

# Diseño de un ambiente virtual de aprendizaje soportado en los estilos de aprendizaje

Katherine Roa-Banquez<sup>1</sup>  
Universidad Santo Tomás  
katherinroa@ustadistancia.edu.co

Crisman Martínez-Barrera<sup>2</sup>  
Universidad Santo Tomás  
crismanmartinez@ustadistancia.edu.co

DOI: <https://doi.org/10.21158/2357514x.v8.n2.2020.2761>

**Cómo citar este artículo:** Roa-Banquez, K.; Martínez-Barrera, C. (2020). Diseño de un ambiente virtual de aprendizaje soportado en los estilos de aprendizaje. *Revista Virtu@lmente*, 8(2), 67-86.

DOI: <https://doi.org/10.21158/2357514x.v8.n2.2020.2761>

**Fecha de recepción:** 08 de agosto de 2020

**Fecha de aprobación:** 09 de octubre de 2020



<sup>1</sup> Ingeniera en Telecomunicaciones - Universidad de Pamplona. Especialista en Gerencia de Tecnología - Universidad EAN. Magister en Informática Educativa - Universidad Metropolitana de Chile. Doctora en Educación - Universidad de Baja California. ORCID: <https://orcid.org/0000-0002-8474-8336>

<sup>2</sup> Ingeniero de Sistemas de la Universidad Central de Colombia. Magister en Teleinformática de la Universidad Distrital Francisco José de Caldas. ORCID: <https://orcid.org/0000-0003-4103-6846>



## RESUMEN

En el espacio académico de Algoritmos de Programación de segundo semestre del programa de Ingeniería en Informática en la modalidad a distancia de la Universidad Santo Tomás —Colombia—, se requirió establecer una estrategia para la enseñanza de las diferentes temáticas que se imparten en esta asignatura, dado que actualmente se presenta una alta reprobación en el país. Ante este panorama el objetivo del proyecto es diseñar una herramienta interactiva que permita identificar los estilos de aprendizaje del estudiante para fortalecer su proceso de enseñanza y aprendizaje en dicha asignatura. La investigación es de carácter cuantitativo, con un diseño cuasiexperimental, en la que se cuenta con dos grupos: grupo experimental y otro grupo control. La herramienta interactiva diseñada para Algoritmos de Programación en modalidad a distancia permitió identificar el estilo de aprendizaje de los estudiantes mediante el uso del test de Felder y Silverman, por lo tanto el desarrollo de la herramienta interactiva y los resultados que esta entrega, hacen posible que por cada estudiante y su estilo de aprendizaje, exista una ruta que lo guíe a las actividades, evaluaciones y estrategias pedagógicas para fortalecer así su aprendizaje en las temáticas propias del espacio académico. Esta investigación se desarrolló para disminuir la reprobación en este espacio académico, debido a que, para muchos estudiantes de los diferentes programas de ingeniería, se torna difícil por la gran cantidad de conceptos, términos, lógica de programación y reglas semánticas que se deben aplicar durante su curso.

**Palabras clave:** estilos de aprendizaje; herramienta tecnológica; herramienta interactiva; test de Felder y Silverman; *blended learning*; enseñanza de ingeniería.

# Designing a virtual learning environment supported on the different learning styles

## ABSTRACT

Given the high rate of failure reported at this moment in the country in the academic space of Programming Algorithms, second semester of the program of Computer Science Engineering in the distance modality of Universidad Santo Tomás -Colombia-, it was necessary to establish a strategy for teaching the different topics that are taught in this subject. Faced with this panorama, the aim of this project is to design an interactive tool that allows to identify the learning styles of the students toward strengthening their teaching and learning process in this subject. The research is quantitative, with a quasi-experimental design, and counts with the participation of two groups: an experimental group and a control group. The interactive tool designed for Programming Algorithms in the distance mode allowed us to identify the learning style of the students by using the Felder and Silverman's test and, therefore, the development of the interactive tool and the results it delivered made it possible to have a route for each student and his learning style, that guided him to the activities, evaluations, and pedagogical strategies toward strengthening his learning of the topics within the academic space. This research was developed with a view to decrease the failure in this academic space, because for many students in the different programs of engineering, it becomes difficult to get good results due to the great amount of concepts, terms, programming logic, and semantic rules that must be applied during the course.

**Keywords:** learning styles; technological tool; interactive tool; Felder and Silverman test; blended learning; engineering education.

# Desenho de um ambiente virtual de aprendizagem apoiado por estilos de aprendizagem

## RESUMO

No espaço acadêmico de Algoritmos de Programação do segundo semestre do curso de Engenharia Informática na modalidade a distância da Universidade de Santo Tomás - Colômbia-, foi necessário estabelecer uma estratégia para o ensino dos diferentes temas que oferece essa disciplina, uma vez que atualmente existe uma grande reprovação no país. Neste contexto, o objetivo do projeto é desenhar uma ferramenta interativa que permita identificar estilos de aprendizagem dos alunos para fortalecer o seu processo de ensino e aprendizagem nesta disciplina. A pesquisa é de natureza quantitativa, com desenho quase-experimental, na qual existem dois grupos: um grupo experimental e um grupo controle. A ferramenta interativa desenvolvida para Algoritmos de Programação a Distância possibilitou identificar o estilo de aprendizagem dos alunos por meio da utilização do teste de Felder e Silverman, pois o desenvolvimento da ferramenta interativa e os resultados que ela entrega, possibilitam para cada aluno e seu estilo de aprendizagem, um roteiro que os guie às atividades, avaliações e estratégias pedagógicas para fortalecer sua aprendizagem nas temáticas do espaço acadêmico. Esta pesquisa foi desenvolvida para diminuir a reprovação neste espaço acadêmico, pois, para muitos alunos dos diferentes cursos de engenharia, torna-se difícil devido ao grande número de conceitos, termos, lógica de programação e regras semânticas que devem ser aplicadas durante seu curso.

**Palavras-chave:** estilos de aprendizagem; ferramenta tecnológica; ferramenta interativa; Teste de Felder e Silverman; *blended learning*; educação em engenharia.

# Conception d'un environnement virtuel d'apprentissage focalisé sur les styles d'apprentissage

## RÉSUMÉ

Cet article traite des stratégies d'enseignement proposées lors du cours de second semestre de programmation d'algorithmes du programme de génie informatique en modalité non-présentielle de l'Université de Santo Tomás -Colombie-. L'objectif de ce projet consiste à concevoir un outil interactif permettant d'identifier les styles d'apprentissage des étudiants pour renforcer leur processus d'acquisition des connaissances en la matière. Cette investigation, de type quantitatif et de conception quasi expérimentale, comprend deux groupes distincts: un groupe expérimental et un groupe témoin. Cet outil interactif, conçu pour le cours de Programmation algorithmiques en modalité non-présentielle, a permis d'identifier le style d'apprentissage des apprenants grâce à la mise en place du test de Felder et Silverman. Le développement de cet outil interactif et l'analyse des résultats obtenus permettent de guider chaque apprenant, avec son mode d'apprentissage particulier, vers des activités, des évaluations et des stratégies pédagogiques précises renforçant son apprentissage des programmes académiques. Ce projet d'investigation a été mis en place pour trouver des solutions au mécontentement envers ce programme académique dans la mesure où, pour de nombreux étudiants d'ingénierie, la complexité tient au grand nombre de concepts, de terminologies, de logiques de programmation et de règles sémantiques devant être appliqués et utilisés durant les cours.

**Mots-clés:** styles d'apprentissage; outil technologique; outil interactif; test de Felder et Silverman; *blended learning*; formation en ingénierie.

## 1. Introducción

En el espacio académico de Algoritmos de Programación<sup>3</sup> de segundo semestre del programa de Ingeniería en Informática en la modalidad a distancia de la Universidad Santo Tomás —Colombia—, se requirió establecer una estrategia para la enseñanza de las diferentes temáticas que se imparten en esta asignatura, dado que actualmente se presenta una alta reprobación en el país, siendo Barranquilla y Cali los Centros de Atención Universitaria (CAU) con la mayor tasa de desaprobación, con un 100 %, seguidos por Bogotá, con una tasa del 58,13 %; Armenia y Montería, con el 50 %; Tunja, con el 45,45 %; Facatativá, con el 44 %; Medellín, con el 40 %; Chiquinquirá, con el 37,87 %, y Duitama y Valledupar, con el 14,28 %.

Esta muestra se obtuvo de los periodos comprendidos entre 2015-01 y 2018-02. En general, la tasa de desaprobación para el espacio académico Algoritmos de Programación es del 48,98 %, que entre 202 alumnos para el periodo señalado equivale a 99 estudiantes; lo cual es muy significativo (Ricardo, 2018 y Sandoval y Silva, 2017). Lo anterior soporta el diseño y ejecución de una herramienta interactiva que permita identificar los estilos de aprendizaje de los estudiantes de primer semestre, mediante el uso del test de Felder y Silverman (1988), y así focalizar el desarrollo temático y práctico de una manera más didáctica en su estilo de aprender.

De esta forma se plantea la pregunta de investigación: ¿cuáles serán las características de la herramienta interactiva que permitan identificar los estilos de aprendizaje del estudiante para el fortalecimiento del proceso de enseñanza y aprendizaje de los Algoritmos de Programación? Esta, responde al objetivo general, el cual consiste en diseñar una herramienta interactiva que permita identificar los estilos de aprendizaje del estudiante para fortalecer su proceso de enseñanza y aprendizaje en dicha asignatura.

La investigación es de carácter cuantitativo, con un diseño cuasiexperimental. Se cuenta con dos grupos: uno que recibirá el tratamiento experimental —grupo experimental— y otro que no —grupo control—; a cada grupo se le aplicará un pretest, teniendo en cuenta que todas las condiciones iniciales son las mismas. Al grupo experimental se le aplicará el tratamiento que, para este caso, será la herramienta interactiva, donde el estudiante inicialmente realiza el test de estilos de aprendizaje y es dirigido de forma automática al módulo que responde a sus necesidades de aprendizaje: sensitivos —intuitivos, visuales—, verbales, secuenciales o globales, y activos-reflexivos. Al final, se les aplicó un postest para observar su efecto.

<sup>3</sup> Universidad Santo Tomás. (2019). Universidad Abierta y a Distancia. Decanatura de División de Educación Abierta y a Distancia. Facultad de Ciencias y Tecnologías. Aula Virtual - Algoritmos de Programación. Recuperado de <http://aula20201.ustadistancia.edu.co>

El proceso investigativo se encuentra en su fase 1, cumpliendo los dos momentos propuestos: a) caracterización de los estilos de aprendizaje: identificar los diferentes estilos de aprendizaje de acuerdo con Felder y Silverman (1988); b) diseño de la herramienta interactiva: se diseñó una herramienta interactiva que analiza el estilo de aprendizaje del estudiante y todo el contenido del espacio académico es trabajado desde su forma de aprendizaje. Se espera una segunda fase donde se trabaje un diseño cuasiexperimental con un grupo control y otro experimental para aplicar y validar la herramienta.

En consecuencia, el capítulo se presenta en cuatro apartados: el primero es un marco teórico en donde se abordan los estilos de aprendizaje y los algoritmos de programación; en el segundo, se presenta la metodología trabajada en el proceso investigativo; en el tercero, se abordan los resultados y, por último, en el cuarto se presentan las conclusiones.

## 2. Marco teórico

### 2.1. Estilos de aprendizaje

La teoría de estilos de aprendizaje, que muchos autores definen como «estilos cognitivos», tiene que ver con la manera en que se comportan los estudiantes en el aula al estar frente al conocimiento y cómo estos logran asimilarlo y apropiarse de él de diferentes maneras. Uno de los grandes precursores del tema fue Witkin, quien exponía que los estilos de aprendizaje eran la manifestación de cómo las personas perciben y procesan la información (Guerrero, 2010).

En los años setenta y bajo los pilares Hunt (1979) describe que estos representan los comportamientos distintivos que sirven como indicadores de cómo una persona aprende y se adapta a su ambiente. Así mismo, Felder y Brent (2005), citando a Keefe (1979), expresan que: «Los estilos de aprendizaje son características cognitivas, afectivas y conductas psicológicas que sirven de indicadores relativamente estables de cómo los alumnos perciben, interaccionan y responden con determinado ambiente de aprendizaje» (p. 58). De la misma forma, Castro y Guzmán (2005) señalan que estos son «preferencias que los estudiantes tienen para pensar, relacionarse con otros en diversos ambientes y experiencias» (p. 86).

Otro aporte importante a la teoría de los estilos de aprendizaje está ligado a lo expuesto por Felder y Silverman (1988): un modelo de estilos de aprendizaje que permite diferenciar a los estudiantes de acuerdo con cuál podría ser su manera de aprender, según unas escalas relativas de las formas en que comprenden y procesan la información recibida. Este modelo fue probado en estudiantes de ingeniería y se llamó Índice de Estilos de Aprendizaje (*ILS Index of Learning Styles*). Este modelo describe los siguientes estilos de aprendizaje: sensitivos/intuitivos, visuales/verbales, secuenciales/globales y activos/reflexivos.

**Sensitivos–intuitivos:** las personas sensitivas son pacientes con los detalles; además de ser buenos memorizando hechos, los intuitivos se agobian por los detalles y les apasionan las complicaciones. Estas características son tendencias de los dos tipos de comportamiento. Una distinción importante es que los intuitivos se sienten más cómodos con símbolos, de manera que los estudiantes de programas de

ingeniería son adecuados en este tipo de estilo, dado que requieren consciencia del entorno, atención a los detalles, experimentalidad y practicidad que son las señas de identidad de los sensitivos; muchos otros requieren de la creatividad, la teoría y el talento inspirado en conjeturas que caracterizan a los intuitivos.

Para ser efectiva, la educación en ingeniería debe llegar a ambos tipos. El material presentado debe ser una combinación de información concreta —hechos, datos, observables, fenómenos— y conceptos abstractos —principios, teorías, matemática modelos—. Por lo anterior, los estudiantes de este estilo perciben la información externa o sensitiva a la vista, oído o tacto, e información interna a través del uso de la memoria, notas y lecturas.

**Visuales-verbales:** científicos han establecido que por lo general nuestros cerebros convierten las palabras escritas en sus palabras equivalentes y las procesan de la misma manera que las palabras habladas; por lo tanto, las palabras escritas no son equivalentes a información visual real: para un alumno visual, una imagen de verdad vale mil palabras, ya sean habladas o escritas. Por lo anterior, el estilo de aprendizaje visual y verbal resuelve este problema al permitir que las palabras habladas y escritas se incluyan en la misma categoría.

De esta forma, el estudiante que se clasifique en este estilo podrá recibir la información externa mediante cuadros visuales, diagramas, demostraciones, gráficos y formatos verbales como sonidos y expresiones.

**Secuencial-global:** el aprendizaje de los estudiantes es secuencial a través de pasos incrementales de progreso de sus habilidades y desde el punto de vista de la generalidad a lo particular. Así es posible identificar y autocompletar elementos no vistos en la enseñanza.

**Activo-reflexivo:** los complejos procesos mentales mediante los cuales se percibe la información para ser convertida en conocimiento se pueden agrupar en dos categorías: experimentación activa y reflexiva, donde la primera implica hacer algo en el mundo externo con la información, discutirlo, explicarlo o probarlo de alguna manera; y observación, la cual implica examinar y manipular la información de forma introspectiva. Un «alumno activo» se siente más cómodo con la experimentación activa que con la observación reflexiva y al contrario para un «alumno reflexivo». Hay indicios de que los ingenieros tienen más probabilidades de ser activos que reflexivos.

Los estudiantes activos no aprenden mucho en situaciones que exijan actividades pasivas —como la mayoría de las conferencias— y los reflexivos no aprenden mucho en situaciones que no brinden la oportunidad de pensar sobre la información que se presenta —como la mayoría de las conferencias—. Los alumnos activos trabajan bien en grupos; los alumnos reflexivos trabajan mejor solos o como máximo con otra persona. Los aprendices activos tienden a ser experimentalistas; los estudiantes reflexivos tienden a ser teóricos.

A primera vista parece haber una considerable superposición entre estudiantes activos y reflexivos, los cuales están involucrados en el mundo externo de fenómenos. Los reflexivos prefieren seleccionar la información disponible en el contexto externo y procesarla ya sea de forma activa o reflexiva; en el último caso, postulando explicaciones e interpretaciones, dibujando analogías y formulando modelos. Del mismo modo, el activo selecciona la información generada internamente, pero puede procesarla reflexiva o activamente, en este caso, configurando un experimento para probar la idea o probarla a un colega. Por lo anterior, el estudiante con este estilo de aprendizaje puede aprender y adquirir conocimiento mediante tareas activas o a través de la reflexión, la controversia y la introspección.

## 2.2. Algoritmos de programación

Los algoritmos datan de los orígenes de la agricultura y el nomadismo cuando se hizo necesario resolver problemas repetitivos básicos de conteo y cálculos, y realizar mediciones. Más adelante, el concepto de «algoritmo» se formaliza gracias al matemático y astrónomo árabe Al'Khwarizmi, quien escribió un tratado sobre los números, cuya versión en latín se llama *Algoritmi de Numero Indorum*. Este trabajo permitió simplificar las matemáticas a un nivel práctico y propuso diagramas para leer y comprender los algoritmos.

López *et al.* (2012) describen la necesidad posterior que surgió para los ingenieros, que hizo que los algoritmos se usaran para formular y resolver problemas formalmente. Aguilar (2008) enfatiza en que antes de enunciar un algoritmo es necesario describir paso a paso, en lenguaje natural, la

solución del problema; con base en esto, se debe construir el algoritmo, que es un método o conjunto de reglas para solucionar un problema. Más recientemente, Buriticá (2017) realiza una propuesta metodológica para construir un algoritmo concreto; este autor se basa en el paradigma de la programación funcional y su estrategia es relacionar un tema problema propio para luego resolverlo. Buriticá (2017) define «algoritmo» como un conjunto ordenado y secuencial de pasos que permiten lograr un objetivo claro, concreto y cuantificable.

En los trabajos de investigación del espacio académico Algoritmos de Programación, Roa y Martínez (2020) definen «algoritmo de programación» como un conjunto finito de pasos lógicos, los cuales se enuncian en una secuencia ordenada de operaciones, para resolver un tipo específico de problema. Además, sugieren que para construir un algoritmo se deben tener en cuenta los siguientes aspectos: a) Finito: un algoritmo debe terminar siempre después de un número finito de pasos; no existen restricciones en la cantidad de pasos —mínimos o máximos—. b) Definido: cada paso debe ser definido de una manera precisa —antes o después de—, lo que le permite ejecutarse de una manera rigurosa y específica: no debe haber ambigüedad. c) Datos de entrada: son los valores necesarios para solucionar el problema. d) Datos de salida: es el resultado de la solución del problema. e) Efectividad: que cumpla con el objetivo de resolver o solucionar el problema. 6) Prueba de escritorio: conjunto de valores de entrada que se le entrega a cada una de las variables para probar el algoritmo, de manera tal que su lógica se siga de forma escrita —arriba, abajo, derecha e izquierda—, automática o robotizada, para

determinar el conjunto de datos de salida y comprobar si el algoritmo cumple con el objetivo. Es necesario para el mismo algoritmo elaborar diferentes pruebas, como alternativas de caso se prevean en la solución del mismo.

En síntesis, el término «algoritmo» es usado en las ciencias de la computación para describir un conjunto de pasos organizados y finitos que lleven a desarrollar una solución óptima y precisa, haciendo de la programación una forma y un método que hace uso de diferentes arquetipos para modelar soluciones informáticas. Los arquetipos pueden ser diagramas, mapas, códigos y modelos, entre otros, en lenguaje natural o artificial, y tienen como objetivo generar una solución a un problema planteado.

Como escriben López et al. (2012), los algoritmos son una herramienta necesaria para los profesionales de la ingeniería, por eso, la Universidad Santo Tomás abre estos espacios académicos a los estudiantes de primeros semestres, permitiendo así el desarrollo, el fortalecimiento y la construcción de la lógica de programación con el fin de que formulen soluciones computacionales a través de ellos. Además, este espacio académico permite resolver la pregunta que orienta la investigación y se centra en obtener el diseño del ambiente virtual soportado en los estilos de aprendizaje para mejorarlo, incrementar el desempeño y aumentar la adquisición de los conocimientos establecidos en el *syllabus*.

El *syllabus* guía los contenidos de aprendizaje para el espacio académico Algoritmos de Programación y propone como núcleo las siguientes preguntas problematizadoras: a) ¿Cómo fundamentar las bases de la algoritmia desde

los aspectos cognitivo, teórico y práctico por parte de los estudiantes? b) ¿Cómo concretar los conocimientos mediante el uso de diferentes herramientas de programación? c) ¿Cómo lograr que los futuros profesionales adquieran la capacidad de aplicar los protocolos, las directivas, las reglas, las estrategias y las técnicas de diseño para modelar, simular e implementar algoritmos en los sistemas de información y de conocimiento a partir de la solución de problemáticas de la vida real, aplicados en los contextos regional, nacional e internacional?

Una vez termine la investigación y se use la herramienta, se complementará con las siguientes preguntas problematizadoras: a) ¿cómo apropiar, desarrollar y fortalecer la lógica de programación mediante objetos virtuales de aprendizaje (OVA), construidos de acuerdo con los estilos de aprendizaje propios de cada estudiante? Y b) ¿cómo puede el alumno proponer soluciones algorítmicas a problemáticas en los contextos local, nacional e internacional?

Los objetos virtuales de aprendizaje, construidos de acuerdo con los estilos de aprendizaje del estudiante, contemplan las siguientes temáticas del *syllabus* para el ciclo académico 2020: a) Métodos de resolución de problemas algorítmicos. b) Variables. c) Constantes. d) Operadores de relación. e) Operadores lógicos. f) Operadores matemáticos. g) Operadores de asignación. h) El condicional «si» y sus variantes. i) El condicional «si no» y sus variantes. j) Las instrucciones de repetición y sus variantes. k) El uso de métodos y funciones. l) El paso de objetos, variables y valores.

Para cada una de estas temáticas se determinan sus atributos y características —tópicos—, con el fin de diseñarlos y poder construir un OVA por cada estilo de aprendizaje; por ejemplo, la enseñanza del condicional «si» tendrá un conjunto de cuatro objetos virtuales de aprendizaje para el estilo activo-reflexivo, otro para el sensitivo-intuitivo, otro para el visual-verbal y por último, para el secuencial-global. De esta forma, cuando un estudiante ingrese al ambiente virtual tendrá en su plataforma un conjunto de OVA según su estilo de aprendizaje predominante.

### 3. Metodología

El presente capítulo tiene como finalidad identificar los estilos de aprendizaje de los estudiantes del espacio académico de Algoritmos de Programación, con el fin de contribuir al diseño de una herramienta interactiva, que sea coherente con la modalidad a distancia y acorde al Modelo Educativo Pedagógico de la Universidad Santo Tomás (2010). Para ello fue indispensable realizar un proceso de recolección y análisis de datos que permitiera correlacionar los resultados de las diferentes variables de la investigación y, de esta forma, responder a la pregunta de investigación.

Por lo anterior, esta investigación es de carácter cuantitativo, dado que busca dar una solución al problema identificado. A saber, el alto grado de repitencia de los estudiantes del espacio académico de Algoritmos de Programación, analizando todos sus aspectos y proponiendo una mejora a la situación de forma práctica, académica y funcional, por medio de la herramienta interactiva, la cual analiza el estilo de

aprendizaje del estudiante y le permite estudiar las temáticas del curso mediante recursos y actividades didácticas acordes a este.

Teniendo en cuenta lo anterior, se trabajó un diseño preexperimental, en el cual se contó con un grupo de estudiantes al que se le aplicó una prueba previa —pretest— de conocimientos; después, se le aplicó la herramienta interactiva de identificación de estilo de aprendizaje y, por último, una prueba al tratamiento —postest—. Esto, con el fin de tener un punto de referencia. Después, para el segundo grupo, se proyectará realizar las mismas pruebas y así identificar si la herramienta didáctica tuvo un impacto o no.

El proceso se desarrolló en dos etapas:

**Etapa 1. Caracterización de los estilos de aprendizaje:** identificar los diferentes estilos de aprendizaje de acuerdo con Felder y Silverman (1988).

**Etapa 2. Diseño de la herramienta interactiva:** teniendo en cuenta los resultados arrojados en la primera fase, se diseñará una herramienta interactiva que sea coherente con la modalidad a distancia y el Modelo Educativo Pedagógico de la universidad.

Se espera que en un futuro el proyecto permita aplicar y validar la herramienta, teniendo como referente un diseño cuasiexperimental con un grupo control y otro experimental, lo que haga posible realizar un análisis comparativo para identificar el fortalecimiento de los conocimientos en el campo de formación de Algoritmos de Programación.

## 4. Resultados

### 4.1. Diseño del ambiente virtual de aprendizaje

Cuenta con dos puntos de entrada para el estudiante. a) Identificación del estilo de aprendizaje —con base en el test Felder y Silverman— cuando ingresa por primera vez a la herramienta. b) Diseño del ambiente virtual de aprendizaje con OVA de acuerdo con los estilos del estudiante. Con base en el estilo de aprendizaje identificado, se define automáticamente la ruta de enseñanza para el estudiante.

El test de estilos de aprendizaje —Felder y Silverman— tiene cuarenta y cuatro preguntas, que al finalizar permiten identificar el estilo de aprendizaje del estudiante.

El ambiente virtual de aprendizaje, por su parte, contiene cuatro temas fundamentales. a) El concepto teórico: explicación conceptual de cada uno de los elementos que se están enseñando. b) El concepto semántico: incorpora la forma, la sintaxis y la semántica de cada elemento de un algoritmo. c) El concepto de refuerzo: incorpora elementos nuevos, con ejemplificación de las problemáticas vistas desde otras disciplinas y vivencias. d) El ejercicio práctico: permite mostrar de forma práctica el concepto que se está enseñando.

**Figura 1.** Elementos del ambiente virtual de aprendizaje



**Fuente.** Elaboración propia.

La figura 1 muestra los componentes de cada temática del espacio académico Algoritmos de Programación; con estos tópicos identificados, se construyen los OVA para cada estilo de aprendizaje.

## 4.2. Arquitectura general de ambiente virtual de aprendizaje

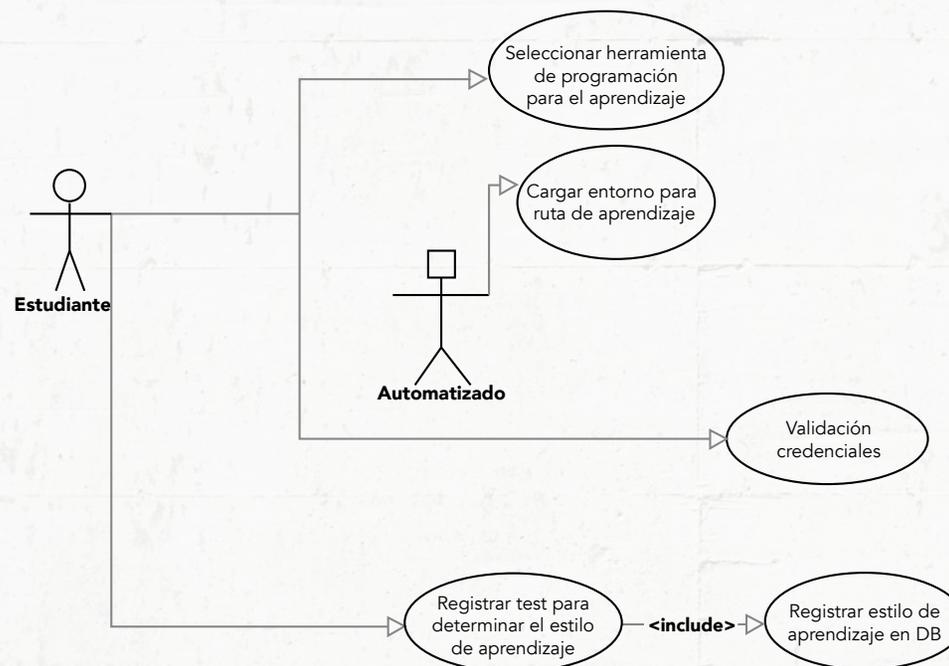
Se usa el Lenguaje de Modelamiento Unificado (UML) para presentar los casos de uso del ambiente virtual de aprendizaje. Los siguientes fueron los casos de uso identificados:

- Superadministrador: actor que tiene a cargo la configuración del ambiente virtual.
- Docente: el tutor virtual a distancia.

- Estudiante: actor al que, a través de la autenticación en el ambiente virtual, se le genera la ruta de aprendizaje de acuerdo con las características de su estilo de aprendizaje, posterior a la realización de la identificación de test.

Los siguientes son los casos de uso que tiene el estudiante cuando ingresa a la herramienta interactiva de aprendizaje:

Figura 2. Casos de uso del estudiante

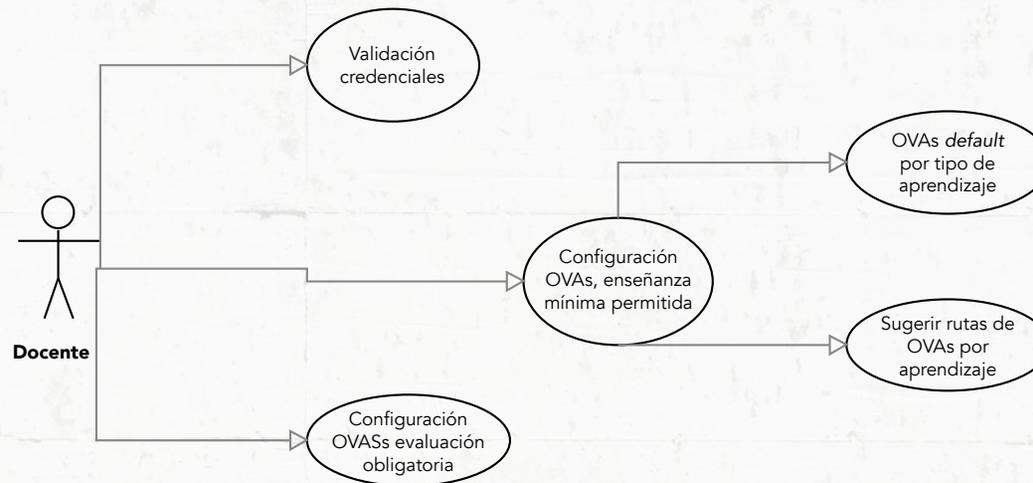


Fuente. Elaboración propia.

El estudiante se autentica en la herramienta y se carga automáticamente la ruta del aprendizaje de acuerdo con el estilo de aprendizaje (véase la figura 2).

Los siguientes son los casos de uso que tiene el docente o tutor virtual a distancia cuando ingresa a la herramienta interactiva de aprendizaje:

**Figura 3.** Casos de uso del docente o tutor

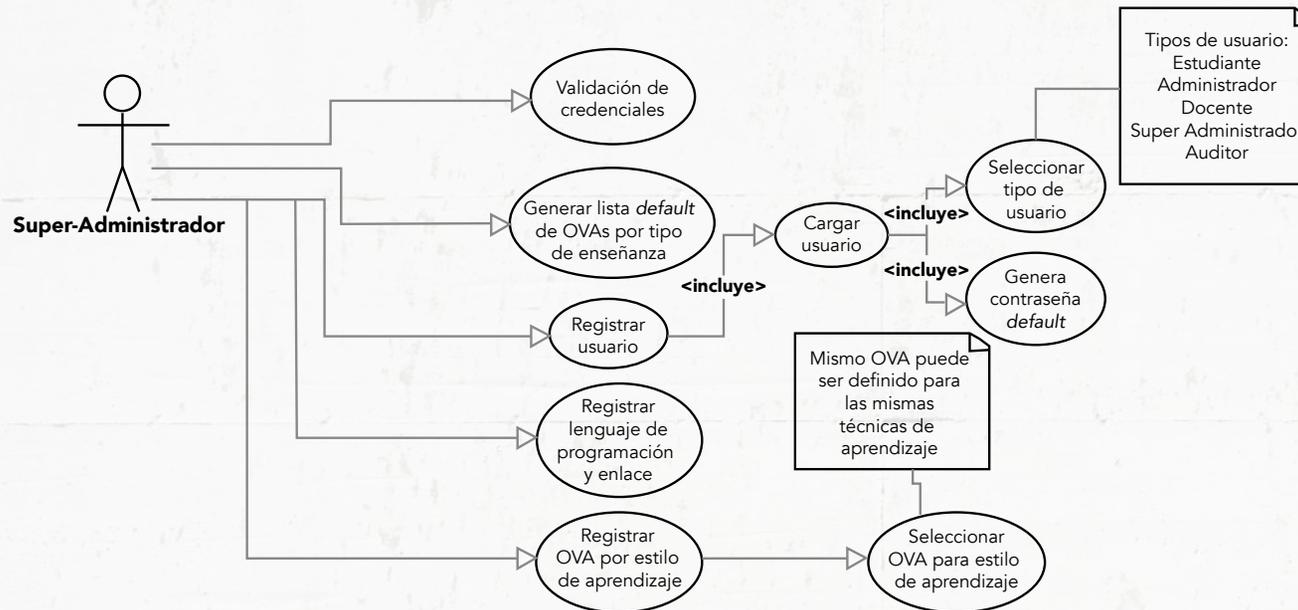


**Fuente.** Elaboración propia.

Una vez registrado, el docente está encargado de obtener las notas de los estudiantes y de redefinir y sugerirles rutas de aprendizaje (véase la figura 3).

Los siguientes son los casos de uso que tiene el superadministrador cuando ingresa a la herramienta interactiva de aprendizaje:

Figura 4. Casos de uso del superadministrador



Fuente. Elaboración propia.

El superadministrador genera automáticamente —ruta *default*— las rutas de aprendizaje de acuerdo con los estilos de aprendizaje; esto lo pueden ver los docentes de las diferentes regiones del país.

### 4.3. Objetos virtuales de aprendizaje de acuerdo con los estilos de aprendizaje

A continuación, se muestran algunos ejemplos de OVA dispuestos en el ambiente interactivo que son utilizados en la enseñanza de las temáticas del espacio académico Algoritmos de Programación.

**Tabla 1.** Objetos virtuales por estilo de aprendizaje

Tipo de OVA	Sensitivos-intuitivos	Visuales- verbales	Secuenciales-globales	Activos-reflexivos
Notación	X			
Mapa mental		X	X	
Audio		X		
Organizar elementos			X	
Ejercicios interactivos prácticos				X

**Fuente.** Elaboración propia.

Una vez asociados los objetos virtuales a cada uno de los tipos de aprendizaje se sugiere construir el OVA para la enseñanza de cada tópico que conforma la temática del espacio académico Algoritmos de Programación (véase la tabla 1), definidos en el syllabus.

A continuación, se muestran algunos OVA dispuestos en el ambiente interactivo que son utilizados en la enseñanza de las temáticas del espacio académico Algoritmos de Programación.

**Tabla 2.** OVA interactivos para la enseñanza del condicional

Condicional básico	Condicional «si no»	Condicional con más de un condicional en el mismo nivel	Condicional múltiple
Modelo didáctico operativo: arrastrar y soltar	Estudio de caso: elección de alternativas falso/verdadero	Aprendizaje por descubrimiento: selección de elementos de tecnología usando condiciones	Aprendizaje basado en problemas: tablero para bailar —semántica del condicional—

**Fuente.** Elaboración propia.

En la tabla 2 se muestran los elementos didácticos, como por ejemplo objetos de arrastrar y soltar, selección de alternativas falso/verdadero, selección de elementos de tecnología usando condiciones y objetos similares a los del baile, en donde aparecen elementos que integran las palabras reservadas del condicional.

A continuación, se presenta el modelo didáctico específico de arrastrar y soltar para desarrollar la lógica algorítmica.

**Figura 5.** Ejemplo actividad de refuerzo: modelo didáctico arrastrar y soltar

Del conjunto de oraciones que a continuación se dan, ordénalas de forma lógica: Así se presentan inicialmente:

- |    |                      |     |                      |
|----|----------------------|-----|----------------------|
| 1. | <input type="text"/> | 7.  | <input type="text"/> |
| 2. | <input type="text"/> | 8.  | <input type="text"/> |
| 3. | <input type="text"/> | 9.  | <input type="text"/> |
| 4. | <input type="text"/> | 10. | <input type="text"/> |
| 5. | <input type="text"/> | 11. | <input type="text"/> |
| 6. | <input type="text"/> |     |                      |

Voy a esperar el bus	Si está lloviendo	Paso la tarjeta
Si está haciendo sol	No llevo saco	Subo al bus
Llevo sombrilla	Bajo del bus	Entro al centro comercial
Salir del apartamento	Llego a la ubicación en donde se encuentra el centro comercial	

**Fuente.** Elaboración propia.

En el desarrollo de la lógica algorítmica se utiliza el lenguaje natural. En este ejemplo (véase figura 5) se muestra un conjunto de pasos para que el estudiante arrastre y suelte en orden cronológico.

#### 4.4. Aprendizaje del estudiante con la herramienta interactiva

Una vez concluye el aprendizaje de cada una de las temáticas incorporadas en el ambiente virtual de aprendizaje del espacio académico Algoritmos de Programación, el estudiante puede realizar simulaciones y prácticas cercanas a la evaluación final.

Además, al concluir el estudio de las temáticas de Algoritmos de Programación de forma independiente y autónoma, el estudiante estará en capacidad de proponer soluciones a problemáticas locales, nacionales e internacionales, así como de usar una herramienta de programación —Java, C++, PHP y Python, entre otras— para programarla computacionalmente. A continuación, se presenta en forma de algoritmo una solución a la que el estudiante debe llegar después de usar la herramienta interactiva de estilos de aprendizaje (véase figura 6).

Ejemplo de la problemática: dados por consola tres precios de tres diferentes referencias de robots, es necesario hallar los precios extremos de los robots —los cuales son producidos por diferentes fabricantes—. La siguiente es la solución a la que el estudiante, una vez utilizada la herramienta de aprendizaje interactiva, debe alcanzar de forma precisa.

Figura 6. Algoritmo en lenguaje pseudocódigo

```
Inicio ExtremosRoboticos
{
    entero primerprecio;
    entero segundoprecio;
    entero tercerprecio;

    cadena primernombre;
    cadena segundonombre;
    cadena tercernombre;

    Escriba ("Digite el nombre del robot");
    Lea (primernombre);
    Escriba ("Digite el precio del robot");
    Lea (primerprecio);

    Escriba ("Digite el nombre del robot");
    Lea (segundonombre);
    Escriba ("Digite el precio del robot");
    Lea (segundoprecio);

    Escriba ("Digite el nombre del robot");
    Lea (tercernombre);
    Escriba ("Digite el precio del robot");
    Lea (tercerprecio);

    si((primerprecio>segundoprecio)AND(primerprecio>tercerprecio)) entonces
        Escriba ("este valor es el externo superior: ",primerprecio);
        si((segundoprecio>tercerprecio)) entonces
            Escriba ("este valor es el externo inferior: ",tercernombre);
        sino
            Escriba ("este valor es el externo inferior: ",segundonombre);
        fin
    fin
    fin

    si((segundoprecio>primerprecio)AND(segundoprecio>tercerprecio)) entonces
        Escriba ("este valor es el extrem superior: ", segundonombre);
        si((primerprecio>tercerprecio)) entonces
            Escriba ("este valor es el extremo inferior: ",tercernombre);
        sino
            Escriba("este valor es el extremo inferior: ",primernombre);
        fin
    fin
    fin

    si((tercerprecio>primerprecio)&&(tercerprecio>segundoprecio)) entonces
        Escriba ("este valor es el externo superior: ",tercernombre);
        si((primerprecio>segundoprecio)) entonces
            Escriba ("este valores el externo inferior: ",segundonombre);
        sino
            Escriba("este valor es el extremo inferior: ",primernombre);
        fin
    fin
    fin

FinSeudocodigo ExtremosRoboticos
```

Fuente. Elaboración propia.

En la figura 6 se muestra un algoritmo en el lenguaje pseudocódigo que resuelve una determinada problemática y que el estudiante, al finalizar el aprendizaje mediante la herramienta interactiva de enseñanza, está en la capacidad de formular.

## 5. Conclusiones

El espacio académico Algoritmos de Programación fue seleccionado como objetivo para realizar esta investigación por presentar una alta tasa de desaprobación en el ámbito nacional. Es de anotar que los estudiantes tienen orígenes socioculturales diferentes —creencias, preconceptos y usan diferentes técnicas de resolución de problemas—, lo que fortalece la identificación de los estilos de aprendizaje para la enseñanza del espacio académico en su modalidad a distancia.

Se estudió el test de Felder y Silverman para obtener los OVA apropiados para cada estilo de aprendizaje —activo-reflexivo, sensitivo-intuitivo, visual-verbal y secuencial-global—.

Se diseñó el ambiente virtual de aprendizaje para soportar los estilos de aprendizaje en la enseñanza de las temáticas incorporadas en el *syllabus* de Algoritmos de Programación.

Para el periodo 2020-02 se proyecta obtener los datos del segundo grupo, el cual utilizará la herramienta interactiva, para la segunda fase de investigación.

Este ambiente interactivo virtual de aprendizaje puede incorporar diferentes OVA, por eso la herramienta puede ser utilizada en otros espacios académicos para formar a los estudiantes a distancia y retomar las ventajas de la identificación de los estilos de aprendizaje en la formación académica de los estudiantes de las diferentes regiones del país.

## Referencias

- Aguilar, L. (2008). *Fundamentos de Programação-: Algoritmos, estruturas de dados e objetos*. Brasil: AMGH Editora.
- Buriticá, O. I. T. (2017). Propuesta metodológica para construir un algoritmo que determine si un número es perfecto usando programación imperativa. *Revista Facultad De Ciencias Básicas*, 13(2), 83-89. DOI: <https://doi.org/10.18359/rfcb.2758>
- Castro, S.; Guzmán, B. (2005). Los estilos de aprendizaje en la enseñanza y el aprendizaje: una propuesta para su implementación. *Revista de Investigación*, (58), 83-102.
- Felder, R. M.; Silverman, L. K. (1988). Learning and teaching styles in engineering education. *Engineering education*, 78(7), 674-681.
- Felder, R.; Brent, R. (2005). Understanding Student Differences. *Journal of Engineering Education*, 94(1), 57-72. DOI: <https://doi.org/10.1002/j.2168-9830.2005.tb00829.x>

- Guerrero, M. (2010). Estilos de aprendizaje y rendimiento académico en alumnos que cursaron genética clínica en el periodo de primavera 2009 en la Facultad de Medicina de la Benemérita Universidad Autónoma de Puebla. *Revista de Estilos de Aprendizaje*, 3(5), 42-52.
- Hunt, D. E. (1979). *Learning Styles and Student's Needs: an introduction to conceptual level*. En *National Association of Secondary School Principals Student's learning styles: diagnosing and prescribing programs*. Reston: National Association of Secondary School Principals
- Joyanes, L. (2008). *Fundamentos de programación. Algoritmos, estructura de datos y objetos*. McGraw Hill.
- López, A. J. G.; Fernández, R. M. P.; Fernández, P. R. (2012). Una Asignatura on-line de Historia de los Algoritmos. *Revista Iberoamericana de Tecnologías del Aprendizaje: IEEE-RITA*, 7(1), 7-12.
- Ricardo, P. (2018). *Informe: posibles causas de la reprobación de espacios académicos por parte de los estudiantes de la Facultad de Ciencias y Tecnologías y estrategias para disminuir esta situación*. Documento institucional inédito.
- Roa, K.; Martínez, C. (2020). *Guion de contenido de autoría para la producción de recursos educativos virtuales de aprendizaje de la Universidad Santo Tomás*. Bogotá: Decanatura de División de Educación Abierta y a Distancia. Universidad Santo Tomás. Documento institucional inédito.
- Sandoval, M.; Silva, A. (2017). *Informe espacios académicos que más perdieron los estudiantes en la facultad de ciencias y tecnologías 2016 01 2016 02*. Bogotá: Programa de Ingeniería en Informática. Documento institucional inédito.
- Universidad Santo Tomás. (2010). *Modelo Educativo Pedagógico*. Recuperado de <https://bit.ly/3aff1Fk>