntó a los alumnos encuestados de acuerdo con sus conocimientos, *¿Cuáles es el procedimiento que se considera sobresaliente para la construcción de diagramas de flujo?*, el 93% respondió: Inicio, cuerpo y fin, el 7% respondió: crear, analizar y distribuir, finalmente el 1% respondió suma, multiplicación y división. La figura 24 muestra los resultados del ítem:

Figura 24.
Procedimientos en la construcción de diagramas de flujo



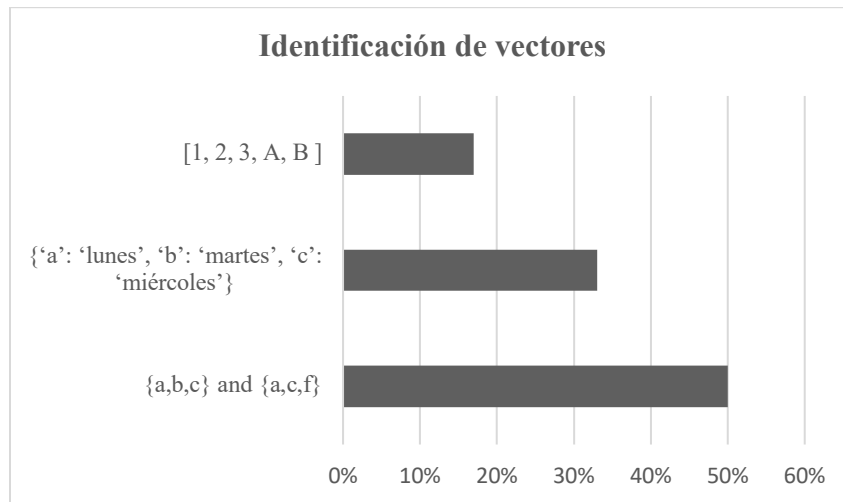
Nota: El gráfico representa el procedimiento sobresaliente para la construcción de diagramas de flujo.

Con los datos recabados en el *nivel de comprensión de la taxonomía de Marzano y Kendall (2007)* se muestra evidencia que el **75%** de los estudiantes encuestados interpretan, traducen, describen y explican las estructuras de control al momento de diseñar algoritmos mediante diagramas de flujo y con ello se identifica que la mayoría de los encuestados se encuentran o superan la segunda categoría de la taxonomía.

d. Desarrollo de software: Nivel de análisis.

Para validar el nivel de análisis de la taxonomía de Marzano y Kendall (2007) respecto a la experimentación, comparación, discusión, ordenación y selección de las estructuras de control al momento de diseñar algoritmos computacionales a través de diagramas de flujo, se les mostró a los estudiantes encuestados una serie de opciones de estructuras de control y se les cuestionó *¿Cuál de las opciones mostradas no es un arreglo (array)?*, en donde el 50% respondió: {a,b,c} and {a,c,f}, el 33% respondió: {'a': 'lunes', 'b': 'martes'} y el 17% respondió [1,2,3,A,B]. La figura 25 muestra los resultados del ítem:

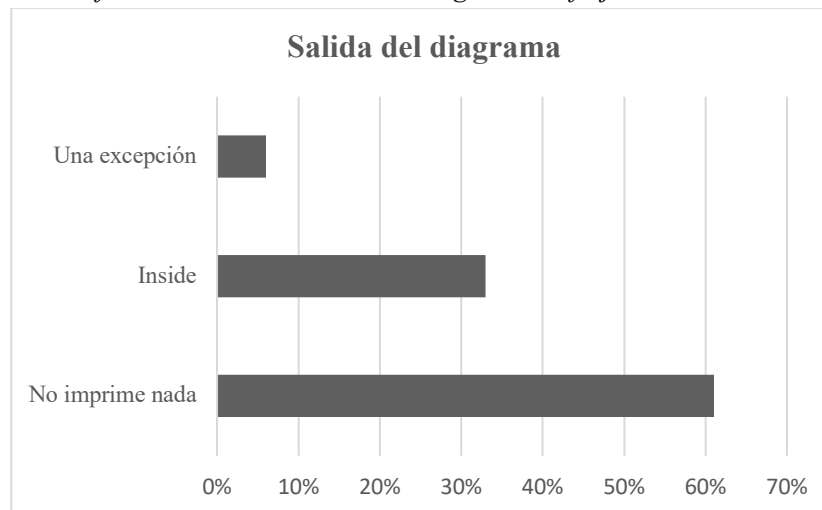
Figura 25.
Identificación de arreglos (array).



Nota: El gráfico representa la identificación de los arreglos (array) en un bloque de opciones.

Así mismo se les presentó a los alumnos encuestados un diagrama de flujo y se les cuestionó *¿Cuál es la salida correcta del diagrama?*, el 61% respondió que no imprime nada, 33% seleccionó que imprime la cadena “inside” y finalmente el 6% respondió que la salida es una excepción. La figura 26 muestra los resultados del ítem:

Figura 26.
Identificación de la salida de un diagrama de flujo.



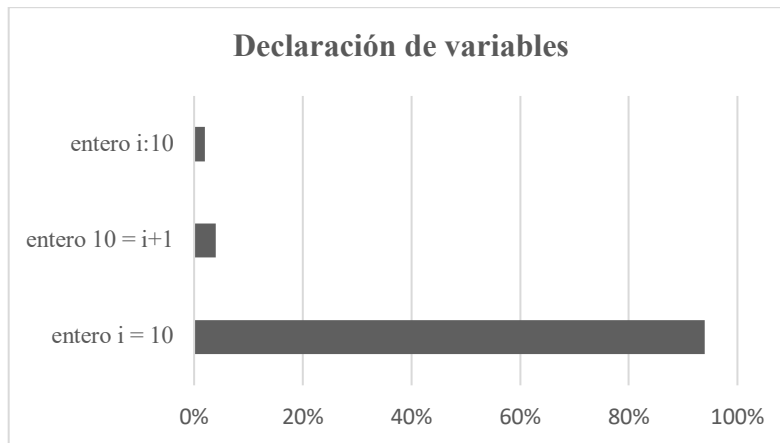
Nota: El gráfico representa la identificación de la salida de un diagrama de flujo.

Con los datos recabados en el *nivel de análisis de la taxonomía de Marzano y Kendall (2007)* se muestra evidencia que el **55%** de los estudiantes encuestados distinguen, analizan, diferencian, comparan, conectan y seleccionan las estructuras de control al momento de diseñar algoritmos mediante diagramas de flujo y con ello se identifica que menos de la mitad de la población encuestada se encuentran o superan la tercera categoría de la taxonomía. A partir de esta categoría el nivel de complejidad de los algoritmos sube y con ello se evidencia que los alumnos están presentando mejoras en el conocimiento del desarrollo de algoritmos posterior al uso del modelo ALGO-ITS.

e. Desarrollo de software: Nivel de aplicación.

Para validar el nivel de análisis de la taxonomía de Marzano y Kendall (2007) respecto a la aplicación, demostración, ilustración, transformación, comprobación y clasificación de las estructuras de control al momento de diseñar algoritmos computacionales a través de diagramas de flujo, se les cuestionó a los estudiantes de acuerdo con la estructura de declaración de variables ¿Cuál es la manera correcta de declaración?, a lo que el 94% respondió: entero $i = 10$, el 4% respondió: entero $10 = i+1$, finalmente el 4% respondió entero $i:10$. La figura 27 muestra los resultados del ítem:

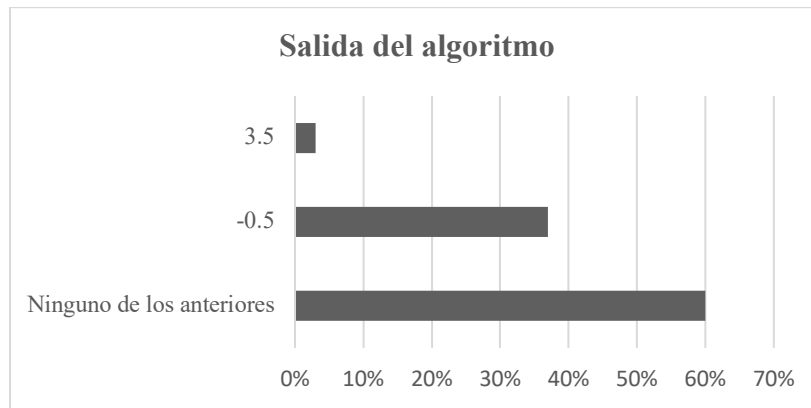
Figura 27.
Identificación de la declaración correcta de variables.



Nota: El gráfico representa la declaración correcta de variables en el desarrollo de algoritmos.

Por otro lado, se les cuestionó a los estudiantes de acuerdo con un algoritmo mostrado en el ítem *¿Cuál es la impresión de salida del algoritmo?*, el 60% respondió que ninguna de las opciones es correcta, el 37% respondió que imprime -0.5 y finalmente el 3% respondió que la salida del algoritmo es 3.5. La figura 28 muestra los resultados del ítem:

Figura 28.
Identificación de la salida de un algoritmo.



Nota: El gráfico representa las salidas seleccionadas de un algoritmo.

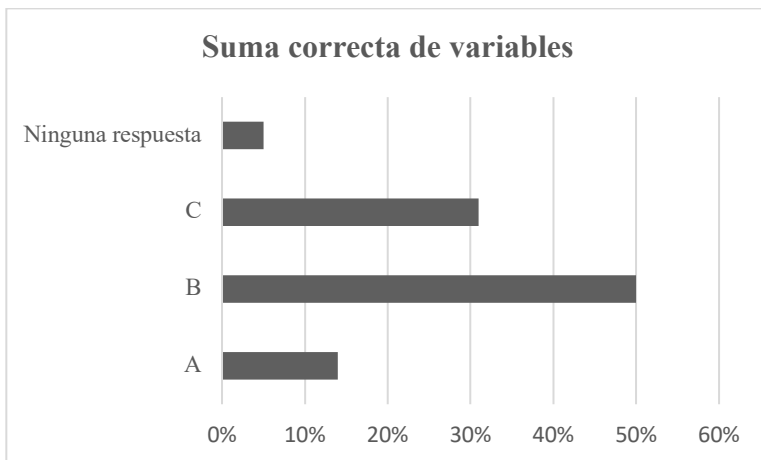
Con los datos recabados en el *nivel de aplicación de la taxonomía de Marzano y Kendall (2007)* se muestra evidencia que el **77%** de los estudiantes encuestados aplican, demuestran, programan, transforman, resuelven y comprueban las estructuras de control al momento de diseñar algoritmos mediante diagramas de flujo y con ello se identifica que un poco más de la mitad de la población encuestada se encuentran o superan la cuarta categoría de la taxonomía.

f. Desarrollo de software: Nivel metacognitivo.

Para validar el nivel metacognitivo de la taxonomía de Marzano y Kendall (2007) respecto a la planeación, diseño, construcción, integración, creación, desarrollo y modificación de las estructuras de control al momento de diseñar algoritmos computacionales a través de diagramas de flujo, se les mostró a los estudiantes encuestados una serie de opciones con distintos diagramas de flujo para posteriormente cuestionarles de acuerdo a su experiencia ¿Cuál es la manera correcta de sumar dos números en un diagrama de flujo? En donde el 50% seleccionó la opción B, el 31%

seleccionó la opción C, el 14% seleccionó la opción A, mientras que el 5 % seleccionó que ninguna de las opciones es correcta. La figura 29 muestra los resultados del ítem:

Figura 29.
Identificación de la suma correcta de variables.



Nota: El gráfico representa la suma correcta de variables.

Con los datos recabados en el *nivel metacognitivo de la taxonomía de Marzano y Kendall (2007)* se muestra evidencia que el **50%** de los estudiantes encuestados planean, diseñan, construyen, integran, crean, desarrollan y modifican estructuras de control al momento de diseñar algoritmos mediante diagramas de flujo.

La tabla 31 muestra los aprendizajes de los estudiantes encuestados en donde se reflejan los conocimientos relacionados al desarrollo de software a través de la taxonomía de Marzano y Kendall (2007) categorizando el nivel de dominio en la programación computacional.

Tabla 31.

Categorización del conocimiento del grupo experimental a partir de la taxonomía de Marzano y Kendall (2007).

Categoría de Marzano y Kendall: Desarrollo de software	Nivel de dominio
Primer nivel: Recuperación.	86%
Segundo nivel: Comprensión.	75%
Tercer nivel: Análisis.	55%
Cuarto nivel: Aplicación.	77%
Quinto nivel: Metacognitivo.	50%

Nota: La tabla representa el nivel de dominio de cada uno de los sujetos del grupo experimental en cada una de las categorías de la taxonomía de Marzano y Kendall (2007) respecto al desarrollo de software posterior al uso del modelo ALGO-ITS.

Posterior al uso del modelo ALGO-ITS se puede identificar que los niveles de conocimiento del grupo experimental a través de la taxonomía de Marzano y Kendall (2007), se ha incrementado en gran medida al momento de construir algoritmos computacionales en comparación a los resultados de la caracterización del conocimiento.

Capítulo V

Conclusiones

Validación de Cumplimiento de los Objetivos de Investigación

En este último capítulo de la investigación se analizará de forma integral los datos estadísticos derivados de las diferentes variables con la finalidad de responder a las preguntas de investigación y explicar los objetivos marcados en el presente trabajo los cuáles van encauzados en el análisis del impacto de la implementación de un Sistema de Tutoría Inteligente que proporcione apoyo en el aprendizaje de algoritmos computacionales en los estudiantes de la Universidad Politécnica de Victoria.

El análisis consistió en verificar los hallazgos encontrados en los resultados de la investigación a través de la metodología implementada. Para llevar a cabo el estudio, en primer lugar, fue necesario validar los instrumentos de recolección de datos de caracterización del conocimiento y evaluación del conocimiento posterior al uso del modelo ALGO-ITS y, al mismo tiempo, proporcionar información basada en datos científicos y estadísticos de los diferentes estudiantes participantes en el grupo experimental con el objeto de avanzar en el estudio de la implementación de un sistema de tutoría inteligente en la Universidad Politécnica de Victoria.

En el cumplimiento de los objetivos de la investigación, estos se abordaron de manera secuencial y ordenada con la finalidad de analizar la viabilidad de implementar un modelo de ITS.

Con base al objetivo específico de analizar las habilidades de los estudiantes de nuevo ingreso de la UPV relacionadas en el desarrollo de algoritmos computacionales se encontró evidencia del bajo rendimiento del conocimiento de los alumnos del grupo experimental al aplicar el instrumento de caracterización en donde los alumnos demostraron carencias en temas

relacionados a estructuras de control, condiciones y definición de variables. Estos indicadores se demostraron en los resultados de la investigación.

Por otro lado, respecto al objetivo específico de implementar un modelo de ITS que integre elementos tecnológicos y educativos para desarrollar conocimientos y habilidades en el desarrollo de algoritmos computacionales, se diseñó e implementó el modelo ALGO-ITS el cual a través de una red neuronal de aprendizaje profundo se pudo integrar estructuras de retroalimentación para los estudiantes ofreciendo una alternativa de motivación en el momento de resolver los ejercicios. La teoría de la psicología cognitiva abarca los procesos de percepción, representación, memorización y aprendizaje del ser humano por ello el modelo de ITS implementado para soporte en el aprendizaje de algoritmos se diseñó y construyó tomando en consideración estas innovaciones de la teoría integrando los elementos tecnológicos para crear un ambiente de aprendizaje y motivación para los alumnos que hicieron uso del modelo.

Dentro de las características de la teoría de la psicología cognitiva que establece Riviere (1987), se observó que el grupo experimental generó a través del modelo de ITS un ambiente de motivación oportuno para el aprendizaje de algoritmos computacionales en donde los alumnos crearon el sentido de regresión de procesos mentales para la generación de memoria, imaginación y proceso de información para la solución de algoritmos computacionales. Esto se demuestra en los resultados de la investigación ya que el 77% de los estudiantes tienen la habilidad de generar procesos metacognitivos y aplicar soluciones a los problemas algorítmicos.

Por otra parte con base al objetivo específico de evaluar el aprendizaje de los estudiantes respecto a la construcción de algoritmos computacionales posterior al uso del modelo ALGO-ITS, se encontró evidencia en los resultados de la investigación que el rendimiento académico del grupo experimental mejoró considerablemente en relación al pre-test, esto quiere decir que el modelo

ayudó en gran medida a crear un ambiente de motivación y aprendizaje de algoritmos computacionales a través de diagramas de flujo.

Dicho lo anterior, el cumplimiento del objetivo general de la investigación de analizar el impacto de la implementación de Sistema de Tutoría Inteligente (ITS) bajo la teoría de la educación de la psicología cognitiva que proporcione soporte en el proceso de aprendizaje de algoritmos computacionales en estudiantes del primer ciclo de formación de la carrera de Ingeniería en Tecnologías de la Información de la Universidad Politécnica de Victoria para reducir el índice de reprobación y bajo aprendizaje en el momento del desarrollo de software, se ratifica con base a los resultados de la investigación que modelo ALGO-ITS fortaleció el aprendizaje del grupo experimental ofreciendo una alternativa tecnológica inteligente de apoyo para construir algoritmos computacionales para mejorar la eficiencia terminal y bajo rendimiento académico.

El impacto que tienen las innovaciones educativas a través de la tecnología como lo es el modelo ALGO-ITS, ayudan en gran medida a mejorar el aprendizaje de los alumnos que lo utilizan, el modelo a través de un entrenamiento inteligente tiene la capacidad de dar soporte y seguimiento a través de la teoría educativa de la psicología cognitiva para crear un ambiente de motivación y retroalimentación de los avances educativos que estén presentando los alumnos.

Al profundizar en la revisión exhaustiva de la literatura sobre las teorías implementadas previamente en el desarrollo de modelos de Sistemas de Tutoría Inteligente, se tiene evidencia que existe un nicho de oportunidad para implementar tecnología e innovación educativa que aporte en el proceso de enseñanza-aprendizaje ya que existe evidencia de que este tipo de modelos no se ha implementado en el subsistema de Universidades Tecnológicas y Politécnicas de donde depende la Universidad Politécnica de Victoria y con ello el modelo ALGO-ITS aparece como una

oportunidad para mejorar el aprendizaje de algoritmos computacionales y para mejorar la eficiencia terminal y disminuir el índice de deserción.

Respecto a los estilos de aprendizaje encontrados en los resultados de evaluación del conocimiento de los alumnos posterior al uso del modelo de ITS, se evidenció que los estudiantes al mejorar su rendimiento académico implementan la conceptualización de los algoritmos para plantear una solución eficiente aplicando una reflexión de mejora sobre posibles errores que le presentan. Dentro de los estilos de aprendizajes planteados por Kolb (1984) se puede observar que el grupo experimental aplica sus principios ya que hacen uso de la experimentación, conceptualización, abstracción, experimentación y reflexión.

Por otro lado, en las rutas de aprendizaje de algoritmos computacionales planteadas por De la Caridad Saez, et.al., (2015) en donde se indica que para aprender a programar computadoras se debe analizar, diseñar, codificar y depurar, los resultados de la investigación demostraron que el 86% de los alumnos del grupo experimental mejoró las habilidades de análisis, diseño y depuración de algoritmos ya que al solucionar problemas a través de la taxonomía de Marzano y Kendall (2007) en donde se evidencia que el uso de la tecnología como herramienta y medio de aprendizaje es de gran relevancia ya que permite a los estudiantes crear un ecosistema de motivación para el aprendizaje de los temas de estudio.

Finalmente respondiendo a la pregunta de investigación sobre ¿cuál es el impacto de la implementación de un Sistema de Tutoría Inteligente (ITS) de apoyo en el aprendizaje de algoritmos computacionales en los estudiantes de primer cuatrimestre del PE de ITI de UPV? Se puede afirmar que a través de la tecnología innovadora se puede crear un software inteligente que ayude en los procesos educativos, el impacto que tiene en los sistemas de tutoría inteligente es de gran relevancia ya que este tipo de sistemas pueden ayudar y dar soporte de asesoría en diferentes

áreas de la ciencia en donde su aportación en gran medida beneficia el mejorar el rendimiento académico pero sobre todo crear un ambiente de motivación entre los estudiantes que hacen uso de él para aprender y crear conocimiento sobre los temas que se estudien. Específicamente el impacto que tuvo la implementación del modelo de ITS en la Universidad Politécnica de Victoria, benefició en gran medida en mejorar los aprendizajes en los alumnos de la carrera de Ingeniería en Tecnologías de la Información ya que con el uso del software se crearon rutas de aprendizaje para el análisis, diseño, codificación y pruebas de los algoritmos planteados. Por otro lado el índice de reprobación bajó en las materias de programación del primer ciclo de formación dando como resultado que el rendimiento académico de los estudiantes se vaya incrementando.

Discusión de los Conocimientos

En este apartado es necesario analizar los hallazgos de los resultados de la caracterización del conocimiento la cual consiste en analizar de manera clara e individual, en el grupo experimental, los conocimientos previos respecto a el desarrollo de algoritmos. Esta exploración muestra las habilidades que los estudiantes adquirieron en el nivel medio superior. Para este análisis se siguió la taxonomía de Marzano y Kendall (2017) la cual consiste en clasificar el conocimiento en 6 niveles para identificar el nivel de dominio de los estudiantes respecto a los temas de estudio. La taxonomía permitió diseñar un instrumento de recolección de datos para identificar la clasificación de los estudiantes del grupo experimental y verificar el nivel de dominio en el desarrollo de algoritmos.

Haciendo una reflexión de los hallazgos del instrumento de caracterización del conocimiento, se puede identificar que a partir del tercer nivel de la taxonomía de Marzano y

Kendall (2007) el **90%** de los estudiantes muestran bajo conocimiento en el desarrollo de algoritmos computacionales.

Respecto a los hallazgos sobre los conocimientos que mostraron los integrantes del grupo experimental posterior al uso del modelo ALGO-ITS los cuales se obtuvieron a partir de un segundo instrumento validado por un grupo de expertos, se puede concluir que se tiene evidencia que el nivel de conocimiento mejoró en gran medida ya que el **50%** de los estudiantes encuestados demostraron capacidades de planear, diseñar, construir, integrar, desarrollar y modificar estructuras de control al momento de diseñar algoritmos computacionales. La tabla 32 muestra la evolución del conocimiento a partir de la taxonomía de Marzano y Kendall (2017) respecto al desarrollo de algoritmos computacionales y se visualiza el comparativo de los resultados de la caracterización del conocimiento y la evaluación del conocimiento del grupo experimental posterior al uso del modelo ALGO-ITS. Existe mejora de conocimientos del grupo experimental.

Tabla 32.

Evolución del conocimiento respecto al desarrollo de algoritmos.

Categoría de Marzano y Kendall: Desarrollo de software	Resultados pre-test	Resultados post-test
Primer nivel: Recuperación.	85%	86%
Segundo nivel: Comprensión.	62%	75%
Tercer nivel: Análisis.	40.65%	55%
Cuarto nivel: Aplicación.	51.9%	77%
Quinto nivel: Metacognitivo.	46.9%	50%

Nota: La tabla representa el nivel de conocimiento del grupo experimental con base a los resultados del pre-test y post-test.

Posterior al uso del modelo ALGO-ITS se identifica que los niveles de conocimiento del grupo experimental a través de la taxonomía de Marzano y Kendall (2007) se incrementaron en gran medida al momento de construir algoritmos computacionales en comparación a los resultados de la caracterización del conocimiento. Con ello se puede aseverar que el uso de la tecnología en

los procesos educativos ayuda en la mejora del desempeño académico de los estudiantes en cualquier disciplina que se estudie. La implementación de modelos de Sistemas de tutoría inteligente fortalece el campo de la innovación educativa ya que con ella el conocimiento de los estudiantes mejora y por consecuencia la eficiencia terminal se incrementa.

Es importante recalcar que en la tercera fase de la investigación los resultados obtenidos del grupo experimental fueron científicamente comprobados gracias a la taxonomía de Marzano y Kendall (2007) la cual permite categorizar los aprendizajes significativos del grupo experimental, en ellos se refleja que el uso del modelo del ITS ALGO-ITS en la UPV ha beneficiado a los estudiantes de la generación 2020 de la carrera de ITI en el ámbito de algoritmos y programación ya que su rendimiento académico mejoró considerablemente en comparación a las estadísticas de generaciones anteriores presentadas en la definición del problema de esta investigación.

Al hacer un análisis detallado de la hipótesis de investigación la cual afirma que el uso e implementación de los sistemas de tutoría inteligente para el apoyo en el aprendizaje de algoritmos computacionales incrementa los conocimientos en el grupo experimental para mejorar la eficiencia terminal y reducir el índice de deserción, y tomando como referencia los resultados del pre-test y el post-test, se identifica claramente que el uso de la tecnología, a través de la inteligencia artificial, beneficia el proceso educativo en el aula para dar soporte y asesoría a los estudiantes para mejorar los aprendizajes en un área en particular.

Respecto a la variable dependiente que hace hincapié en los aprendizajes significativos para el desarrollo de algoritmos computacionales, ésta se ve en gran medida mejorada a través de la metodología de investigación exploratoria que se aplicó para este trabajo en donde el grupo experimental que utilizó la tecnología del tutor inteligente demostraron mejor rendimiento académico y por consiguiente el índice de reprobación se decremento.

Limitaciones de la Investigación

Tomando en consideración que la investigación se llevó a cabo siguiendo una metodología sistemática, en donde se alcanzaron los objetivos de investigación que se proyectaron al inicio del trabajo que están enfocados en alizar el impacto de los ITS en la educación, se tiene evidencia de resultados sobresalientes respecto a los aprendizajes significativos de los estudiantes que hacen uso de la tecnología como lo son los Sistemas de Tutoría Inteligente para el apoyo en el aprendizaje de algoritmos computacionales, conviene indicar algunas limitaciones que se enlistan a continuación que han de tomarse en cuenta para un trabajo futuro.

- Los resultados de esta investigación deben considerarse como primera fase de un estudio exploratorio ya que como su metodología lo indica se hizo un análisis de un grupo experimental a través de un censo a conveniencia y esto marca la pauta para poder hacer un trabajo futuro analizando el rendimiento académico de los estudiantes de materias de desarrollo de algoritmos en otras carreras.
- Al no tener evidencia de la existencia de un modelo de Sistema de Tutoría Inteligente en Dirección General de Universidades Tecnológicas y Politécnicas de México, hace imposible hacer un comparativo del módulo de conocimiento (donde radica la inteligencia del comportamiento exacto y oportuno de los modelos de ITS) entre algún modelo existente y el modelo desarrollado denominado ALGO-ITS.

Trabajo Futuro

Mientras que existen diferentes investigaciones relacionadas con la implementación de modelos de ITS que versan sobre la adopción de la Inteligencia Artificial para proporcionar soporte en diferentes áreas educativas, es necesario hacer un análisis profundo sobre la construcción del

módulo de conocimiento para examinar los modelos implementados y proponer nuevos diseños innovadores. Por ello, es posible plantear diversas oportunidades para ampliar el estudio de los ITS que permita profundizar en las siguientes líneas de investigación que, al mismo tiempo, permitirán superar algunas de las limitaciones planteadas anteriormente:

- El estudio de investigación se concentró en el grupo experimental conformado por los estudiantes del primer ciclo de formación de la carrera de Ingeniería en Tecnologías de la Información de la Universidad Politécnica de Victoria, se considera pertinente hacer el estudio y análisis para los estudiantes del segundo y tercer grupo de formación a través de una prueba probabilística aplicando un muestreo a conveniencia para obtener información de los estudiantes de nivel avanzado en las materias de programación y poder hacer un comparativo del conocimiento entre los diferentes grupos experimentales.
- Resultaría conveniente implementar el modelo ALGO-ITS para dar soporte en temas avanzados de estructuras de datos algorítmicas para analizar las bases de la programación computacional en los alumnos de niveles altos en las materias de programación y base de datos.
- Finalmente, con la red neuronal diseñada e implementada en el modelo ALGO-ITS, sería prudente implementarla en otro módulo de conocimiento con entrenamiento de temas de otra asignatura de otra carrera de la UPV para analizar los indicadores y resultados derivados del uso del nuevo modelo de ITS.

Glosario

APRENDIZAJE: Adquirir conocimiento de algo por estudio, práctica o experiencia, especialmente el conocimiento necesario para aprender un arte u oficio.

COGNICIÓN: Adquirir conocimiento de algo por estudio, práctica o experiencia, especialmente el conocimiento necesario para aprender un arte u oficio.

COMPETENCIA: La competencia son los conocimientos, habilidades y destrezas que una persona desarrolla para comprender, transformar y aplicar en la práctica el mundo en el que se desenvuelve.

CONSTRUCTIVISMO: Explicación teórica del proceso de aprendizaje a partir de conocimientos previos.

MACHINE LEARNING: El objetivo es crear programas que sean capaces de generalizar comportamientos a partir de la información no estructurada proporcionada como ejemplos.

METACOGNICIÓN: Se refiere a la capacidad de las personas para reflexionar sobre sus procesos de pensamiento y cómo aprenden. A través de la metacognición, los humanos pueden experimentar y regular sus propios procesos mentales básicos que median en su percepción.

CONSTRUCTIVISMO: Teoría explicativa de los procesos de aprendizaje a partir de conocimientos ya adquiridos.

Referencias Bibliográficas

- Abreu Alvarado, Yelena, Barrera Jiménez, Ana Delia, Worosz, Taymí Breijo, y Vichot, Ivón Bonilla. (2018). El proceso de enseñanza-aprendizaje de los Estudios Lingüísticos: su impacto en la motivación hacia el estudio de la lengua. *Mendive. Revista de Educación*, 16(4), 610-623.
- Aleven, V., Sewall, B. M. M. J., & Koedinger, K. R. (2006). The Cognitive Tutor Authoring Tools (CTAT): Preliminary evaluation of efficiency gains. In M. Ikeda, K. D. Ashley & T. W. Chan (Eds.), 8th International Conference on Intelligent Tutoring Systems Lecture Notes in Computer Science (Vol. 4053, pp. 61-70). Berlin Heidelberg: Springer-Verlag. Retrieved from <http://www.cs.cmu.edu/~bmcclaren/pubs/AlevenEtAl-CTAT-ITS2006.pdf>. doi: 10.1007/11774303
- Aleven, V., Stahl, E., Schworm, S., Fischer, F., & Wallace, R. (2003). Help Seeking and Help Design in Interactive Learning Environments. *Review of Educational Research*, 73(3), 277-320. <https://doi.org/10.3102/00346543073003277>
- Álvarez de Zayas, CM. (1993). La escuela en la vida. Ciudad de La Habana: Editorial Pueblo y Educación; 1999. ISSN: 978-959-13-0681-4.
- Álvarez Cisternas, M., y Valdebenito Pinochet, S. (2017). Análisis conceptual de la taxonomía de Marzano y Kendall como marco de referencia para la evaluación de las intervenciones verbales de los estudiantes en clases [Review of *Análisis conceptual de la taxonomía de Marzano y Kendall como marco de referencia para la evaluación de las intervenciones verbales de los estudiantes en clases*]. En G. González-García, I. Silva-Peña, C. Sepúlveda-Parra, y T. Del Valle Contreras (Eds.), *Investigación para la formación de profesores*. (pp. 15-31). Ediciones UCSH. <https://www.researchgate.net/publication/317371151>
- Arevalillo-Herráez, M., Arnau, D., González-Calero, J. A., & Ayesh, A. (2012). Domain specific knowledge representation for an intelligent tutoring system to teach algebraic reasoning. In S. A. Cerri, W. J. Clancey, G. Papadourakis, y K. Panourgia (Eds.), *Intelligent Tutoring Systems (LNCS 7315)* (pp. 630-631). Berlin, Heidelberg: Springer-Verlag. http://dx.doi.org/10.1007/978-3-642-30950-2_95
- Arias S., Francisco J., Jiménez B., Jovani A., y Ovalle C., Demetrio A. (2009). Modelo de planificación instruccional en sistemas tutoriales inteligentes. *Revista Avances en Sistemas e Informática*, 6(1),155-164.
- Askew, S. (2000). *Feedback for Learning*. London: Routledge.
- Barraza Macías, Arturo (2005). Una conceptualización comprehensiva de la innovación educativa. *Innovación Educativa*, 5(28),19-31.

- Barreras Hernández, F. y Ginoris Quesada, O. (2003) Modelo pedagógico para la formación y desarrollo de habilidades, hábitos y capacidades. Ciudad de La Habana, Cuba, Instituto Pedagógico Latinoamericano y CARIBEÑO, IPLAC. Material docente básico de la Maestría en Educación, p.12.
- Berger, P; y Luckman, T. (1968). La construcción social de la realidad. Buenos Aires: Amorrortu
- Bolívar, A. (1996) Aprender en el centro, construir la innovación, en Educación Acción (CEP de Granada), num. 0 (enero-marzo), pp. 33-43.
- Borja, A. M. (2005). El doble estatus de la psicología cognitiva: Como enfoque y como área de investigación. *Revista de Investigación en Psicología*, 8(1).
- Borzone Valdebenito, M. A. (2017). Autoeficacia y vivencias académicas en estudiantes universitarios. *Acta Colombiana de Psicología*, 20(1), 266-274. <https://doi.org/10.14718/ACP.2017.20.1.13>
- Brown, E.; Palinsar, B. (1989) Guided, Cooperative learning and the individual knowledge acquisition. En Resnick B. (comp.) *Knowing, learning, and instruction*. Hillsdale. N.J.
- Brown, J. y Burton, R. 1978 Diagnostic Models for Procedural Bugs in Mathematical Skills, *Cognitive Science*, no. 2, pp. 155-192.
- Buchanan, S. y LaViola Jr., J. (2012). CSTutor: a pen-based tutor for data structure visualization. *Actas del 43rd ACM Technical Symposium on Computer Science Education* (pp. 565-570), Raleigh, NC. doi:10.1145/2157136.2157297
- Burns, H. L., & Capps, C. G. (1988). Foundations of intelligent tutoring systems: an introduction. *Foundations of Intelligent Tutoring Systems*, 1–19. <https://hal.archives-ouvertes.fr/hal-00699852>
- Burton, R. R.; Brown, J. S. (1981). An investigation of computer coaching for informal learning activities. In: Sleeman, D., Brown, J. (eds.): *Intelligent Tutoring Systems*, Capítulo 4, p. 79-98, London: Academic Press.
- Cabada, R., Barrón, M., y Olivares, J. M. (2014). Reconocimiento automático y aspectos éticos de emociones para aplicaciones educativas. *Komputer Sapiens*, 27-33
- Carbonell, J. R. (1970). AI in CAI: An artificial intelligence approach to computer assisted instruction. *IEEE transaction on Man Machine System*. Volumen 11 número 4, p. 190-202.
- Cataldi, Z., & Lage, F. J. (2009). Sistemas tutores inteligentes orientados a la enseñanza para la comprensión. *EduTec Revista Electrónica de Tecnología Educativa*, 28, a108. <https://doi.org/10.21556/edutec.2009.28.456>

- Cataldi, Z., y Lage, F. J. (2010). Modelado del estudiante en sistemas tutores inteligentes. *Revista iberoamericana de tecnología en educación y educación en tecnología*, 5, 29–38. <https://doi.org/10.24215/18509959.0.p.29-38>
- Cho, B. (2000). *Dynamic Planning Models to Support Curriculum Planning and Multiple Tutoring Protocols in Intelligent Tutoring Systems*. [Tesis doctoral]. Illinois Institute of Technology.
- Clancey, W. J. (1991). Intelligent tutoring systems: A tutorial survey, en *App. Artificial Intelligence: A Sourcebook*. McGraw-Hill.
- COMEST (19 de octubre de 2020) World Commission on the Ethics of Scientific Knowledge and Technology (COMEST). <https://en.unesco.org/themes/ethics-science-and-technology/comest>
- Conejo, R., Guzmán, E., Millán, E., Trella, M., Pérez-de-la-Cruz, J.L. y Ríos, A. (2004). SIETTE: a Web-based tool for adaptive testing. *I. J. Artificial Intelligence in Education*. 14. 29-61.
- Cook, T.D. y Campbell, D.T. (1986). The causal assumptions of quasiexperimental practice. *Synthese*, 68, 141-180.
- Crow, T., Luxton-Reilly, A., & Wuensche, B. (2018). Intelligent tutoring systems for programming education: A systematic review. *Proceedings of the 20th Australasian Computing Education Conference on - ACE '18*.
- Cuevas, A. (1999). *Una propuesta didáctica para la enseñanza de las matemáticas, basada en la psicología de Jean Piaget*. México: Departamento de Matemática Educativa, CINVESTAV-IPN.
- De la Caridad Saez Villavicencio, Anisley, Ciudad Ricardo, Febe Ángel, Puentes Puentes, Ursula y Menéndez Pérez, Jorge Sergio. (2015). El desarrollo de la habilidad: implementar algoritmos. Teoría para su operacionalización. *Revista Cubana de Ciencias Informáticas*, 9(3), 99-112.
- Durán Molina, J. C. (2012). *Teorías del aprendizaje y modelos pedagógicos*. Quito, Ecuador: Vinuesa Puente Luis Ramiro - Ediciones Opción
- Durkheim, E. (1974): *Educación y sociología*, Buenos Aires, Shapire.
- Ertmer, P. y Newby, T. (1993). Conductismo, cognitivismo y constructivismo: una comparación de los aspectos críticos desde la perspectiva del diseño de instrucción. *Performance improvement quarterly*, 6(4), 50-72.
- Feldman, R.S. (2005) *Psicología: con aplicaciones en países de habla hispana*. (Sexta edición) México, McGrawHill.

- Fuenmayor, G., y Villasmil, Y. (2008). La percepción, la atención y la memoria como procesos cognitivos utilizados para la comprensión textual. *Revista de Artes y Humanidades UNICA*, 9(22), 187–202. <https://www.redalyc.org/articulo.oa?id=170118859011>
- Galvin, T. (1994). *Mebuilder: An Object-Oriented Lesson Authoring System for Procedural Skills* (9/94). [Tesis doctoral].
- Gallego, M (2011). *Innovación a través de las tecnologías de la información y la comunicación en formación profesional. Estudio de caso*. [Tesis doctoral]. Universidad de Granada, Granada, España.
- Gardner, H. (1995). *Mentes Creativas. Una anatomía de la creatividad humana* (p.53). Barcelona: Paidós.
- Gardner, H. (1998). *Mentes Extraordinarias: Cuatro retratos para descubrir nuestra propia excepcionalidad*. Barcelona: Kairos.
- George, D., & Mallery, P. (2003). *SPSS for Windows step by step: A simple guide and reference*. 11.0 update (4th ed.). Boston: Allyn & Bacon
- Gertner, A. S; Conati, C.; VanLehn, K. (1998). Learning Procedural help in Andes: Generating hints using a Bayesian network student model. *Research & Development*. American Association for Artificial Intelligence.
- Gertner, A.S. & VanLehn, K. (2000). Andes: A Coached Problem Solving Environment for Physics. *Lecture Notes In Computer Science; Vol. 1839 Proceedings of the 5th International Conference on Intelligent Tutoring Systems*. Pages: 133 – 142
- Giraffa, L.; Nunes, M. A. & Viccari, R.M. (1997) MultiEcological: A Learning Environment using Multi-Agent architecture. *MASTA'97: Multi-Agent System: Theory and Applications*. Proceedings. Coimbra: DE-Univ. de Coimbra.
- Gispert Irigoyen, G. D. (2014). *El diseño instruccional en la expresión gráfica arquitectónica*. [Tesis doctoral]. Universidad de los Andes Uniandes, Colombia.
- Gomes-Penna, A. (1984). *Introducao á psicologia cognitiva*. Sao Paulo, EPU.
- González (2004). *Calidad en la Universidad. Evaluación e indicadores*. Salamanca: Ediciones Universidad de Salamanca, 164 pp
- González González, M. T. (2016). El Alumno ante la Escuela y su Propio Aprendizaje: Algunas Líneas de Investigación en Torno al Concepto de Implicación. *REICE. Revista Iberoamericana Sobre Calidad, Eficacia Y Cambio En Educación*, 8(4). Recuperado a partir de <https://revistas.uam.es/reice/article/view/4735>

- Graesser, A. C., Chipman, P., Haynes, B. C., & Olney, A. (2005). AutoTutor: An intelligent tutoring system with mixed-initiative dialogue. *IEEE Transactions on Education*, 48(4), 612–618. <https://doi.org/10.1109/TE.2005.856149>
- Grasha, A. y Hruska-Riechmann, S. (1992), "The Grasha- Riechmann student learning style scales: Research findings and applications", en J. Keefe (ed.). *Student learning styles and brain behavior*. Reston, VA: NASSP.
- Hashimoto, D. A., Rosman, G., Rus, D., y Meireles, O. R. (2018). Artificial Intelligence in Surgery: Promises and Perils. *Annals of surgery*, 268(1), 70–76. <https://doi.org/10.1097/SLA.0000000000002693>
- Heale, R. y Twycross, A. (2015). Validity and reliability in quantitative studies. *Evidence-Based Nursing*, 18(3), 66 – 68.
- Hernández, J., Rengifo, Y., (2015) Los sistemas tutores inteligentes y su aplicabilidad en la educación. *Iberoamericana*, 17, 104-115.
- Hernández Sampieri, R., Fernández Collado, C., & Baptista Lucio, P. (2014). *Metodología de la investigación* (6a. ed.). México D.F.: McGraw-Hill.
- Hersey, P. & Blanchard, K.H. (1988). *Management of organizational behavior: Utilizing human resources*. (5th ed.). Englewood Cliffs, NJ: Prentice-Hall.
- Hilgard, E., y Bower, G. (1980). *Teorías del Aprendizaje*. México: Trillas.
- Huapaya, C. (2009). *Sistemas tutoriales inteligentes. Un análisis crítico*. Facultad de informática. Universidad Nacional de la Plata.
- Hume, G.; Evens, M. (1992) *Student modeling and the classification of errors cardiovascular intelligent tutoring system*. Proceedings of the 4th Midwest Artificial Intelligence and Cognitive Science Society Conference, Utica, IL.
- Johnson, W. L. (1986). Intention-based diagnosis of novice programming errors. Morgan-Kauffman.
- Joyanes Aguilar, L. (2008). *Fundamentos de programación: Algoritmos, estructura de datos y objetos / Luis Joyanes Aguilar* (4a. ed.). Madrid: McGraw-Hill.
- Kerlinger, F. y H. Lee (2002). *Investigación del comportamiento*. México: McGraw Hill/ Interamericana.
- Kezar, A. J. (2001). Understanding and facilitating organizational change in the 21st century. ASHE-ERIC Higher Education Report, 28(4), 147.
- Kim, J. H. (1989). *CIRCSIM-Tutor: An Intelligent Tutoring System for Circulatory Physiology*. [Tesis doctoral]. Illinois Inst. of Tech.

- Kim, J. H. (2000). *Natural Language Analysis and Generation for Tutorial Dialogue*. [Tesis doctoral]. Illinois Institute of Technology.
- Klimenko, Olena (2009). La enseñanza de las estrategias cognitivas y metacognitivas como una vía de apoyo para el aprendizaje autónomo en los niños con déficit de atención sostenida. *Revista Virtual Universidad Católica del Norte*, (27),1-19.
- Kolb, D. (1984), *Experiential learning experiences as the source of learning development*. Nueva York: Prentice Hall.
- Ledesma-Ayora, M. (2014). Análisis de la teoría de Vygotsky para la reconstrucción de la inteligencia social. Cuenca: Universidad Católica de Cuenca
- Leiva, C. (2005). Conductismo, cognitivismo y aprendizaje. *Revista Tecnología En Marcha*, 18(1). Recuperado a partir de https://181.193.125.13/index.php/tec_marcha/article/view/442
- Lind D., Marchal W, Wathen S., 2008, “Estadística aplicada a los negocios y la economía”, Editorial Mc. Graw Hill, China.
- López-Roldán, P. y Fachelli, S. (2015). Metodología de la investigación social cuantitativa. España: Universitat Autònoma de Barcelona.
- Machin Torres Ive, I. T. (2018). Intelligent tutor for programming system using multiple intelligences. *IEEE Latin America Transactions*, 16(2), 634–638. <https://doi.org/10.1109/tla.2018.8327423>
- Mark, M., & Greer, J. (1993). Evaluation methodologies for intelligent tutoring systems. *Journal of Artificial Intelligence in Education*, 4, 129-129. <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.52.6842&rep=rep1 &type=pdf>
- Marzano, R. J. y Kendall, J. S. (2007). *The New Taxonomy of Educational Objectives*. California: ee.uu.: Corwnin Press.
- Mateo, M. et al. (2007). *Una nueva metodología docente para un autoaprendizaje bajo entornos virtuales*. Universitat Rovira i Virgili.
- Mayer, R.E. (1985). *El futuro de la psicología cognitiva*. Madrid, Alianza Editorial.
- Mena, B. (1997). *Didáctica y currículum escolar*. Anthena: Salamanca.
- México con déficit en ingenieros: SEP - Excelsior. Obtenido 04, 2019, de <https://www.excelsior.com.mx/nacional/mexico-con-deficit-en-ingenieros-sep/1292443>.
- Mitrovic, A., Ohlsson, S., Barrow, D. (2013) The effect of positive feedback in a constraint-based intelligent tutoring system. *Computers & Education*, 60(1), pp. 264-272.

- Mitrovic, A., & Ohlsson, S. (1999). Evaluation of a constraintbased tutor for a database language. *International Journal of Artificial Intelligence in Education*, 10, 238–256.
- Moreno, M. (2004). *Historias de innovación educativa. “Un documento conmemorativo”*. México: INNOVA.
- Moreno (2014). La construcción del ser en educación: una mirada desde el constructivismo. *Sophia: colección de Filosofía de la Educación*, 17(2), pp.193-209.
- Nilsson, N. J. (2014). *Principles of Artificial Intelligence*. Morgan Kaufmann.
- Niño, V. (2011). *Metodología de la Investigación: diseño y ejecución*. Bogotá: Ediciones de la U.
- Nunamaker Jr., J. F., Chen, M., & Purdin, T. D. M. (1990). Systems development in information systems research. *Journal of Management Information Systems*, 7(3), 89-106.
- Oulhaci, M. A., Tranvouez, E., Espinasse, B. and Fournier, S. (2013). Intelligent Tutoring Systems and Serious Game for Crisis Management: a Multi- Agents Integration Architecture. *2013 Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises*.
- Ovalle, D., & Jiménez, J. (2006). Ambiente inteligente distribuido de aprendizaje: Integración de ITS y CSCL por medio de agentes pedagógicos. *Revista EIA*(6), 89-104.
- Ovalle, D. A. (2007). *Entorno Integrado de Enseñanza - aprendizaje basado en Sistemas Tutoriales Inteligentes & Ambientes Colaborativos*. Posgrado en Ingeniería de Sistemas. Medellín: Universidad Nacional de Colombia.
- Oviedo, Heidi Celina, y Campo-Arias, Adalberto. (2005). Aproximación al uso del coeficiente alfa de Cronbach. *Revista Colombiana de Psiquiatría*, 34(4), 572-580. Retrieved November 08, 2021, from http://www.scielo.org.co/scielo.php?script=sci_arttext&pid=S0034-74502005000400009&lng=en&tlng=es.
- Parra Sarmiento, S., Gómez Zermeño, M., y Pintor Chávez, M. (2015). Factores que inciden en la implementación de las TIC en los procesos de enseñanza-aprendizaje en 5° de Primaria en Colombia. *Revista Complutense De Educación*, 26, 197-213. https://doi.org/10.5209/rev_RCED.2015.v26.46483
- Pinto, Bismarck. (2003). EVOLUCIÓN DE LOS MODELOS COGNITIVOS. *Ajayu Órgano de Difusión Científica del Departamento de Psicología UCBSA*, 1(2), 29-48. Recuperado a partir de http://www.scielo.org.bo/scielo.php?script=sci_arttext&pid=S2077-21612003000200005&lng=es&tlng=es.
- Quiroga, L. (2016). *Prototipo de Tutor Inteligente para el aprendizaje de la programación de computadores*. [Tesis doctoral]. Universidad Católica de Colombia, Bogotá, Colombia.

- Quiroga, M., y Rodríguez, F. (2002). Estilo Cognitivo Reflexibilidad, Impulsividad diferencias individuales en la Gestión Individual de la relación Velocidad, Exactitud. Recuperado a partir de <https://www.redalyc.org/pdf/3761/376140372005.pdf>
- Quispe, R. (2014) *Tutor inteligente para fortalecer el aprendizaje de la estructura morfosintáctica a nivel primero de primaria*. [Tesis doctoral]. Universidad Mayor de San Andrés, La Paz, Bolivia.
- Ramos Guajardo, A. B., González López, M. J., & González Ruiz, I. (2015). Analysis of the reliability of the fuzzy scale for assessing the students' learning styles in Mathematics. *Proceedings of the 2015 Conference of the International Fuzzy Systems Association and the European Society for Fuzzy Logic and Technology*.
- Regil, V. (23 de enero del 2019). Globalización 4.0. El Economista. Recuperado a partir de <https://www.economista.com.mx/opinion/Globalizacion-4.0-20190123-0034.html>
- Reséndiz, D. (2008). El rompecabezas de la ingeniería, por qué y cómo se transforma el mundo. México: Fondo de Cultura Económica.
- Riviere, A. (1987). El sujeto de la psicología cognitiva. Madrid, Alianza.
- Rodríguez Aguilar, Rosa María; Castillo González, José Luis Miguel; Lira Campos, Alicia Lucrecia. (2013). Diseño de un sistema tutorial inteligente. Apertura, vol. 5, núm. 1. Recuperado a partir de <http://www.udgvirtual.udg.mx/apertura/index.php/apertura/article/view/371>
- Rodríguez Chávez, Mario Humberto. (2021). Sistemas de tutoría inteligente y su aplicación en la educación superior. RIDE. Revista Iberoamericana para la Investigación y el Desarrollo Educativo, 11(22), e015. Epub 21 de mayo de 2021. <https://doi.org/10.23913/ride.v11i22.848>
- Rowe N. C. and T. Galvin, (1998) An authoring system for intelligent tutors for procedural skills. IEEE Intelligent Systems, 13, 3 May/June, 61-69.
- Saavedra, A. (2012). Dry to dynamic civic education curricula. In D. Campbell, F. Hess, & M. Levinson (Eds.), Making civics count. Cambridge, MA: Harvard Education Press.
- Saarikoski, L.; Salojärvi, S.; Del Corso, D.; Ovcin, E. (2001): "The 3DE: An Environment for the Development of Learner-Oriented Customised Educational Packages". ITHET, 4-6 Julio, 2001, Kumamoto. (www.eecs.kumamoto-u.ac.jp/ITHET01/proceedings.htm)
- Salgueiro, F. A, Costa, G., Cataldi, Z., García Martínez, R. y Lage, F. J. (2005). Sistemas inteligentes para el modelado del tutor. Proc. in GCETE'2005, marzo 13-15.
- Sánchez-Cardona, I., y Rodríguez-Arocho, W. C. (2011). Valoración crítica a la teoría de aprendizaje situado y del concepto de comunidades de práctica desde el enfoque histórico-cultural. *Pedagogía*, 44(1), 113-132. Recuperado a partir de <https://revistas.upr.edu/index.php/educacion/article/view/16563>

- Sancho, L. (2002). *Sistemas Tutores Inteligentes: Una alternativa para el uso de computadoras en educación*. Education Net. Red Global de educación a distancia. (DistEdNet) Universidad Estatal a Distancia.
- Sawyer, R. K. (2006). Introduction: The New Science of Learning. In R. K. Sawyer (Ed.), *The Cambridge handbook of: The learning sciences* (pp. 1–16). Cambridge University Press.
- Shah, F. (1997). *Recognizing and Responding to Student Plans in an Intelligent Tutoring System: Circsim-Tutor*. [Tesis doctoral]. Illinois Ins. of Tech.
- Sharma, S., Ghorpade, S., Sahni, A. y Saluja, N. (2014). Survey of Intelligent Tutoring Systems: a review on the development of expert/intelligent tutoring systems, various teaching strategies and expert tutoring system design suggestions. *International Journal of Engineering Research & Technology*, 3(11), 37-42. Recuperado a partir de <https://www.ijert.org/browse/volume-3-2014/november-2014-edition>.
- Sistema Tutores Inteligentes*. (s/f). Ecured.cu. Recuperado a partir de https://www.ecured.cu/Sistema_Tutores_Inteligentes
- Steneck, N. H. (2007). *Ori introduction to the responsible conduct of research, 2004 (revised)*. Claitor's Law Books and Publishing Division.
- Schmeck, R. R. Geisler-Brenstein, E. y Cercy, S. P. (1991). Self-concept and learning: The Revised Inventory of Learning Processes. *Educational Psychology*, 11, 343-362.
- Self, J. (1999). The defining characteristics of intelligent tutoring systems research: ITS care, precisely. *Journal of Artificial Intelligence in Education*, vol. 10, núm. 3-4, pp. 350-364.
- Stevens, A.; Collins, A. (1977). The goal structure of a Socratic tutor. *Procs of the National ACM Conference*. New York: ACM.
- Salgueiro, F., Costa, G., Cataldi, Z., Lage, F., García-Martínez, R. (2005). Nuevo enfoque metodológico para el diseño de de los sistemas tutores inteligentes a partir de un acercamiento distribuido. *Revista de Informática Educativa y Medios Audiovisuales*, 2, 25-32. Recuperado a partir de: <http://laboratorios.fi.uba.ar/lie/Revista/Articulos/020205/A3ago2005.pdf>
- Sánchez, I.I., Cabrera, J.M. y Martínez, J. E. (2015). Ayudas virtuales como apoyo al aprendizaje inclusivo en la ingeniería. *Revista Horizontes Pedagógicos* 17(2), 104-116.
- Suárez Granados, J. J., Arencibia Rodríguez del Rey, Y., & Pérez Fernández, A. C. (2016). Metodología para desarrollar un sistema tutor inteligente basado en la Web, para estudiantes de ingeniería. *Universidad y Sociedad [seriada en línea]*, 8 (4). pp. 108-115. Recuperado a partir de <http://rus.ucf.edu.cu/>

- Taherdoost, H. (2016). Sampling Methods in Research Methodology; How to Choose a Sampling Technique for Research. *International Journal of Academic Research in Management (IJARM)*, 5, 18-27. <https://doi.org/10.2139/ssrn.3205035>
- Tarongí, V. A. (2010). *Sistema Tutor Inteligente Adaptativo para Laboratorios virtuales y remotos*. [Tesis Maestría]. Universitario en Automática e Informática Industrial, Valencia: Universidad Politécnica de Valencia.
- Telukdariea, A., Buhulaiga, E., Bag, S., Gupta, S. & Luo, Z. (2018). Industry 4.0 implementation for multinationals. *ScienceDirect*, 118, 316-329.
- TensorFlow*. (s/f). TensorFlow. Recuperado el 17 de enero de 2021, de <https://www.tensorflow.org/>
- Toribio, L. (2019, enero 24). *México con déficit en ingenieros: SEP*. Excélsior. <https://www.excelsior.com.mx/nacional/mexico-con-deficit-en-ingenieros-sep/1292443>
- Torres, C. A. B., y Palma, O. F. (2016). *Metodología de la investigación : administración, economía, humanidades y ciencias sociales* (4a ed.). Pearson Educación.
- Tünnermann Bernheim, Carlos (2011). El constructivismo y el aprendizaje de los estudiantes. *Universidades*, (48), 21-32. ISSN: 0041-8935. Disponible en: <https://www.redalyc.org/articulo.oa?id=37319199005>
- Ulate (2013). Conductismo vs. Constructivismo: Sus principales aportes en la pedagogía, Diseño Curricular e Instruccional en el Área de las Ciencias Naturales. *Ensayos Pedagógicos*. (pp. 67-83).
- UNESCO. (2019). *Steering AI and Advanced ICTs for Knowledge Societies*. UNESCO Publishing, 1, 201.
- Urretavizcaya, M. (2001). Sistemas inteligentes en el ámbito de la educación. *Revista Iberoamericana de Inteligencia Artificial*. N° 12, pp. 5-12. ISSN 1137-3601
- VanLehn, K (1988). *Student Modelling*. M. Polson. *Foundations of Intelligent Tutoring systems*. Hillsdale. N.J. Lawrence Erlbaum Associates, 55-78.
- Vázquez Lizárraga, Rosa Isela. (2012). ¿Qué ingenieros necesita México?. *Innovación educativa* (México, DF), 12(60), 125-135. http://www.scielo.org.mx/scielo.php?script=sci_arttext&pid=S1665-26732012000300009&lng=es&tlng=es.
- Viger, P., Nkambou, R., Nguifo, E., Mayers, A., Faghihi, U. (2013, 10). A multiparadigm intelligent tutoring system for robotic arm training. *IEEE*, 6, 364-377.
- Watson, J. B. (1924). *Psychology from the standpoint of a behaviorist*. Philadelphia: Lippincott.

- Watzlawick, P. (Ed.). (1980). *The invented reality: How do we know what we believe we know?* WW Norton.
- Weragama, D. S. (2013). *Intelligent tutoring system for learning PHP*. [Tesis maestría]. Queensland University of Technology. Recuperado a partir de <https://eprints.qut.edu.au/63202/>
- Welch y Comer (1988). Coeficiente de Alpha de Cronbach. Recuperado de <http://www.uv.es/~friasnav/AlfaCronbach.pdf>.
- Wenger, E. (2002). Legitimate peripheral participation in communities of practice. In R. Harrison (Ed.), *Supporting lifelong learning: Volume I: Perspectives on learning* (pp. 111-126). London: Routledge Falmer.
- Wolf, B. (1984). *Context Dependent Planning in a Machine Tutor*. [Tesis doctoral]. University of Massachusetts, Amherst, Massachusetts.
- Yubero Jiménez, S. (2004). Socialización y aprendizaje social. En *Psicología social, cultura y educación/ coord. por Darío Páez Rovira, Itziar Fernández Sedano, Silvia Ubillos Landa, Elena Mercedes Zubieta*, pp. 819-844. Madrid: Pearson.
- Zapata-Ros, Miguel (2015). Teorías y modelos sobre el aprendizaje en entornos conectados y ubicuos. Bases para un nuevo modelo teórico a partir de una visión crítica del “conectivismo”. *Education in the Knowledge Society*, 16(1),69-102.
- Zatarain, R. (2018). Reconocimiento afectivo y gamificación aplicados al aprendizaje de Lógica algorítmica y programación. *Revista Electrónica de Investigación Educativa*, 20(3), 115-125. <https://doi.org/10.24320/redie.2018.20.3.1636>

Anexos

Anexo 1: Instrumento de recolección de datos

Instrumento de recolección de datos utilizado en la fase uno de la metodología de investigación: Caracterización de los aprendizajes relacionados al desarrollo de algoritmos.



Cuestionario: Caracterización de habilidades en el desarrollo de algoritmos computacionales.

Estimado alumno:

Con el propósito de desarrollar un proyecto de investigación sobre el uso de los Sistemas de Tutoría Inteligente para el aprendizaje de algoritmos computacionales, te solicitamos de la manera más atenta contestar de forma objetiva el siguiente cuestionario:

I. DATOS GENERALES DEL ALUMNO

Description (optional)

1. Edad: *

Short answer text
.....

2. Estado civil *

Soltero

Casado

Otro

3. Sexo *

Masculino

Femenino

4. Bachillerato: *

COBAT

ITACE

Preparatoria

CBTis

CETIS

Otro

5. Especialidad: *

Fisico - matematico

Programación

Soporte técnico

Otro

3. Sexo *

Masculino

Femenino

4. Bachillerato: *

COBAT

ITACE

Preparatoria

CBTis

CETIS

Otro

5. Especialidad: *

Fisico - matematico

Programación

Soporte técnico

Otro

II. CONOCIMIENTOS SOBRE DESARROLLO DE SOFTWARE

Description (optional)

6. ¿Tienes experiencia desarrollando algoritmos? *

Sí

No

7. Si tu respuesta fue Sí en la pregunta 6, ¿Qué nivel de experiencia?

Alto

Alto - Medio

Medio

Bajo

Ninguno

8. Si tu respuesta fue Sí en la pregunta 6, ¿Haz desarrollado algoritmos con diagramas de flujo?

Sí

No

9. Si tu respuesta fue Si en la pregunta 6, ¿Haz desarrollado algoritmos con pseudo código?

Si

No

10. ¿Dominas algún lenguaje de programación? *

Si

No

11. Si tu respuesta fue Si a la pregunta 10, ¿Qué lenguaje de programación?

Lenguaje C

C++

Visual Basic

Java

Python

PHP

Otro

III. MATERIALES DIDÁCTICOS

Description (optional)

12. ¿Los recursos didácticos y tecnológicos empleados en clase influenciaron tu motivación e interés por la misma? *

Sí

No

13. Al inicio de cada materia, ¿se realizó un diagnóstico de tus conocimientos previos sobre cada una de ellas? *

Sí

No

14. En la materia de algoritmos, ¿realizaste prácticas y ejercicios graduales en complejidad hasta llegar a un proyecto o un producto? *

Sí

No

15. ¿Las calificaciones que obtuviste son el reflejo de tus conocimientos, habilidades, destrezas y actitudes? *

Sí

No

IV. ESTUDIOS, CONOCIMIENTOS Y HABILIDADES

16. ¿Cuáles de las siguientes competencias dominas?

a). Aplicar metodologías para implementar soluciones eficientes. *

Si

No

b). Desarrollar soluciones a problemas de programación para diseñar soluciones eficientes. *

Si

No

c). Definir y diseñar sistemas de información de acuerdo a los requerimientos solicitados. *

Si

No

V. TUTORÍAS Y ASESORÍAS

17. ¿Cuáles de las siguientes competencias dominas?

18. ¿Conoces el motivo de la asignación del tutor *

Si

No

19. ¿Has interactuado con tu tutor? *

Si

No

20. ¿Has tenido la necesidad de un asesor *

Si

No

21. Si tu respuesta fue sí en la pregunta 15, ¿En qué área relacionadas al software?

Algoritmos

Programación

Metodologías de desarrollo

Otro

22. ¿Las asesorías te ayudaron a solucionar el problema de aprendizaje? *

Si

No

Por su colaboración, muchas gracias.

Description (optional)

Anexo 2: Matriz de consistencia

Título: Sistema de Tutoría Inteligente como herramienta de apoyo en el aprendizaje de algoritmos computacionales a nivel licenciatura.							
Autor: Mario Humberto Rodríguez Chávez							
Planteamiento del problema	Objetivos	Hipótesis	Variables	Sujetos de investigación	Diseño de investigación	Instrumentos de recolección	Validación
<p>Pregunta general.</p> <p>¿Cuál es el impacto de la implementación de un Sistema de Tutoría Inteligente (ITS) de apoyo en el aprendizaje de algoritmos computacionales en los estudiantes de primer cuatrimestre del PE de ITI de UPV?</p>	<p>Objetivo general.</p> <p>Analizar el impacto de la implementación de Sistema de Tutoría Inteligente (ITS) bajo la <i>teoría de la educación de la psicología cognitiva</i> que proporcione soporte en el proceso de aprendizaje de algoritmos computacionales en estudiantes del primer ciclo de formación de la carrera de Ingeniería en Tecnologías de la Información de la Universidad Politécnica de Victoria, reduciendo el índice de reprobación y bajo aprendizaje en el momento del desarrollo de software.</p>	<p>La implementación de un ITS en la ITI de la UPV de apoyo en el aprendizaje de algoritmos computacionales mejora los conocimientos, reduce el índice de reprobación y maximiza la eficiencia terminal académica.</p>	<p>Variable independiente.</p> <p>Sistema de tutoría inteligente (ITS) de soporte y apoyo en el proceso de aprendizaje de algoritmos computacionales.</p> <p>Conceptualización.</p> <p>Software computacional de apoyo y soporte para la construcción de algoritmos computacionales.</p> <p>Operación.</p> <p>Medir el Grado de aprendizaje y aprovechamiento en la construcción de algoritmos computacionales.</p>	<p>Población:</p> <p>Alumnos generación 2020 programa educativo de ITI.</p> <p>Censo del 100% de los 94 alumnos generación 2020.</p>	<p>Metodología:</p> <p>Cuantitativa.</p> <p>Método:</p> <p>Descriptivo</p> <p>Diseño:</p> <p>Cuasiexperimental exploratorio</p>	<p>Encuesta cuestionario</p> <p>Pre-test de caracterización.</p> <p>Post-test evaluación de conocimientos al uso del ITS.</p>	<p>Alfa de Cronbach</p>
<p>Preguntas específicas.</p> <p>¿Cuáles son las habilidades de los alumnos del primer cuatrimestre en desarrollo de algoritmos computacionales?</p> <p>¿Cuál es el nivel de conocimiento de los estudiantes que hacen uso de un ITS respecto a la construcción de algoritmos?</p>	<p>Objetivos específicos.</p> <p>Analizar las habilidades de los estudiantes de nuevo ingreso de la UPV relacionadas en el desarrollo de algoritmos computacionales.</p> <p>Implementar un modelo de ITS que integre elementos tecnológicos y educativos para desarrollar conocimientos y habilidades en el desarrollo de algoritmos computacionales, para reducir el índice de reprobación y bajo aprendizaje de algoritmos computacionales.</p> <p>Evaluar el aprendizaje de los estudiantes respecto a la construcción de algoritmos computacionales a través de diagramas de flujo.</p>		<p>Variable dependiente.</p> <p>Aprendizaje de algoritmos computacionales.</p> <p>Conceptualización.</p> <p>Proceso en el cual los estudiantes adquieren o modifican habilidades, destrezas y conocimientos relacionados en la construcción de algoritmos a través de diagramas de flujo.</p> <p>Operación.</p> <p>Evaluación de la manera de implementar soluciones algorítmicas a problemas propuestos a los estudiantes. Caracterización previa de conocimientos (pre – test) y verificación de los logros obtenidos (post – test)</p>				

Anexo 3: Base de datos software SPSS

Instrumento 1: Pre-test caracterización de conocimiento.

Edad	Correo	Cuatrimestre	Estado civil	Sexo	Bachillerato	Especialidad	Experiencia desarrollo	Nivel experiencia	Desarrollo diagramas	Nivel desarrollo diagramas	Domina lenguaje	Lenguajes programación	N1 Recuperación_P1	N1 Recuperación_P2
2	18 2030252@upv.edu.mx	Primero	Soltero	Masculino	Otro	Otro	0	0	0	0	0	0	1	0
3	18 2030406@upv.edu.mx	Primero	Soltero	Masculino	CBTIS	Programación	1	0	1	1	0	0	1	0
4	19 1930161@upv.edu.mx	Quinto	Soltero	Femenino	CBTIS	Soporte técnico	1	0	1	1	1	Lenguaje C	0	0
5	20 1930151@upv.edu.mx	Cuarto	Soltero	Masculino	Preparatoria	Soporte técnico	0	0	0	0	0	1 Python	0	0
6	19 1930118@upv.edu.mx	Cuarto	Soltero	Masculino	CBTIS	Programación	1	0	0	1	0	0	0	0
7	19 1930611@upv.edu.mx	Quinto	Soltero	Femenino	Preparatoria	Otro	0	0	0	0	0	0	0	0
8	20 1930672@upv.edu.mx	Cuarto	Soltero	Masculino	Preparatoria	Otro	1	0	1	1	0	0	1	0
9	18 2030342@upv.edu.mx	Primero	Soltero	Masculino	CBTIS	Programación	1	1	0	1	0	Lenguaje C	1	0
10	19 1930147@upv.edu.mx	Quinto	Soltero	Masculino	CBTIS	Programación	1	1	1	1	0	0	1	0
11	20 1930151@upv.edu.mx	Sexto	Soltero	Masculino	CBTIS	Físico - matemático	1	1	1	1	1	1 Java	1	0
12	18 2030270@upv.edu.mx	Primero	Soltero	Femenino	CBTIS	Programación	1	1	1	1	1	1 Lenguaje C	0	0
13	21 1830307@upv.edu.mx	Quinto	Soltero	Masculino	CBTIS	Físico - matemático	1	0	0	1	0	0	1	1
14	18 2031043@upv.edu.mx	Primero	Soltero	Masculino	COBAT	Otro	1	0	1	1	1	0 Lenguaje C	1	0
15	21 1930692@upv.edu.mx	Cuarto	Soltero	Masculino	CBTIS	Programación	1	1	1	1	1	0	1	0
16	20 2030300@upv.edu.mx	Primero	Soltero	Masculino	COBAT	Otro	1	1	1	1	1	0	1	0
17	20 1930318@upv.edu.mx	Quinto	Soltero	Masculino	Preparatoria	Otro	1	1	1	1	1	1 Lenguaje C	1	0
18	18 2030429@upv.edu.mx	Primero	Otro	Masculino	CBTIS	Soporte técnico	0	0	1	1	1	0 Otro	1	0
19	20 1830069@upv.edu.mx	Séptimo	Soltero	Masculino	CBTIS	Soporte técnico	0	0	0	0	0	0	1	0
20	1930050@upv.edu.mx	Primero	Soltero	Masculino	CBTIS	Otro	1	0	1	1	0	0	1	0
21	18 2030044@upv.edu.mx	Primero	Soltero	Masculino	CBTIS	Programación	1	0	1	1	0	0	1	0
22	20 1930124@upv.edu.mx	Quinto	Soltero	Masculino	CBTIS	Programación	1	1	1	1	1	0	1	0
23	19 1930185@upv.edu.mx	Quinto	Soltero	Masculino	CBTIS	Programación	1	1	1	1	1	0	1	0
24	21 1730355@upv.edu.mx	Noveno	Otro	Masculino	CBTIS	Otro	1	1	1	1	0	1 Python	1	0
25	21 1730355@upv.edu.mx	Noveno	Otro	Masculino	CBTIS	Otro	1	1	1	1	0	1 Python	1	0
26	21 1730355@upv.edu.mx	Noveno	Otro	Masculino	CBTIS	Otro	1	1	1	1	0	1 Python	1	0
27	21 1730355@upv.edu.mx	Noveno	Otro	Masculino	CBTIS	Otro	1	1	1	1	0	1 Python	1	0
28	21 1830261@upv.edu.mx	Séptimo	Soltero	Masculino	CBTIS	Soporte técnico	1	1	0	1	1	1 C++	1	0
29	19 1930064@upv.edu.mx	Quinto	Soltero	Femenino	CBTIS	Programación	1	1	1	1	1	1 Java	1	0
30	18 2030192@upv.edu.mx	Primero	Soltero	Masculino	CBTIS	Programación	1	1	1	1	1	1 Otro	1	0
31	18 2030396@upv.edu.mx	Primero	Soltero	Masculino	CBTIS	Programación	1	1	1	1	1	1 Java	1	0
32	22 1830502@upv.edu.mx	Cuarto	Soltero	Masculino	Preparatoria	Otro	1	1	1	1	1	1 Java	1	0
33	18 2030418@upv.edu.mx	Primero	Soltero	Femenino	Preparatoria	Programación	1	1	0	1	1	1 Lenguaje C	0	0

Instrumento 1: Resultados pre-test caracterización de conocimiento.

Advertencias
El determinante de la matriz de covarianzas es cero o aproximadamente cero. Las estadísticas basadas en su matriz inversa no se pueden calcular y se visualizan como valores perdidos por el sistema.

Escala: Instrumento 1

Resumen de procesamiento de casos

Casos	Válido	N	%
	Válido	94	100,0
	Excluido ^a	0	,0
	Total	94	100,0

a. La eliminación por lista se basa en todas las variables del procedimiento.

Estadísticas de fiabilidad

Alfa de Cronbach	N de elementos
,802	,791 30

SPSS Statistics Archivo Editar Ver Datos Transformar Insertar Formato Analizar Gráficos Utilidades Ampliaciones Ventana Ayuda

Resultado77.sps [Documento96] - IBM SPSS Statistics Visor

Resultado

Estadísticas de elemento

	Media	Desviación estándar	N
Experiencia_desarrollo	.90	.296	94
Nivel_experiencia	.66	.476	94
Desarrollo_diagramas	.82	.387	94
Nivel_des_diagramas	.87	.335	94
Dominaas_lenguaje	.52	.502	94
N1_Recuperacion_P1	.88	.323	94
N1_Recuperacion_P2i	.44	.499	94
N1_Recuperacion_P3i	.65	.480	94
N2_Compresion_P1i	.80	.404	94
N2_Compresion_P2i	.81	.396	94
N2_Compresion_P3	.77	.426	94
N3_Analisis_P1	.26	.438	94
N3_Analisis_P2i	.34	.476	94
N3_Analisis_P3	.60	.493	94
N4_Aplicacion_P1i	.61	.491	94
N4_Aplicacion_P2	.34	.476	94
N4_Aplicacion_P3	.94	.246	94
N5_Metacognitivo_P1	.53	.502	94
N5_Metacognitivo_P2	.06	.246	94
N5_Metacognitivo_P3	.48	.502	94
Recursos_motivacion	.84	.368	94
Diagnostico_inicio_materia	.76	.432	94
Algo_gradual_complejo	.90	.296	94
Calificaciones_destrezas	.84	.368	94
Competencias_metodologica	.72	.450	94
Competencias_soluciones_programacion	.74	.438	94
Competencias_Sistemas_inf	.74	.438	94
Conoce_motivo_tutor	.76	.432	94
Interaccion_tutor	.86	.347	94
Ayudaron_asesorias	.70	.460	94

SPSS Statistics Archivo Editar Ver Datos Transformar Insertar Formato Analizar Gráficos Utilidades Ampliaciones Ventana Ayuda

Resultado77.sps [Documento96] - IBM SPSS Statistics Visor

Matriz de correlaciones entre elementos

	Experiencia_desarrollo	Nivel_experiencia	Desarrollo_diagramas	Nivel_des_diagramas	Dominaas_lenguaje	N1_Recuperacion_P1	N1_Recuperacion_P2i	N1_Recuperacion_P3i	N2_Compresion_P1i	N2_Compresion_P2i	N2_Compresion_P3	N3_Analisis_P1	N3_Analisis_P2i	N3_Analisis_P3	N4_Aplicacion_P1i
Experiencia_desarrollo	1.000	.453	.505	.634	.195	.106	.140	.215	.286	.301	.162	-.141	.157	.248	.182
Nivel_experiencia	.453	1.000	.362	.263	.480	.018	.134	.130	.253	.278	.292	.060	.279	.094	.110
Desarrollo_diagramas	.505	.362	1.000	.400	.214	.173	.135	.233	.245	.263	.197	-.105	.221	.176	.131
Nivel_des_diagramas	.634	.263	.400	1.000	.080	.059	.208	.320	.125	.138	.316	-.068	.140	.075	.279
Dominaas_lenguaje	.195	.480	.214	.080	1.000	-.150	.113	.098	.207	.237	.024	.170	.239	.035	.187
N1_Recuperacion_P1	.106	.018	.173	.059	-.150	1.000	-.120	.148	.229	.243	.190	-.166	.052	.374	.181
N1_Recuperacion_P2i	.140	.134	.135	.208	.113	.120	1.000	.647	.389	.428	.182	.518	.817	.200	.709
N1_Recuperacion_P3i	.215	.130	.233	.320	.098	.148	.647	1.000	.629	.662	.225	.328	.528	.212	.867
N2_Compresion_P1i	.286	.253	.245	.125	.207	.229	.389	.629	1.000	.967	.160	.234	.306	.233	.516
N2_Compresion_P2i	.301	.278	.263	.138	.237	.243	.428	.662	.967	1.000	.178	.223	.350	.260	.549
N2_Compresion_P3	.162	.292	.197	.316	.024	.190	.182	.225	.160	.178	1.000	-.022	.185	.108	.172
N3_Analisis_P1	-.141	.060	-.105	-.068	.170	-.166	.518	.328	.234	.223	-.022	1.000	.455	-.015	.372
N3_Analisis_P2i	.157	.279	.221	.140	.239	.052	.817	.528	.306	.350	.185	.455	1.000	.180	.579
N3_Analisis_P3	.248	.094	.176	.075	.035	.374	.200	.212	.233	.260	.108	-.015	.180	1.000	.268
N4_Aplicacion_P1i	.182	.110	.131	.279	.187	.181	.709	.867	.516	.549	.172	.372	.579	.268	1.000
N4_Aplicacion_P2	.005	-.005	-.129	-.196	-.120	-.018	.228	.011	-.030	.007	-.292	.146	.195	.089	.119
N4_Aplicacion_P3	-.085	-.096	-.123	.422	-.163	-.095	.230	.264	-.023	-.016	.267	.153	.188	-.126	.235
N5_Metacognitivo_P1	.202	.136	.335	.344	.125	.123	.180	.293	.165	.194	.186	.207	.179	.053	.161
N5_Metacognitivo_P2	-.063	.004	.010	-.031	.163	-.176	.121	.101	.023	.016	.144	.047	.180	-.228	.121
N5_Metacognitivo_P3	.022	-.031	-.048	-.208	.066	.150	.274	.169	.223	.358	-.024	.171	.255	.182	.162
Recursos_motivacion	-.142	.055	-.205	-.167	-.127	-.068	.032	.106	.142	.157	-.035	.122	.068	-.063	.065
Diagnostico_inicio_materia	-.101	-.148	-.075	-.144	-.199	.255	.251	.204	.268	.226	-.198	.106	.148	-.015	.200
Algo_gradual_complejo	-.106	.071	.223	-.124	.122	-.006	.213	.291	.286	.301	-.094	.191	.234	.027	.256
Calificaciones_destrezas	.056	-.068	-.054	-.080	-.243	.022	.022	.106	.142	.083	-.035	-.011	.007	-.122	.006
Competencias_metodologica	-.040	.309	.204	-.165	.122	-.077	.064	.093	.222	.243	.108	.035	.143	-.122	.037
Competencias_soluciones_programacion	.058	.197	.105	-.078	.123	.015	.171	.285	.374	.397	.137	.119	.215	.114	.177
Competencias_Sistemas_inf	.058	.094	-.022	-.078	.123	-.061	.171	.387	.313	.335	-.151	.175	.163	.065	.327
Conoce_motivo_tutor	-.101	-.096	-.010	.153	-.050	-.130	.401	.307	.206	.226	.095	.277	.357	-.217	.251
Interaccion_tutor	-.026	-.158	-.028	.308	-.199	-.146	.104	.286	.029	.040	.142	.093	.093	-.079	.182
Ayudaron_asesorias	-.054	-.124	-.004	-.110	.074	-.020	.338	.155	.193	.156	-.250	.168	.272	.032	.285

SPSS Statistics Archivo Editar Ver Datos Transformar Insertar Formato Analizar Gráficos Utilidades Ampliaciones Ventana Ayuda

Resultado77.spv [Documento9] - IBM SPSS Statistics Visor

	N3_Analisis_P3	N4_Aplicacion_P1	N4_Aplicacion_P2	N4_Aplicacion_P3	N5_Metacognitivo_P1	N5_Metacognitivo_P2	N5_Metacognitivo_P3	Recursos_motivacion	Diagnostico_inicio_materia	Algo_gradual_complejo	Calificaciones_destrezas	Competencias_metodologia	Competencias_soluciones_programacion	Competencias_Sistemas_inf	Conoces_motivo_tutor	Interaccion_tutor	Ayudaron_asesorias
.248	.182	-.005	-.085	.202	-.063	.022	-.142	-.101	-.106	.056	-.040	.058	.058	-.101	-.026	-.054	
.094	.110	-.005	-.096	.136	.004	-.031	.055	-.148	.071	-.068	.309	.197	.094	-.096	-.158	-.124	
.176	.131	-.129	-.123	.335	.010	-.048	-.205	-.075	.223	-.054	.204	.105	-.022	-.010	-.028	-.004	
.075	.279	-.196	.422	.344	-.031	-.208	-.167	-.144	-.124	-.080	-.165	-.078	-.078	.153	.308	-.110	
.035	.187	-.120	-.163	.125	.163	.066	-.127	-.199	.122	-.243	.122	.123	.123	-.050	-.199	.074	
.374	.181	-.018	-.095	.123	-.176	.150	-.068	.255	-.006	.022	-.077	.015	-.061	-.130	-.146	-.020	
.200	.709	.228	.230	.180	.121	.274	.032	.251	.213	.032	.064	.171	.171	.401	.104	.338	
.212	.867	.011	.264	.293	.101	.169	.106	.204	.291	.106	.093	.285	.387	.307	.286	.155	
.233	.516	-.030	-.023	.165	.023	.323	.142	.268	.286	.142	.222	.374	.313	.306	.029	.193	
.260	.549	.007	-.016	.194	.016	.358	.157	.226	.301	.083	.243	.397	.335	.226	.040	.156	
.108	.172	-.292	.267	.186	.144	-.024	-.035	-.198	-.094	-.035	.108	.137	-.151	.095	.142	-.250	
-.015	.372	.146	.153	.207	.047	.171	.122	.106	.191	-.011	.035	.119	.175	.277	.093	.168	
.180	.579	.195	.188	.179	.180	.255	.068	.148	.234	.007	.143	.215	.163	.357	.093	.272	
1.000	.268	.089	-.126	.053	-.228	.182	-.063	-.015	.027	-.122	-.122	.114	.065	-.217	-.079	.032	
.268	1.000	.119	.235	.161	.121	.162	.065	.200	.256	.006	.037	.177	.327	.251	.182	.285	
.089	.119	1.000	-.180	-.136	-.004	.165	.068	.043	.081	.068	-.208	-.043	.009	-.113	-.102	.124	
-.126	.235	-.180	1.000	.191	-.110	-.098	.005	-.149	.063	-.114	-.064	-.053	-.153	.256	.400	-.075	
.053	.161	-.136	.191	1.000	.071	.131	-.059	-.038	.057	-.059	-.103	-.109	-.060	.061	.057	-.052	
-.228	.121	-.004	-.110	.071	1.000	.098	-.005	-.155	.085	-.005	-.033	-.047	.053	.149	-.021	.075	
.182	.162	.165	-.098	.131	.098	1.000	.069	-.049	.167	.011	.212	.170	.170	.001	-.110	-.074	
-.063	.065	.068	.005	-.059	-.005	.069	1.000	.022	.056	.127	.120	.344	.278	.157	.078	.034	
-.015	.200	.043	-.149	-.038	-.155	-.049	.022	1.000	.151	.225	-.020	-.050	.121	.194	-.156	.387	
.027	.256	.081	.063	.057	.085	.167	.056	.151	1.000	-.043	.122	.058	.224	.235	-.130	.183	
-.122	.006	.068	-.114	-.059	-.005	.011	.127	.225	-.043	1.000	-.010	-.055	.011	.090	.162	.097	
-.122	.037	-.208	-.064	-.103	-.033	.212	.120	-.020	.122	-.010	1.000	.292	.183	.146	-.041	-.091	
.114	.177	-.043	-.053	-.109	-.047	.170	.344	-.050	.058	-.055	.292	1.000	.385	-.050	.119	-.168	
.065	.327	.009	-.153	-.060	.053	.170	.278	.121	.224	.011	.183	.385	1.000	.178	.048	-.008	
-.217	.251	-.113	.256	.061	.149	.001	.157	.194	.235	.090	.146	-.050	.178	1.000	.130	.225	
-.079	.182	-.102	.400	.057	-.021	-.110	.078	-.156	-.130	.162	-.041	.119	.048	.130	1.000	.009	
.032	.285	.124	-.075	-.052	.075	-.074	.034	.387	.183	.097	-.091	-.168	-.008	.225	.009	1.000	

SPSS Statistics Archivo Editar Ver Datos Transformar Insertar Formato Analizar Gráficos Utilidades Ampliaciones Ventana Ayuda

Resultado77.spv [Documento9] - IBM SPSS Statistics Visor

Matriz de covarianzas entre elementos

	Experiencia_desarrollo	Nivel_experiencia	Desarrollo_diagramas	Nivel_des_diagramas	Domina_jen_guaje	N1_Recuperacion_P1	N1_Recuperacion_P2	N1_Recuperacion_P3	N2_Compresion_P1	N2_Compresion_P2	N2_Compresion_P3	N3_Analisis_P1	N3_Analisis_P2	N3_Analisis_P3	N4_Aplicacion_P1
Experiencia_desarrollo	.088	.064	.058	.063	.029	.010	.021	.031	.034	.035	.020	-.018	.022	.036	.026
Nivel_experiencia	.064	.227	.067	.042	.115	.003	.032	.030	.049	.052	.059	.013	.063	.022	.026
Desarrollo_diagramas	.058	.067	.150	.052	.042	.022	.026	.043	.038	.040	.032	-.018	.041	.034	.025
Nivel_des_diagramas	.063	.042	.052	.113	.013	.006	.035	.051	.017	.018	.045	-.010	.022	.012	.046
Domina_jen_guaje	.029	.115	.042	.013	.252	-.024	.028	.024	.042	.047	.005	.018	.057	.009	.046
N1_Recuperacion_P1	.010	.003	.022	.006	-.024	.104	.019	.023	.010	.011	.026	-.024	.008	.060	.029
N1_Recuperacion_P2	.021	.032	.026	.035	.028	.019	.249	.155	.078	.084	.039	.113	.194	.049	.174
N1_Recuperacion_P3	.031	.030	.043	.051	.024	.023	.155	.230	.122	.126	.046	.069	.121	.050	.204
N2_Compresion_P1	.034	.049	.038	.017	.042	.030	.078	.122	.163	.154	.027	.041	.059	.046	.102
N2_Compresion_P2	.035	.052	.040	.018	.047	.011	.084	.126	.154	.156	.030	.039	.066	.051	.107
N2_Compresion_P3	.020	.059	.032	.045	.005	.026	.039	.046	.027	.030	.181	-.004	.038	.023	.036
N3_Analisis_P1	-.018	.013	-.018	-.010	.038	-.024	.113	.069	.041	.039	-.004	.192	.095	-.003	.080
N3_Analisis_P2	.022	.063	.041	.022	.057	.008	.194	.121	.059	.066	.038	.095	.227	.042	.135
N3_Analisis_P3	.036	.022	.034	.012	.009	.060	.049	.050	.046	.051	.023	-.003	.042	.243	.065
N4_Aplicacion_P1	.026	.026	.025	.046	.046	.029	.174	.204	.102	.107	.036	.080	.135	.065	.241
N4_Aplicacion_P2	.001	-.001	-.024	-.031	-.029	-.003	.054	.003	-.006	.001	-.059	.030	.044	.021	.028
N4_Aplicacion_P3	-.006	-.011	-.012	.035	-.020	-.008	.028	.031	-.002	-.002	.028	.016	.022	-.015	.028
N5_Metacognitivo_P1	.030	.032	.065	.058	.032	.020	.045	.070	.033	.038	.040	.046	.043	.013	.040
N5_Metacognitivo_P2	-.005	.000	-.001	-.003	.020	-.014	.015	.012	.002	.002	.015	.005	.021	-.028	.015
N5_Metacognitivo_P3	.003	-.007	-.009	-.035	.017	.024	.069	.041	.066	.071	-.005	.038	.061	.045	.040
Recursos_motivacion	-.015	.010	-.029	-.021	-.023	-.008	.006	.019	.021	.023	-.005	.020	.012	-.011	.012
Diagnostico_inicio_materia	-.013	-.030	-.012	-.021	-.043	.036	.054	.042	.047	.039	-.036	.020	.030	-.003	.042
Algo_gradual_complejo	-.009	.010	.026	-.012	.018	-.001	.031	.041	.034	.035	-.012	.025	.033	.004	.037
Calificaciones_destrezas	.006	-.012	-.008	-.010	-.045	.003	.006	.019	.021	.012	-.005	-.002	.001	-.022	.001
Competencias_metodologia	-.005	.066	.035	-.025	.027	-.011	.014	.020	.040	.043	.021	.007	.031	-.027	.008
Competencias_soluciones_programacion	.008	.041	.018	-.011	.027	.002	.037	.060	.066	.069	.026	.023	.045	.025	.038
Competencias_Sistemas_inf	.008	.020	-.004	-.011	.027	-.009	.037	.081	.055	.058	-.028	.034	.034	.014	.070
Conoces_motivo_tutor	-.013	-.020	-.002	.022	-.011	-.018	.086	.064	.036	.039	.017	.052	.073	-.046	.053
Interaccion_tutor	-.003	-.026	-.004	.036	-.035	-.016	.018	.048	.004	.005	.021	.014	.015	-.013	.031
Ayudaron_asesorias	-.007	-.027	-.001	-.017	.017	-.003	.078	.034	.036	.028	-.049	.034	.059	.007	.064

SPSS Statistics Archivo Editar Ver Datos Transformar Insertar Formato Analizar Gráficos Utilidades Ampliaciones Ventana Ayuda Resultado77.spv [Documento96] - IBM SPSS Statistics Visor

	N3_Analisis_P3	N4_Aplicacion_P1i	N4_Aplicacion_P2	N4_Aplicacion_P3	N5_Metacognitivo_P1	N5_Metacognitivo_P2	N5_Metacognitivo_P3	Recursosmotivacion	Diagnostico_inicio_materia	Algo_gradual_complejo	Calificaciones_destrezas	Competencias_metodologia	Competencias_soluciones_programacion	Competencias_Sistemas_inf	Conocimientos_tutor	Interaccion_tutor	Ayudaron_asesorias
.036	.026	.001	-.006	.030	-.005	.003	-.015	-.013	-.009	.006	-.005	.008	.008	-.013	-.003	-.007	
.022	.026	-.001	-.011	.032	.000	-.007	.010	-.030	.010	-.012	.066	.041	.020	-.020	-.026	-.027	
.034	.025	-.024	-.012	.065	.001	-.009	-.029	-.012	.026	-.008	.035	.018	-.004	-.002	-.004	-.001	
.012	.046	-.031	.035	.058	-.003	-.035	-.021	-.021	-.012	-.010	-.025	-.011	-.011	.022	.036	-.017	
.009	.046	-.029	-.020	.032	.020	.017	-.023	-.043	.018	-.045	.027	.027	.027	-.011	-.035	.017	
.060	.029	-.003	-.008	.020	-.014	.024	-.008	.036	-.001	.003	-.011	.002	-.009	-.018	-.016	-.003	
.049	.174	.054	.028	.045	.015	.069	.006	.054	.031	.006	.014	.037	.037	.086	.018	.078	
.050	.204	.003	.011	.070	.012	.041	.019	.042	.041	.019	.020	.060	.081	.064	.048	.034	
.046	.102	-.006	-.002	.033	.002	.066	.021	.047	.034	.021	.040	.066	.055	.036	.004	.036	
.051	.107	.001	-.002	.038	.002	.071	.023	.039	.035	.012	.043	.069	.058	.019	.005	.028	
.023	.036	-.059	.028	.040	.015	-.005	-.005	-.036	-.012	-.005	.021	.026	-.028	.017	.021	-.049	
-.003	.080	.030	.016	.046	.005	.038	.020	.020	.025	-.002	.007	.023	.034	.052	.014	-.034	
.042	.135	.044	.022	.043	.021	.061	.012	.030	.033	.001	.031	.045	.034	.073	.015	.059	
.243	.065	.021	-.015	.013	-.028	.045	-.011	-.003	.004	-.022	-.027	.025	.014	-.046	-.013	.007	
.065	.241	.028	.028	.040	.015	.040	.012	.042	.037	.001	.008	.038	.070	.053	.031	.064	
.021	.028	.227	-.021	-.032	.000	.040	.012	.009	.011	.012	-.045	-.009	.002	-.023	-.017	.027	
-.015	.028	-.021	.060	.024	-.007	-.012	.000	-.016	.005	-.010	-.007	-.006	-.016	.027	.034	-.008	
.013	.040	-.032	.024	.252	.009	.033	-.011	-.008	.008	-.011	-.023	-.024	-.013	.013	.010	-.012	
-.028	.015	.000	-.007	.009	.060	.012	.000	-.016	.006	.000	-.004	-.005	.006	.016	-.002	.008	
.045	.040	.040	-.012	.033	.012	.252	.013	-.011	.025	.002	.048	.038	.038	.000	-.019	-.017	
-.011	.012	.012	.000	-.011	.000	.013	.136	.004	.006	.017	.020	.056	.045	.025	.010	.006	
-.003	.042	.009	-.016	-.008	-.016	-.011	.004	.187	.019	.036	-.004	-.009	.023	.036	-.023	.077	
.004	.037	.011	.005	.008	.006	.025	.006	.019	.088	-.005	.016	.008	.029	.030	-.013	.025	
-.022	.001	.012	-.010	-.011	.000	.002	.017	.036	-.005	.136	-.002	-.009	.002	.014	.021	.016	
-.027	.008	-.045	-.007	-.023	-.004	.048	.020	-.004	.016	-.002	.202	.058	.036	.028	-.006	-.019	
.025	.038	-.009	-.006	-.024	-.005	.038	.056	-.009	.008	-.009	.058	.192	.074	-.009	.018	-.034	
.014	.070	.002	-.016	-.013	.006	.038	.045	.023	.029	.002	.036	.074	.192	.034	.007	-.002	
-.046	.053	-.023	.027	.013	.016	.000	.025	.036	.030	.014	.028	-.009	.034	.187	.020	.045	
-.013	.031	-.017	.034	.010	-.002	-.019	.010	-.023	-.013	.021	-.006	.018	.007	.020	.120	.001	
.007	.064	.027	-.008	-.012	.008	-.017	.006	.077	.025	.016	-.019	-.034	-.002	.045	.001	.211	

SPSS Statistics Archivo Editar Ver Datos Transformar Insertar Formato Analizar Gráficos Utilidades Ampliaciones Ventana Ayuda Resultado77.spv [Documento96] - IBM SPSS Statistics Visor

Estadísticas de elemento de resumen

	Media	Mínimo	Máximo	Rango	Máximo / Mínimo	Varianza	N de elementos
Medias de elemento	.671	.064	.936	.872	14,667	.047	30
Varianzas de elemento	.177	.060	.252	.192	4,176	.004	30
Covarianzas entre elementos	.021	-.059	.204	.264	-3,450	.001	30
Correlaciones entre elementos	.112	-.292	.567	1,259	-3,309	.033	30

SPSS Statistics Archivo Editar Ver Datos Transformar Insertar Formato Analizar Gráficos Utilidades Ampliaciones Ventana Ayuda Resultado77.spv [Documento96] - IBM SPSS Statistics Visor

Estadísticas de total de elemento

	Media de escala si el elemento se ha suprimido	Varianza de escala si el elemento se ha suprimido	Correlación total de elementos corregida	Correlación múltiple al cuadrado	Alfa de Cronbach si el elemento se ha suprimido
Experiencia_desarrollo	19,23	22,740	.290	.	.797
Nivel_experiencia	19,48	22,059	.304	.	.797
Desarrollo_diagramas	19,32	22,413	.296	.	.797
Nivel_des_diagramas	19,27	22,799	.229	.	.799
Domina_lenguaje	19,62	22,497	.189	.	.802
N1_Recuperacion_P1	19,26	23,117	.137	.	.802
N1_Recuperacion_P2i	19,70	20,147	.727	.	.775
N1_Recuperacion_P3i	19,49	20,059	.781	.	.773
N2_Compression_P1i	19,34	20,894	.702	.	.780
N2_Compression_P2i	19,33	20,804	.745	.	.779
N2_Compression_P3	19,37	22,688	.192	.	.801
N3_Analisis_P1	19,88	21,911	.376	.	.793
N3_Analisis_P2i	19,80	20,443	.691	.	.778
N3_Analisis_P3	19,54	22,487	.196	.	.802
N4_Aplicacion_P1i	19,53	20,187	.729	.	.775
N4_Aplicacion_P2	19,80	23,432	-.002	.	.811
N4_Aplicacion_P3	19,20	23,389	.083	.	.803
N5_Metacognitivo_P1	19,61	22,263	.239	.	.800
N5_Metacognitivo_P2	20,07	23,425	.068	.	.803
N5_Metacognitivo_P3	19,66	22,184	.256	.	.799
Recursosmotivacion	19,30	23,093	.118	.	.803
Diagnostico_inicio_materia	19,38	22,927	.129	.	.804
Algo_gradual_complejo	19,23	22,697	.306	.	.797
Calificaciones_destrezas	19,30	23,416	.027	.	.806
Competencias_metodologia	19,41	22,761	.159	.	.803
Competencias_soluciones_programacion	19,39	22,220	.299	.	.797
Competencias_Sistemas_inf	19,39	22,155	.315	.	.796
Conoces_motivo_tutor	19,38	22,282	.289	.	.797
Interaccion_tutor	19,28	23,256	.081	.	.804
Ayudaron_asesorias	19,44	22,700	.168	.	.803

Estadísticas de escala

Media	Varianza	Desviación estándar	N de elementos
20,14	23,647	4,863	30

Instrumento 2: Post-test evaluación de conocimiento.

Visible: 37 de 37 variables

	Lenguajes_dominio	N1_Recuperacion_P1	N1_Recuperacion_P2	N1_Recuperacion_P3	N2_Compreension_P1	N2_Compreension_P2	N2_Compreension_P3	N3_Analisis_P1	N3_Analisis_P2	N3_Analisis_P3	M4_Aplicacion_P1	M4_Aplicacion_P2	M4_Aplicacion_P3	N5_Metacognitivo_P1
1	Lenguaje C.C++	1	1	1	1	1	1	1	1	1	1	1	1	1
2	C++ Java.PHP	1	1	1	1	1	1	1	1	1	1	1	1	1
3	Lenguaje C.C++	1	1	1	1	1	1	1	1	1	0	1	1	1
4	ninguno	1	1	1	1	1	1	1	1	1	0	1	1	1
5	Lenguaje C.C++	1	1	1	1	1	1	1	1	1	0	1	1	1
6	ninguno	1	1	1	1	1	1	1	1	1	0	1	1	1
7	ninguno	1	1	1	1	1	1	1	1	1	1	1	1	1
8	no	1	1	1	1	1	1	1	1	1	1	1	1	1
9	por ahora ninguno a la perfeccion	1	1	1	1	1	1	1	1	1	1	1	1	1
10	C++	1	1	1	1	1	1	1	1	1	1	1	1	1
11	Lenguaje C	1	1	1	1	1	1	1	1	1	1	1	1	1
12	Java	1	1	1	1	1	1	1	1	1	0	1	1	0
13	NA	1	1	1	1	1	1	1	1	1	0	1	1	1
14	Lenguaje C.Python	1	1	1	1	1	0	1	1	1	0	1	1	1
15	nada	1	1	1	1	0	0	1	1	1	0	1	1	1
16	Java	1	0	1	1	1	1	1	1	1	1	1	0	1
17	a	1	0	1	1	1	1	1	1	1	0	1	1	1
18	Lenguaje C	1	0	1	1	1	1	1	1	1	1	1	1	1
19	Ninguno, no lo domino pero lo comprendo.	1	0	1	1	1	1	1	1	1	1	1	1	1
20	Java.PHP.Javascript	1	0	1	1	1	1	1	1	1	0	1	1	1
21	ninguno	1	0	1	1	1	1	1	1	1	1	1	1	1
22	ninguno	1	0	1	1	1	1	1	1	1	0	1	1	1
23	C++ Visual Basic Java.Python	1	0	1	1	1	1	1	1	1	0	1	1	1
24	Lenguaje C Java.Python.PHP	1	0	1	1	1	1	1	1	1	0	1	1	1
25	Lenguaje C Java.Python	1	0	1	1	1	1	1	1	1	0	1	1	1
26	Lenguaje C.C++	1	0	1	1	1	1	1	1	1	0	1	1	1
27	Lenguaje C.C++	1	0	1	1	1	1	1	1	1	0	1	1	1
28	Lenguaje C.C++	1	0	1	1	1	1	1	1	1	0	1	1	1
29	Lenguaje C Java.Python.PHP	1	0	1	1	1	1	1	1	1	0	1	1	1
30	Lenguaje C	1	0	1	1	1	1	1	1	1	0	1	1	1
31	ninguno	1	0	1	1	1	1	1	1	1	0	1	1	1
32	ninguno	1	0	1	1	1	1	1	1	1	0	1	1	1

Instrumento 2: Resultados post-test evaluación de conocimiento.

IBM SPSS Statistics Processor está listo Unicode:ACTIVADO Clásico

Fiabilidad

Advertencias
El determinante de la matriz de covarianzas es cero o aproximadamente cero. Las estadísticas basadas en su matriz inversa no se pueden calcular y se visualizan como valores perdidos por el sistema.

Escala: ALL VARIABLES

Resumen de procesamiento de casos

Casos	Válido	N	%
	Válido ^a	94	100,0
	Excluido ^b	0	,0
	Total	94	100,0

a. La eliminación por lista se basa en todas las variables del procedimiento.

Estadísticas de fiabilidad

Alfa de Cronbach	Alfa de Cronbach basada en elementos estandarizados	N de elementos
,823	,841	15

SPSS Statistics Archivo Editar Ver Datos Transformar Insertar Formato Analizar Gráficos Utilidades Ampliaciones Ventana Ayuda Resultado90.sps [Documento91] - IBM SPSS Statistics Visor

Estadísticas de elemento

	Media	Desviación estándar	N
N1_Recuperacion_P1	,95	,226	94
N1_Recuperacion_P2	,17	,378	94
N1_Recuperacion_P3	,51	,503	94
N2_Compresion_P1	,85	,358	94
N2_Compresion_P2	,93	,264	94
N2_Compresion_P3	,88	,323	94
N3_Analisis_P1	,83	,378	94
N3_Analisis_P2	,31	,464	94
N3_Analisis_P3	,57	,497	94
N4_Aplicacion_P1	,44	,499	94
N4_Aplicacion_P2	,31	,464	94
N4_Aplicacion_P3	,91	,281	94
N5_Metacognitivo_P1	,93	,264	94
N5_Metacognitivo_P2	,30	,460	94
N5_Metacognitivo_P3	,88	,323	94

SPSS Statistics Archivo Editar Ver Datos Transformar Insertar Formato Analizar Gráficos Utilidades Ampliaciones Ventana Ayuda Resultado90.sps [Documento91] - IBM SPSS Statistics Visor

Matriz de correlaciones entre elementos

	N1_Recuperacion_P1	N1_Recuperacion_P2	N1_Recuperacion_P3	N2_Compresion_P1	N2_Compresion_P2	N2_Compresion_P3	N3_Analisis_P1	N3_Analisis_P2	N3_Analisis_P3	N4_Aplicacion_P1	N4_Aplicacion_P2	N4_Aplicacion_P3	N5_Metacognitivo_P1	N5_Metacognitivo_P2	N5_Metacognitivo_P3
N1_Recuperacion_P1	1,000														
N1_Recuperacion_P2	-,019	1,000													
N1_Recuperacion_P3	,242	,387	1,000												
N2_Compresion_P1	,567	,110	,427	1,000											
N2_Compresion_P2	,294	-,087	,128	,450	1,000										
N2_Compresion_P3	,209	-,099	,041	,405	,023	1,000									
N3_Analisis_P1	,523	,130	,463	,924	,411	,363	1,000								
N3_Analisis_P2	,158	,617	,654	,279	,014	,100	,303	1,000							
N3_Analisis_P3	,084	-,068	-,068	,063	,002	,222	,125	-,170	1,000						
N4_Aplicacion_P1	,113	,115	,174	,187	,168	-,080	,170	-,123	,193	1,000					
N4_Aplicacion_P2	,158	,617	,654	,279	,014	,100	,303	1,000	-,170	-,123	1,000				
N4_Aplicacion_P3	,437	-,065	,083	,408	,059	,008	,369	,039	,123	,114	,039	1,000			
N5_Metacognitivo_P1	,836	,021	,290	,678	,383	,149	,626	,189	,084	,168	,189	,494	1,000		
N5_Metacognitivo_P2	,154	,633	,638	,272	,008	,092	,295	,975	-,145	-,104	,975	,032	,185	1,000	
N5_Metacognitivo_P3	,651	,077	,372	,870	,527	,279	,804	,243	,021	,187	,243	,363	,779	,237	1,000

SPSS Statistics Archivo Editar Ver Datos Transformar Insertar Formato Analizar Gráficos Utilidades Ampliaciones Ventana Ayuda Resultado90.sps [Documento91] - IBM SPSS Statistics Visor

Matriz de covarianzas entre elementos

	N1_Recuperacion_P1	N1_Recuperacion_P2	N1_Recuperacion_P3	N2_Compresion_P1	N2_Compresion_P2	N2_Compresion_P3	N3_Analisis_P1	N3_Analisis_P2	N3_Analisis_P3	N4_Aplicacion_P1	N4_Aplicacion_P2	N4_Aplicacion_P3	N5_Metacognitivo_P1	N5_Metacognitivo_P2	N5_Metacognitivo_P3
N1_Recuperacion_P1	,051	-,002	,027	,046	,018	,015	,045	,017	,009	,013	,017	,028	,050	,016	,047
N1_Recuperacion_P2	-,002	,143	,073	,015	-,009	-,012	,019	,108	-,013	,022	,108	-,007	,002	,110	-,009
N1_Recuperacion_P3	,027	,073	,253	,077	,017	,007	,088	,153	-,017	,044	,153	,012	,038	,147	,060
N2_Compresion_P1	,046	,015	,077	,128	,043	,047	,125	,046	,011	,033	,046	,041	,064	,045	,101
N2_Compresion_P2	,018	-,009	,017	,043	,070	,002	,041	,002	,000	,022	,002	,004	,027	,001	,045
N2_Compresion_P3	,015	-,012	,007	,047	,002	,104	,044	,015	,036	-,013	,015	,001	,013	,014	-,029
N3_Analisis_P1	,045	,019	,088	,125	,041	,044	,143	,053	,024	,032	,053	,039	,062	,051	,098
N3_Analisis_P2	,017	,108	,153	,046	,002	,015	,053	,216	-,039	-,028	,216	,005	,023	,208	,036
N3_Analisis_P3	,009	-,013	-,017	,011	,000	,036	,024	-,039	,247	,048	-,039	,017	,011	-,033	,003
N4_Aplicacion_P1	,013	,022	,044	,033	,022	-,013	,032	-,028	,048	,249	-,028	,016	,022	-,024	,030
N4_Aplicacion_P2	,017	,108	,153	,046	,002	,015	,053	,216	-,039	-,028	,216	,005	,023	,208	,036
N4_Aplicacion_P3	,028	-,007	,012	,041	,004	,001	,039	,005	,017	,016	,005	,079	,037	,004	,033
N5_Metacognitivo_P1	,050	,002	,038	,064	,027	,013	,062	,023	,011	,022	,023	,037	,070	,022	,066
N5_Metacognitivo_P2	,016	,110	,147	,045	,001	,014	,051	,208	-,033	-,024	,208	,004	,022	,211	,035
N5_Metacognitivo_P3	,047	,009	,060	,101	,045	,029	,098	,036	,003	,030	,036	,033	,066	,035	,104

SPSS Statistics Archivo Editar Ver Datos Transformar Insertar Formato Analizar Gráficos Utilidades Ampliaciones Ventana Ayuda

Resultado90.spv [Documento91] - IBM SPSS Statistics Visor

N5_Metacognitivo_P3 .047 .009 .060 .101 .045 .029 .098 .036 .003 .030 .036 .033 .066 .035 .104

Estadísticas de elemento de resumen

	Media	Mínimo	Máximo	Rango	Máximo / Mínimo	Varianza	N de elementos
Medias de elemento	,651	,170	,947	,777	5,563	,082	15
Varianzas de elemento	,152	,051	,253	,202	4,962	,005	15
Covarianzas entre elementos	,036	-,039	,216	,255	-5,480	,002	15
Correlaciones entre elementos	,260	-,170	1,000	1,170	-5,866	,078	15

Estadísticas de total de elemento

	Media de escala si el elemento se ha suprimido	Varianza de escala si el elemento se ha suprimido	Correlación total de elementos corregida	Correlación múltiple al cuadrado	Alfa de Cronbach si el elemento se ha suprimido
N1_Recuperacion_P1	8,82	9,096	,507	.	,813
N1_Recuperacion_P2	9,60	8,846	,378	.	,816
N1_Recuperacion_P3	9,26	7,827	,625	.	,798
N2_Comprension_P1	8,91	8,229	,721	.	,795
N2_Comprension_P2	8,84	9,340	,265	.	,822
N2_Comprension_P3	8,88	9,309	,215	.	,825
N3_Analisis_P1	8,94	8,146	,718	.	,794
N3_Analisis_P2	9,46	7,993	,620	.	,798
N3_Analisis_P3	9,19	9,554	,012	.	,848
N4_Aplicacion_P1	9,33	9,213	,124	.	,839
N4_Aplicacion_P2	9,46	7,993	,620	.	,798
N4_Aplicacion_P3	8,85	9,289	,274	.	,821
N5_Metacognitivo_P1	8,84	8,845	,587	.	,808
N5_Metacognitivo_P2	9,47	8,015	,619	.	,799
N5_Metacognitivo_P3	8,88	8,470	,671	.	,800

Estadísticas de escala

Media	Varianza	Desviación estándar	N de elementos
9,77	9,837	3,136	15

Sistema de Tutoría Inteligente como herramienta de apoyo en el aprendizaje de algoritmos computacionales a nivel licenciatura.

INFORME DE ORIGINALIDAD

17%

ÍNDICE DE SIMILITUD

FUENTES PRIMARIAS

1	www.scielo.org.mx Internet	3508 palabras — 11%
2	repositorio.ucv.edu.pe Internet	353 palabras — 1%
3	scielo.sld.cu Internet	252 palabras — 1%
4	alobibivicmicha.blogspot.com Internet	206 palabras — 1%
5	sedici.unlp.edu.ar Internet	177 palabras — 1%
6	es.scribd.com Internet	153 palabras — < 1%
7	www.slideshare.net Internet	149 palabras — < 1%
8	www.coursehero.com Internet	122 palabras — < 1%
9	repositorio.ute.edu.ec Internet	101 palabras — < 1%

10	dspace.utpl.edu.ec Internet	64 palabras — < 1%
11	hdl.handle.net Internet	64 palabras — < 1%
12	archive.org Internet	58 palabras — < 1%
13	Cardenas Valdivia, Christian Pedro Carrasco Huachaca, Rodolfo Diaz Diaz, Gilmer Hunter Abad, Javier Alejandro. "Calidad en el Sector Seguridad en Lima Metropolitana", Pontificia Universidad Catolica del Peru - CENTRUM Catolica (Peru), 2021 ProQuest	34 palabras — < 1%
14	calpullihardcore.blogspot.com Internet	26 palabras — < 1%
15	repositorio.uned.ac.cr Internet	25 palabras — < 1%
16	www.redalyc.org Internet	25 palabras — < 1%
17	www.uv.es Internet	24 palabras — < 1%
18	ibero-revistas.metabiblioteca.org Internet	18 palabras — < 1%
19	e-notabene.ru Internet	17 palabras — < 1%
20	es.wikipedia.org Internet	17 palabras — < 1%

21 2014.uat.edu.mx

Internet

15 palabras — < 1%

22

15 palabras — < 1%

dilemascontemporaneoseduccionpoliticayvalores.com

Internet

EXCLUIR CITAS

ACTIVADO

EXCLUIR COINCIDENCIAS < 15 PALABRAS

EXCLUIR BIBLIOGRAFÍA

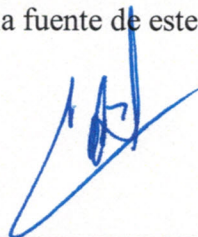
ACTIVADO

UNIVERSIDAD AUTÓNOMA DE TAMAULIPAS
DIVISIÓN DE ESTUDIOS DE POSGRADO E INVESTIGACIÓN

CARTA DE CESIÓN DE DERECHOS

En Ciudad Victoria, Tamaulipas el día 1 del mes de febrero del año 2022, el que suscribe M.S.I. Mario Humberto Rodríguez Chávez alumno del Programa de Doctorado en Gestión e Innovación Educativa con número de matrícula 9724684, adscrito a la Unidad Académica Multidisciplinaria de Ciencias Educación y Humanidades de la Universidad Autónoma de Tamaulipas, manifiesto que soy autor intelectual del presente trabajo de Tesis bajo la dirección del Dr. Arturo Amaya Amaya y cede los derechos del trabajo titulado Sistema de Tutoría Inteligente como Herramienta de Apoyo en el Aprendizaje de Algoritmos Computacionales a Nivel Licenciatura, a la Universidad Autónoma de Tamaulipas para su difusión, con fines académicos y de investigación.

Los usuarios de la información no deben reproducir el contenido textual, gráficas o datos del trabajo sin el permiso expreso del autor y/o director del trabajo. Este puede ser obtenido escribiendo a las siguientes direcciones: mrodriguez@upv.edu.mx o arturo.amaya@docentes.uat.edu.mx. Si el permiso se otorga, el usuario deberá dar el agradecimiento correspondiente y citar la fuente de este.



M.S.I. Mario Humberto Rodríguez Chávez