A Methodology for Designing Programming-Based Learning Experiences to Foster Computational Thinking in Mathematics Education

Christopher Nikulin[1]; Mauricio Castillo-Vergara[2,3]; Cristian Geldes[2]; Rodrigo Ortiz-Henriquez[2]; Enrique Vazquez[1]

[1] Facultad de Ingeniería, Universidad Alberto Hurtado, Santiago, Chile *cnikulin@uahurtado.cl , evazuqez@uahuartado.cl*

[2]Facultad de Economía y Negocios, Universidad Alberto Hurtado, Santiago, *mhcastillo@uahurtado.cl, cgeldes@uahurtado.cl, rortiz@uahurtado.cl*

[3]*Vicerrectoría de Investigación y Postgrado, Universidad de La Serena, La Serena 1700000, Chile*

[4]*CITAGRO, Centro de Investigación para la Transformación Agroalimentaria, Santiago, Chile (ANID CITAGRO CIN250015)*

Abstract*– The integration of programming into mathematics education has gained increasing relevance due to its potential to strengthen computational thinking and support the understanding of mathematical concepts through interactive learning experiences. However, many teachers face difficulties integrating programming tools into their teaching practices due to the lack of methodologies that systematically guide the design of these experiences. In response to this need, this research proposes a methodology for designing programming-based learning experiences in mathematics education. The proposal integrates three complementary dimensions—gamification, BPMN 2.0 process modeling, and programming—organized into six stages that guide teachers from the identification of a pedagogical need to the implementation and evaluation of the learning experience. The methodology is illustrated through two case studies using GeoGebra and Scratch. In addition, perceived workload was evaluated using the NASA-TLX instrument with pre-service mathematics teachers (n = 13). The results show a moderate workload, low levels of frustration, and high perceived performance. Overall, the methodology provides a systematic and adaptable procedure to support teachers in designing programming-based learning experiences..

Keywords*-- Mathematics education, educational innovation, computational thinking, educational programming, mathematics teaching, BPMN 2.0, gamification.

Una metodología para el diseño de experiencias de aprendizaje basadas en programación para fortalecer el pensamiento computacional en la enseñanza de la matemática

Christopher Nikulin[1]; Mauricio Castillo-Vergara[2,3]; Cristian Geldes[2,4]; Rodrigo Ortiz-Henriquez[2]; Enrique Vazquez[1]

[1] Facultad de Ingeniería, Universidad Alberto Hurtado, Santiago, Chile *cnikulin@uahurtado.cl , evazuqez@uahuartado.cl*

[2]Facultad de Economía y Negocios, Universidad Alberto Hurtado, Santiago, *mhcastillo@uahurtado.cl, cgeldes@uahurtado.cl, rortiz@uahurtado.cl*

[3]*Vicerrectoría de Investigación y Postgrado, Universidad de La Serena, La Serena 1700000, Chile*

[4]*CITAGRO, Centro de Investigación para la Transformación Agroalimentaria, Santiago, Chile (ANID CITAGRO CIN250015)*

Resumen: ***La incorporación de la programación en la enseñanza de la matemática ha adquirido una creciente relevancia debido a su potencial para fortalecer el pensamiento computacional y apoyar la comprensión de conceptos matemáticos mediante experiencias de aprendizaje interactivas. Sin embargo, muchos docentes presentan dificultades para integrar herramientas de programación en sus prácticas debido a la ausencia de metodologías que orienten sistemáticamente el diseño de estas experiencias. En respuesta a esta necesidad, la presente investigación propone una metodología para el diseño de experiencias educativas basadas en programación para la enseñanza de la matemática. La propuesta integra tres dimensiones complementarias —gamificación, modelamiento mediante BPMN 2.0 y programación— organizadas en seis etapas que orientan al docente desde la identificación de una necesidad pedagógica hasta la implementación y evaluación de la experiencia. La metodología se ilustra mediante dos casos de estudio utilizando GeoGebra y Scratch. Asimismo, se evaluó la carga de trabajo percibida mediante el instrumento NASA-TLX en estudiantes de Pedagogía en Matemáticas (n = 13). Los resultados muestran una carga de trabajo moderada, bajos niveles de frustración y una alta percepción de rendimiento. En conjunto, la metodología proporciona un procedimiento sistemático y adaptable para apoyar a los docentes en el diseño de experiencias educativas basadas en programación.***

Keywords—Educación, innovación educativa, pensamiento computacional, programación educativa, enseñanza de la matemática, BPMN 2.0, gamificación.

I. Introducción

La incorporación de la programación en la educación matemática ha adquirido una creciente relevancia debido a que proporciona un entorno interactivo para representar, explorar y comprender conceptos matemáticos de mayor complejidad. Más que desarrollar habilidades de programación, estas herramientas permiten visualizar relaciones entre variables, experimentar con modelos matemáticos y fortalecer la resolución de problemas mediante la interacción con representaciones dinámicas. Desde los trabajos pioneros de Papert (1980), se ha planteado que el aprendizaje se fortalece cuando los estudiantes construyen artefactos digitales significativos, dando origen al enfoque construccionista como base para integrar programación y aprendizaje matemático. Posteriormente, el desarrollo de entornos como Scratch, GeoGebra, Python y otras plataformas de programación educativa ha ampliado las posibilidades para abordar funciones, geometría, estadística y modelamiento matemático desde una perspectiva visual, experimental e interactiva [1][2][3]. En este contexto, la programación se ha consolidado como un recurso que favorece la comprensión de conceptos abstractos y promueve un aprendizaje más activo a través de la experimentación y la retroalimentación inmediata [4].

En paralelo, la literatura sobre pensamiento computacional ha destacado la importancia de habilidades como la abstracción, la descomposición de problemas, el reconocimiento de patrones y el diseño de algoritmos como competencias fundamentales [5][6]. A su vez, existe evidencia que la integración del pensamiento

computacional en distintas asignaturas favorece el compromiso estudiantil, la comprensión conceptual y el desarrollo de habilidades de resolución de problemas y colaboración [7].

Estas competencias presentan una relación natural con la matemática escolar, particularmente en contenidos asociados al álgebra, funciones y modelamiento. Sin embargo, diversos estudios indican que muchos docentes presentan dificultades para integrar programación en sus prácticas debido a la falta de formación específica, tiempo de planificación y metodologías que permitan traducir contenidos curriculares en actividades programables [8][9].

Por otra parte, enfoques como la gamificación [10][11], el diseño centrado en el usuario [12], el aprendizaje basado en proyectos y el Design Thinking han demostrado efectos positivos sobre la motivación y el compromiso estudiantil. No obstante, la mayoría de estas propuestas se enfocan en el estudiante y existe una limitada cantidad de investigaciones que proporcionen un procedimiento sistemático para que los profesores aprendan programación y, simultáneamente, diseñen experiencias de aprendizaje basadas en ella.

En este contexto, surge la necesidad de metodologías que permitan a los docentes transitar desde la enseñanza tradicional hacia experiencias educativas mediadas por programación. La propuesta desarrollada en esta investigación aborda esta brecha mediante una metodología estructurada en tres dimensiones y seis etapas, que guía al profesor desde la identificación de un contenido matemático hasta su transformación en una experiencia gamificada y programable. La integración de herramientas como Scratch, GeoGebra, Excel o R, junto con técnicas de BPMN para modelar la interacción entre docente, estudiante y software, proporciona un marco replicable para la innovación pedagógica. De esta forma, la metodología no solo promueve el aprendizaje de programación por parte de los profesores, sino que también facilita la creación sistemática de experiencias de aprendizaje que fortalecen la comprensión matemática, la motivación y el pensamiento computacional.

II. Estado del Arte

1. Creatividad docente, diseño de experiencias y enseñanza de la matemática

La educación contemporánea ha evolucionado desde modelos centrados en la transmisión de contenidos hacia enfoques que promueven la participación activa del estudiante en la construcción de su propio aprendizaje [13]. En este contexto, el rol del docente ha transitado desde ser un expositor de conocimientos hacia un diseñador de experiencias de aprendizaje capaces de conectar contenidos curriculares con situaciones significativas y motivadoras [14]. Esta transformación resulta particularmente relevante en la enseñanza de la matemática, donde conceptos abstractos suelen generar dificultades de comprensión y baja motivación estudiantil.

La creatividad docente emerge como una competencia fundamental para transformar contenidos matemáticos tradicionales en actividades dinámicas, contextualizadas y orientadas a la resolución de problemas [15][16]. Diversas investigaciones destacan que los profesores que diseñan experiencias activas favorecen mayores niveles de compromiso, comprensión conceptual y transferencia del aprendizaje [4]. En este sentido, la planificación de actividades basadas en desafíos, narrativas y exploración guiada constituye una estrategia efectiva para acercar contenidos matemáticos a contextos cercanos a los estudiantes.

2. Pensamiento computacional y programación en educación matemática

La incorporación del pensamiento computacional en los sistemas educativos ha impulsado nuevas formas de abordar la enseñanza de la matemática. El pensamiento computacional se define como un conjunto de habilidades relacionadas con la abstracción, el reconocimiento de patrones, la descomposición de problemas y el diseño de algoritmos [5]. Estas competencias presentan una estrecha relación con el razonamiento matemático, ya que ambas disciplinas requieren modelar situaciones, establecer relaciones funcionales y construir soluciones lógicas.

Asimismo, las experiencias basadas en pensamiento computacional permiten a los estudiantes desarrollar simultáneamente habilidades de resolución de problemas, colaboración y perseverancia, favoreciendo procesos de aprendizaje más profundos y significativos [7].

Herramientas como Scratch, GeoGebra, Python y R han demostrado ser recursos efectivos para facilitar la comprensión de conceptos matemáticos mediante simulaciones, representaciones dinámicas y exploración interactiva [1][2][3]. Particularmente, Scratch permite que estudiantes y docentes desarrollen programas utilizando bloques visuales, reduciendo las barreras técnicas asociadas a

la programación tradicional y favoreciendo la construcción progresiva de modelos matemáticos [1].

Sin embargo, la literatura evidencia que muchos docentes presentan dificultades para incorporar programación en sus clases debido a la falta de formación específica y a la ausencia de metodologías que orienten la transformación de contenidos curriculares en experiencias programables [9]. La mayoría de las investigaciones se enfocan en los resultados del aprendizaje de los estudiantes, mientras que existe una menor cantidad de estudios orientados a proporcionar procedimientos sistemáticos para apoyar el diseño pedagógico de experiencias basadas en programación.

3. Gamificación, experiencia del usuario y modelamiento de experiencias educativas

La gamificación ha sido ampliamente utilizada como estrategia para aumentar la motivación y el compromiso estudiantil mediante la incorporación de elementos propios del juego, tales como misiones, desafíos, recompensas, niveles y narrativas [10]. Desde la perspectiva de la teoría metamotivacional de Apter [17], estas estrategias favorecen estados paratélicos caracterizados por la exploración, el disfrute y la reducción de la ansiedad asociada al error, generando condiciones favorables para el aprendizaje de contenidos complejos.

Paralelamente, el enfoque de Experiencia del Usuario (UX) ha comenzado a incorporarse en el ámbito educativo. Por ejemplo, Castillo-Vergara [18], introduce de manera lúdica el lienzo de problemas que busca conectar estudiantes con problemas reales de emprendimiento, como consecuencia, un buen resultado debería proponer que las actividades de aprendizaje deben diseñarse considerando las necesidades cognitivas, emocionales y motivacionales de los estudiantes [18][19]. Desde esta perspectiva, una experiencia educativa efectiva no depende únicamente del contenido, sino también de la forma en que los estudiantes interactúan con herramientas, recursos y actividades durante el proceso formativo.

En este contexto, metodologías de modelamiento como BPMN (Business Process Model and Notation) ofrecen una alternativa para representar formalmente las interacciones entre docente, estudiante y tecnología [20][21]. BPMN permite visualizar secuencias de actividades, puntos de decisión y flujos de información, facilitando el diseño, análisis y mejora de experiencias educativas complejas. Aunque BPMN ha sido ampliamente utilizado en ingeniería y gestión de procesos, su aplicación en educación aún es limitada, especialmente en el diseño de experiencias de aprendizaje mediadas por programación.

Por ello, surge la necesidad de metodologías integradas que permitan a los profesores aprender programación y, simultáneamente, transformar contenidos matemáticos en experiencias de aprendizaje gamificadas, modeladas y evaluables. La propuesta desarrollada en esta investigación busca responder a esta necesidad mediante una metodología estructurada en tres dimensiones y seis etapas, orientada a facilitar la creación sistemática de experiencias educativas basadas en programación, pensamiento computacional y diseño centrado en el estudiante.

III. Metodología

Esta investigación adopta un enfoque metodológico mixto y exploratorio, fundamentado en principios del diseño centrado en el usuario y del *Learning Experience Design* (LXD), que promueve la creación de experiencias educativas considerando las necesidades, motivaciones y características de los estudiantes [18][22]. Asimismo, incorpora la evaluación de carga de trabajo mediante el instrumento NASA-TLX [23].

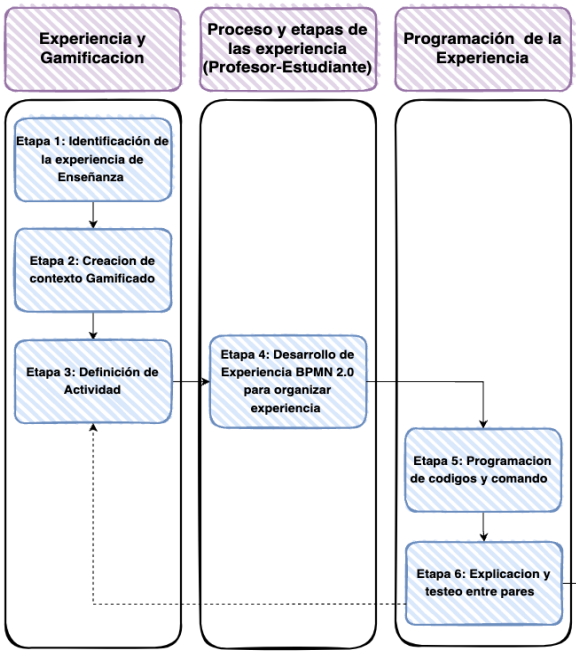


Fig 1: Metodología propuesta

Dimensión 1: Experiencia y Gamificación

Etapa 1: Identificación de la experiencia de enseñanza

La metodología comienza con la identificación de necesidades pedagógicas reales surgidas desde la práctica docente. A través de observaciones de aula, entrevistas y análisis colaborativos, se detectan contenidos complejos, abstractos o con baja motivación estudiantil. Esta etapa busca comprender la experiencia actual del estudiante y definir los desafíos educativos que requieren nuevas formas de representación y aprendizaje. El proceso se fundamenta en enfoques de diseño centrado en el usuario (UX) y en la comprensión de las necesidades emocionales, cognitivas y sensoriales de los estudiantes [11][24][25].

Etapa 2: Creación del contexto gamificado

Una vez identificado el desafío educativo, se diseña una experiencia de aprendizaje basada en principios de ludificación y teoría metamotivacional. Se incorporan elementos propios del juego, tales como exploración, manipulación, descubrimiento, desafíos y participación activa, con el propósito de inducir estados paratélicos asociados al disfrute, la curiosidad y la creatividad [17][26]. Asimismo, la experiencia se diseña considerando mecanismos que favorezcan emociones positivas y compromiso durante la interacción con la actividad [29]. El diseño de la experiencia considera además principios de experiencia del usuario (UX), buscando generar entornos emocionalmente atractivos y significativos para el estudiante [18][22].

Etapa 3: Definición de la actividad educativa

En esta etapa se estructura la actividad de aprendizaje definiendo objetivos, contenidos, interacciones esperadas y resultados de aprendizaje. Se establecen los mecanismos mediante los cuales los estudiantes interactuarán con los recursos didácticos y se especifican los elementos de gamificación asociados a la experiencia. Esta fase permite transformar una necesidad educativa en una propuesta concreta de intervención pedagógica alineada con los objetivos curriculares y motivacionales [18][27][34].

Dimensión 2: Proceso y etapas de la experiencia (Profesor–Estudiante)

Etapa 4: Desarrollo de la experiencia mediante BPMN 2.0

La experiencia educativa se modela utilizando BPMN 2.0 como herramienta para representar la interacción entre profesor, estudiante y recursos didácticos. El proceso permite visualizar las actividades, decisiones, retroalimentaciones y puntos de interacción durante la experiencia de aprendizaje. BPMN facilita la organización sistemática de la secuencia pedagógica y permite asegurar coherencia entre los objetivos educativos y la experiencia diseñada. Asimismo, permite identificar oportunidades de mejora y estandarización para futuras implementaciones [20][21].

Durante esta etapa también se integran herramientas de diseño conceptual como los modelos mapa de Empatía y modelos ilustrativos, los cuales permiten estructurar la transformación de ideas docentes en soluciones didácticas concretas [27][35][36].

Dimensión 3: Programación de la Experiencia

Etapa 5: Programación de códigos y comandos

Una vez validado el flujo educativo, se desarrollan los recursos físicos o digitales que soportarán la experiencia. En el presente estudio esta etapa corresponde al diseño de programas educativos mediante herramientas de programación y modelamiento computacional tales como Scratch, GeoGebra, R y Microsoft Excel. El desarrollo contempla criterios de usabilidad, accesibilidad, seguridad y experiencia de usuario, asegurando que la herramienta sea funcional y atractiva para los estudiantes [11].

Etapa 6: Explicación, aplicación y testeo entre pares

Finalmente, la experiencia es aplicada y evaluada entre pares con el propósito de analizar su funcionamiento y la experiencia percibida durante el desarrollo de las actividades. En esta investigación se utilizó el instrumento NASA-TLX para evaluar la carga de trabajo percibida asociada a las tres dimensiones de la metodología: Gamificación, Modelamiento mediante BPMN 2.0 y Programación [23][28]. Los resultados fueron analizados mediante estadística descriptiva, considerando las dimensiones de esfuerzo, demanda mental, demanda física, demanda temporal, rendimiento percibido y frustración. Esta evaluación permite caracterizar la carga de trabajo percibida durante la aplicación de la metodología e

identificar oportunidades de mejora para futuras iteraciones. Los resultados obtenidos mediante NASA-TLX se interpretan exclusivamente en términos de carga de trabajo percibida y no como una medida directa del aprendizaje alcanzado por los participantes.

IV. Caso de estudios y Resultados

En esta sección se presentan dos casos de estudios de la aplicación metodológica utilizada en este artículo, los casos de estudios se describen a continuación.

Caso de Estudio 1: Aplicación de una Experiencia de Programación Lúdica en Matemática para el Aprendizaje de Vectores en 7° Básico

Dimensión 1: Experiencia y Gamificación

Etapa 1: Identificación de la experiencia de enseñanza

La experiencia de enseñanza se orienta al aprendizaje de vectores en el plano cartesiano para estudiantes de 7° básico, considerando el Objetivo de Aprendizaje OA_14 del currículum nacional chileno [30]: identificar puntos en el plano cartesiano usando pares ordenados y vectores de forma concreta y pictórica. A partir de este objetivo, se define una actividad centrada en el reconocimiento de movimientos horizontales y verticales, el uso de vectores y la comparación de distancias mediante circunferencias.

Etapa 2: Creación del contexto gamificado

La experiencia se presenta mediante una narrativa lúdica titulada "Misión completar el álbum del Mundial de la FIFA 2026". En esta misión, los estudiantes deben ayudar a Carlos, un niño fanático del fútbol, a recorrer la ciudad para conseguir láminas faltantes de su álbum. El estudiante asume un rol activo como estratega de rutas, tomando decisiones sobre recorridos hacia la tienda Panini, la casa del primo y la casa de los abuelos.



Fig 2: Applet de GeoGebra con el mapa de la ciudad y los puntos de recorrido.

Etapa 3: Definición de la actividad educativa

La actividad se estructura en tres preguntas progresivas. Primero, el estudiante realiza un recorrido guiado usando deslizadores horizontales y verticales. Luego, debe determinar los movimientos necesarios para llegar a la casa del primo. Finalmente, debe ayudar a Carlos a regresar a su casa y comparar distancias entre distintos destinos usando vectores y circunferencias.

Dimensión 2: Proceso y etapas de la experiencia Profesor–Estudiante

Etapa 4: Desarrollo de la experiencia mediante BPMN 2.0

El proceso de enseñanza-aprendizaje se modela mediante un diagrama BPMN 2.0, representando la interacción entre docente, estudiante y software GeoGebra. El diagrama permite organizar la secuencia de instrucciones, activación de casillas, uso de deslizadores, construcción de puntos, representación de vectores y verificación de distancias. Esta representación facilita comprender el flujo de la clase y asegurar que cada acción del estudiante tenga una respuesta dentro del software.

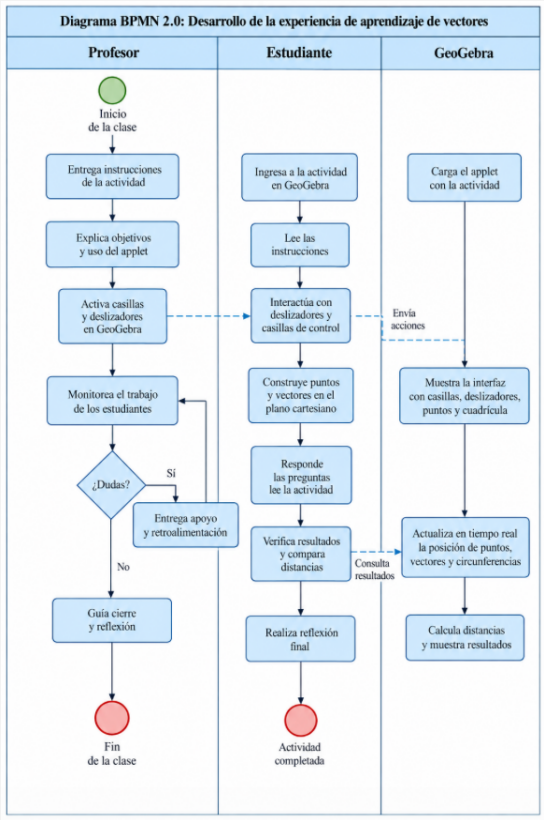


Fig 3: imagen referencial de diagrama BPMN 2.0 de la actividad planteada.

Dimensión 3: Programación de la Experiencia

Etapa 5: Programación de códigos y comandos

La programación de la experiencia se desarrolla mediante GeoGebra, utilizando herramientas como casillas de control, deslizadores, puntos, vectores y circunferencias. Estas herramientas permiten representar dinámicamente los movimientos de Carlos en el plano cartesiano, modificar parámetros, construir rutas y comparar módulos de vectores. La guía de comandos permite que los estudiantes comprendan cómo interactuar con el software durante la actividad.

Etapa 6: Explicación, aplicación y testeo entre pares

La experiencia finaliza con la aplicación de la actividad y su evaluación mediante una pauta de desempeño. La evaluación considera seis indicadores: reconocimiento de movimientos horizontales y verticales, uso de deslizadores, representación de vectores, formulación de conjeturas, respuesta a preguntas contextualizadas y comparación de módulos de vectores. Además, se promueve una reflexión profesional sobre el uso de BPMN, la ludificación y GeoGebra como herramientas para fortalecer el aprendizaje matemático.

Caso de Estudio 2: Aplicación de la Metodología Propuesta al Proyecto "El Algoritmo Secreto de Stream-Craft"

El proyecto "El Algoritmo Secreto de Stream-Craft" tiene como propósito que estudiantes de 1° Medio descubran empíricamente una función afín utilizando Scratch, mediante una experiencia gamificada basada en programación visual y pensamiento computacional.

Dimensión 1: Experiencia y Gamificación

Etapa 1: Identificación de la experiencia de enseñanza

Se identifica una dificultad frecuente en el aprendizaje del álgebra: los estudiantes suelen aprender la función afín como una fórmula abstracta sin comprender el significado de la pendiente y el intercepto. Por ello, la experiencia busca que los estudiantes construyan el modelo matemático a partir de evidencia empírica obtenida mediante interacción con un simulador Scratch. El objetivo es pasar desde la observación de datos hacia la construcción de la expresión algebraica general de la forma:

$$y=mx+b$$

En este caso específico:

$$y=3x+2$$

donde:

- x = Minutos de transmisión.
- y = Gemas de Poder obtenidas.

Etapa 2: Creación del contexto gamificado

Se construye una narrativa basada en una plataforma ficticia llamada Stream-Craft. Los estudiantes asumen el rol de:

- Auditores de software.
- Detectives digitales.
- Hackers éticos.

La misión consiste en descubrir el algoritmo oculto que calcula las recompensas entregadas por la plataforma a los streamers.

La actividad incorpora:

- Historia contextual.
- Misiones progresivas.
- Trabajo colaborativo.
- Recompensas finales.
- Insignia de "Ingeniero de Software Senior".

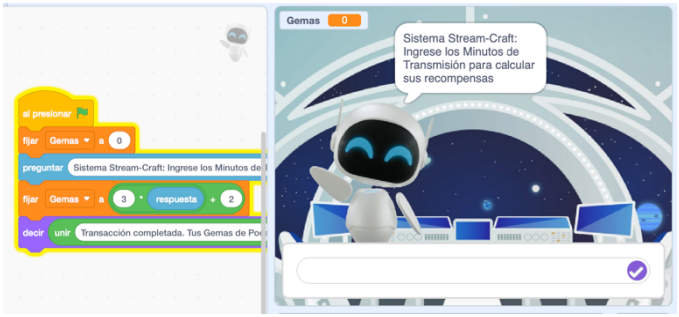


Fig 4. Interfaz de Stream-Craft desarrollada en Scratch

Etapa 3: Definición de la actividad educativa

La actividad se organiza en tres niveles de experiencias, lo cual hace que la experiencia de enseñanza pueda ser en varias sesiones de estudios.

Nivel 1: Exploración

Los estudiantes ingresan valores pequeños y registran los resultados obtenidos en una bitácora.

TABLA I: Ejemplo de datos ingresados acorde la experiencia.

Minutos	Gemas
1	5
2	8
3	11

Esta bitácora ayuda a ver la relación existente, sin conocer originalmente la relación basada en la función.

Nivel 2: Descubrimiento

Los estudiantes prueban valores extremos y detectan:

- incremento constante de 3 unidades.
- valor inicial de 2 gemas.

Formulan la hipótesis:

$$y=3x+2$$

El objetivo es identificar la función que relaciona sus valores con el resultado.

Nivel 3: Predicción

Los estudiantes deben calcular manualmente el valor para: x=30, antes de utilizar Scratch para verificar la respuesta, con esto consolida la hipótesis de búsqueda de la función.

Dimensión 2: Proceso y etapas de la experiencia Profesor–Estudiante

Etapa 4: Desarrollo de la experiencia mediante BPMN 2.0

El proceso es modelado mediante BPMN para coordinar las acciones de docente, estudiante, software, que se presentan en la figura 5 a continuación, el objetivo del BPMN es proporcionar una representación clara y sistemática de cómo se llevará a cabo la enseñanza.

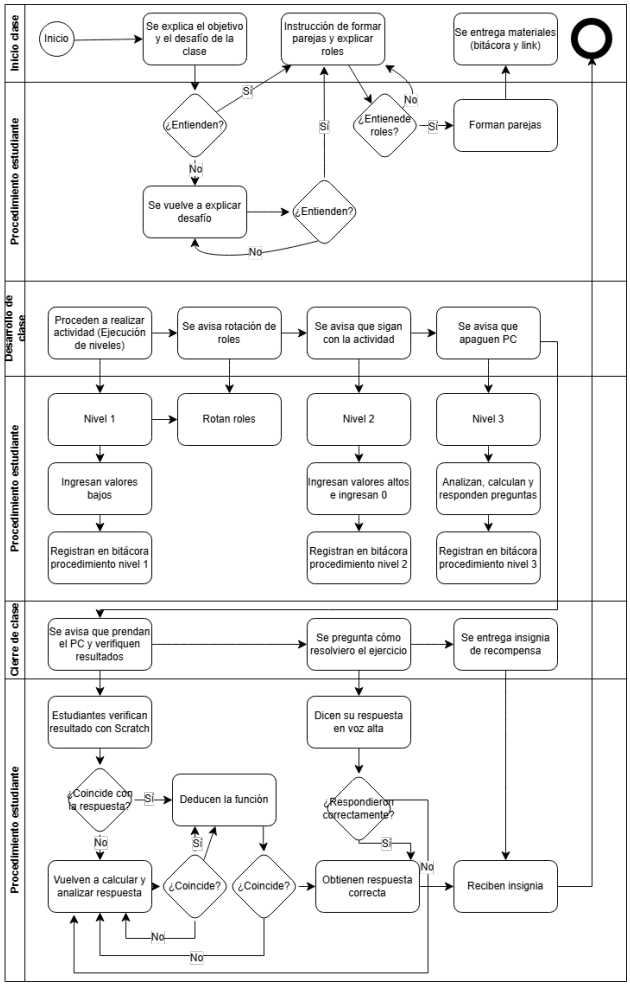


Fig 5: Representación de BPMN de la experiencia educativa.

Finalmente, el BPMN permite representar las decisiones pedagógicas y los puntos de interacción entre actores.

Dimensión 3: Programación de la Experiencia
Etapa 5: Programación de códigos y comandos

La experiencia utiliza Scratch como herramienta principal.
Los estudiantes analizan los comandos que conforman el algoritmo:

Tabla II: Ejemplo de algoritmos de programación

Comando	Función
Bandera Verde	Inicio del sistema
Preguntar y esperar	Entrada de datos
Variable Gemas	Almacenar resultados
Operador 3×x+2	Cálculo matemático
Decir	Mostrar resultado

El código permite automatizar el cálculo de la función afín y transformar una expresión algebraica en un sistema interactivo observable por los estudiantes.

Etapa 6: Explicación, aplicación y testeo entre pares

Finalmente, la experiencia es aplicada en modalidad Pair Programming.
Los estudiantes:

- alternan roles de conductor y navegador;
- registran evidencia en una bitácora;
- contrastan sus hipótesis;
- explican la función descubierta frente al curso.

Los dos casos de estudio tienen como propósito ilustrar la aplicación de las seis etapas de la metodología en diferentes contenidos matemáticos y plataformas de programación. En consecuencia, los resultados de aprendizaje descritos corresponden a resultados esperados del diseño de las experiencias y no a mediciones empíricas del aprendizaje alcanzado por estudiantes escolares.

V. RESULTADOS NASA-TLX

Se utilizó el instrumento NASA-TLX con un grupo de estudiantes de Pedagogía en Matemáticas (n = 13) para evaluar la carga de trabajo percibida durante las tres dimensiones de la metodología propuesta. Los resultados muestran una carga de trabajo moderada a lo largo de la experiencia, caracterizada por demandas mentales y temporales medias (4,77–5,62 y 5,00–5,31, respectivamente), bajos niveles de demanda física (2,38–3,15) y frustración (2,15–2,85), junto con una alta percepción de rendimiento (7,15–8,46) en una escala de 1 a 10 (Tabla III). En conjunto, estos resultados sugieren una distribución relativamente estable de la carga de trabajo percibida entre las tres dimensiones, sin observarse incrementos importantes a medida que avanza la secuencia metodológica. Estos resultados permiten caracterizar la experiencia de aplicación de la metodología, pero no constituyen una medida directa del aprendizaje alcanzado por los participantes [31][32].

Tabla III NASA-TLX en estudiantes

Dimensión NASA-TLX	Gamificación	BPMN 2.0	Programación
Esfuerzo	4.69	4.08	4.15
Demanda Mental	5.62	4.77	5.08
Demanda Física	2.46	2.38	3.15
Demanda Temporal	5.00	5.08	5.31
Rendimiento	7.69	7.15	8.46
Frustración	2.15	2.62	2.85

En la dimensión de Gamificación se observa la mayor demanda mental (5,62), acompañada de un esfuerzo moderado (4,69). Este resultado podría estar asociado a que corresponde a la dimensión inicial de la metodología, donde los participantes deben comprender la dinámica de trabajo, identificar el problema y adaptarse a una nueva forma de desarrollar la actividad. Sin embargo, esta dimensión presenta también el menor nivel de frustración (2,15), mostrando que la mayor demanda mental no se encuentra acompañada por un incremento equivalente en la frustración percibida.

La dimensión de Modelamiento mediante BPMN 2.0 presenta el promedio general de carga de trabajo más bajo (4,35), aunque mantiene una demanda temporal moderada (5,08). Este comportamiento podría estar asociado al tiempo destinado al análisis y organización de las actividades antes de su implementación, sin observarse incrementos relevantes en la demanda mental, el esfuerzo o la frustración. De esta manera, BPMN 2.0 presenta una carga de trabajo percibida relativamente equilibrada dentro de la secuencia metodológica.

Finalmente, la dimensión de Programación alcanza el mayor rendimiento percibido (8,46), junto con la mayor demanda temporal (5,31). Esto indica que, aunque esta dimensión requiere una mayor dedicación temporal percibida, los participantes reportan también el mayor nivel de rendimiento. La demanda mental (5,08) se mantiene próxima a los valores observados en las otras dimensiones y por debajo de Gamificación (5,62). En conjunto, los resultados muestran una distribución relativamente estable de la carga de trabajo percibida entre las tres dimensiones,

sin observarse incrementos importantes a medida que avanza la secuencia metodológica.

VI. DISCUSIÓN

Los resultados obtenidos sugieren que la metodología propuesta proporciona una estructura sistemática para incorporar la programación en la enseñanza de la matemática mediante tres dimensiones: gamificación, modelamiento del proceso educativo mediante BPMN 2.0 y programación. Más que centrarse en herramientas específicas como Scratch o GeoGebra, la metodología organiza el diseño de la experiencia desde la identificación de una necesidad pedagógica hasta su transformación en una actividad programable. Los casos de vectores y funciones afines ilustran que esta estructura puede aplicarse a diferentes contenidos matemáticos y plataformas, manteniendo una secuencia metodológica común.

Esta aproximación es consistente con investigaciones que destacan el potencial del pensamiento computacional para favorecer el compromiso estudiantil, la comprensión conceptual y habilidades como la colaboración y la perseverancia [7]. Asimismo, responde a una de las principales barreras identificadas para su incorporación en contextos educativos: la falta de procedimientos que orienten a los docentes en el diseño e implementación de experiencias basadas en programación [9]. En este sentido, la integración de diseño pedagógico, BPMN 2.0 y programación constituye el principal aporte de la propuesta, al proporcionar una secuencia que permite estructurar progresivamente la experiencia educativa.

La evaluación mediante NASA-TLX aporta información complementaria sobre la carga de trabajo percibida durante la aplicación de la metodología. Los resultados muestran una carga relativamente estable entre las tres dimensiones, con demandas mentales y temporales moderadas, bajos niveles de frustración y una alta percepción de rendimiento. La Gamificación presenta la mayor demanda mental (5,62) y el menor nivel de frustración (2,15), mientras que la Programación alcanza el mayor rendimiento percibido (8,46) y la mayor demanda temporal (5,31). Estos resultados sugieren que la progresión entre las dimensiones no produce incrementos importantes en la carga de trabajo percibida. Sin embargo, NASA-TLX evalúa la percepción de carga de trabajo y, por lo tanto, estos resultados no deben interpretarse como evidencia directa de aprendizaje o efectividad pedagógica.

Los casos de estudio permiten ilustrar cómo las seis etapas pueden utilizarse para transformar contenidos matemáticos en experiencias gamificadas, modeladas y programables. En consecuencia, los resultados de aprendizaje planteados en estos casos corresponden a resultados esperados del diseño de las experiencias y no a mediciones empíricas del aprendizaje alcanzado por estudiantes escolares. Esta distinción permite separar la aplicación de la metodología de su futura validación en términos de resultados de aprendizaje.

Finalmente, los resultados deben interpretarse considerando el carácter exploratorio del estudio. La evaluación mediante NASA-TLX se realizó con una muestra limitada de estudiantes de Pedagogía en Matemáticas (n=13) y sin un grupo de control, mientras que los casos se concentran en dos contenidos de matemática y dos plataformas de programación. Estas condiciones limitan la generalización de los resultados. Futuras investigaciones deberían ampliar el número y diversidad de participantes, incorporar otros niveles educativos y disciplinas STEM, y evaluar indicadores específicos de aprendizaje mediante diseños comparativos que permitan ampliar la evidencia empírica sobre la metodología.

.

VII. CONCLUSIÓN

La presente investigación propone una metodología para apoyar el diseño de experiencias educativas basadas en programación en la enseñanza de la matemática. La propuesta integra tres dimensiones —gamificación, modelamiento mediante BPMN 2.0 y programación— organizadas en seis etapas que orientan al docente desde la identificación de una necesidad pedagógica hasta el desarrollo de una actividad programable. Los casos de estudio ilustran que esta estructura puede aplicarse a diferentes contenidos matemáticos y plataformas, manteniendo una secuencia metodológica común.

La evaluación mediante NASA-TLX muestra una carga de trabajo percibida relativamente estable, con demandas mentales y temporales moderadas, bajos niveles de frustración y una alta percepción de rendimiento. Estos resultados permiten caracterizar la experiencia de aplicación de la metodología, pero no constituyen una medida directa del aprendizaje alcanzado por los participantes.

La principal contribución del trabajo corresponde a la integración del diseño pedagógico, el modelamiento de procesos y el pensamiento computacional dentro de una misma secuencia, proporcionando un procedimiento que puede apoyar a los docentes en el diseño de experiencias

basadas en programación. Como trabajo futuro, se propone ampliar su aplicación a un mayor número de participantes, otros niveles educativos y disciplinas STEM, incorporando indicadores específicos de aprendizaje que permitan fortalecer su validación empírica.

REFERENCIAS

[1] Nikulin, C., Zuniga, M., Akhloufi, M., Manzi, C., Wiche, C., & Pinones, E. (2018). Enhancing creativity for development of automation solutions using OTSM-TRIZ: A systematic case study in agronomic industry. *Advances in Mechanical Engineering*, *10*(1), 1687814017751950.

[2] Muslim, N. E. I., Zakaria, M. I., & Fang, C. Y. (2023). *A Systematic Review of GeoGebra in Mathematics Education*. International Journal of Academic Research in Progressive Education and Development, 12(3), 1192–1203. https://doi.org/10.6007/IJARPED/v12-i3/19133.

[3] Rais, D., & Zhao, X. (2023). *Human Cognitive: Learning Mathematics through Python Programming to Support Students' Problem-Solving Skills*. *Anatolian Journal of Education, 8*(2), 85–98. https://doi.org/10.29333/aje.2023.826a

[4] Rais, D., & Zhao, X. (2024). *Elevating student engagement and academic performance: A quantitative analysis of Python programming integration in the Merdeka Belajar curriculum*. *Journal on Mathematics Education, 15*(2), 495–516. https://doi.org/10.22342/jme.v15i2.pp495-516.

[5] J. M. Wing, "Computational thinking," Communications of the ACM, vol. 49, no. 3, pp. 33–35, 2006, doi: 10.1145/1118178.1118215.

[6] Kallia, M., van Borkulo, S. P., Drijvers, P., Barendsen, E., & Tolboom, J. (2021). *Characterising computational thinking in mathematics education: A literature-informed Delphi study*. Research in Mathematics Education, 23(2), 159–187. https://doi.org/10.1080/14794802.2020.1852104

[7] Vázquez-Uscanga, E., Nussbaum, M, & Naranjo, I. (2025*). Integrating unplugged computational thinking across curricula: A qualitative study of students' and teachers' perspectives.* International Journal of Instruction, 18(1), 357-378. https://doi.org/10.29333/iji.2025.18120a

[8] Yadav, A., Hong, H., & Stephenson, C. (2016). *Computational thinking for all: Pedagogical approaches to embedding 21st century problem solving in K-12 classrooms*. TechTrends, 60(6), 565–568. https://doi.org/10.1007/s11528-016-0087-7

[9] Liu, Z., Gearty, Z., Richard, E., Orrill, C. H., Kayumova, S., & Balasubramanian, R. (2024). *Bringing computational thinking into classrooms: A systematic review on supporting teachers in integrating computational thinking into K-12 classrooms*. International Journal of STEM Education, 11, 51. https://doi.org/10.1186/s40594-024-00510-6

[10] Deterding, S., Dixon, D., Khaled, R., & Nacke, L. (2011). From game design elements to gamefulness: Defining gamification. In Proceedings of the 15th International Academic MindTrek Conference (pp. 9–15). ACM. https://doi.org/10.1145/2181037.2181040

[11] Deci, E. L., & Ryan, R. M. (2000). The "what" and "why" of goal pursuits: Human needs and the self-determination of behavior. Psychological Inquiry, 11(4), 227–268. https://doi.org/10.1207/S15327965PLI1104_01

[12] Norman, D. A. (2013). The design of everyday things (Revised ed.). Basic Books.

[13] Rodríguez-Gallego, M. R., & Ordóñez-Sierra, R. (2015). Una experiencia de aprendizaje-servicio en comunidades de aprendizaje.

[14] Bruner, J. S. (1997). La educación, puerta de la cultura. Gedisa.

[15] Reyes, C. Q. (2020). Pandemia Covid-19 e Inequidad Territorial: El agravamiento de las desigualdades educativas en Chile. *Revista internacional de educación para la justicia social*, *9*(3), 3-5.

[16] Sawyer, R. K. (2012). Explaining creativity: The science of human innovation. Oxford University Press.

[17] Apter, M. J. (2001). Motivational styles in everyday life: A guide to reversal theory. American Psychological Association.

[18] Castillo-Vergara M., Lienzo de Problemas. Innocore Toolkit, Zenodo, 2024. doi: 10.5281/zenodo.13921837.

[19] Rogers, C., & Schnepp, J. (2024). Implementing Learner Experience Design in University Teaching: An Action Research Study on Enhancing Faculty-Student Engagement and Motivation. Journal of the Scholarship of Teaching and Learning, 24(4), 194–206. https://doi.org/10.14434/josotl.v24i4.35977

[20] Object Management Group. (2011). Business Process Model and Notation (BPMN) Version 2.0. Object Management Group. Recuperado de https://www.omg.org/spec/BPMN/2.0

[21] Dumas, M., La Rosa, M., Mendling, J., & Reijers, H. A. (2018). *Fundamentals of Business Process Management* (2nd ed.). Springer. https://doi.org/10.1007/978-3-662-56509-4

[22] Camere, S., & Bordegoni, M. (2015). The Experience Map as a Tool for Design Education. The Design Journal, 18(2), 291–307. https://doi.org/10.2752/175630615X14212498964385

[23] Hart, S. G. (2006). NASA-Task Load Index (NASA-TLX): 20 years later. Proceedings of the Human Factors and Ergonomics Society Annual Meeting, 50(9), 904–908.

[24] Pucillo, F., & Cascini, G. (2014). A framework for user experience, needs and affordances. *Design studies*, *35*(2), 160-179.

[25] Hatchuel, A., & Weil, B. (2009). C-K design theory: An advanced formulation. Research in Engineering Design, 19(4), 181–192. https://doi.org/10.1007/s00163-008-0043-4

[26] Keller, J., & Bless, H. (2008). Flow and regulatory compatibility: An experimental approach to the flow model of intrinsic motivation. Personality and Social Psychology Bulletin, 34(2), 196–209. https://doi.org/10.1177/0146167207310026

[27] Nikulin Chandia, C., Viveros Gunckel, P., Dorochesi Fernandois, M., Crespo Márquez, A., & Lay Bobadilla, P. (2017). Metodología para el análisis de problemas y limitaciones en emprendimientos universitarios. *Innovar*, *27*(63), 91-105.

[28] Nikulin, C., Lopez, G., Piñonez, E., Gonzalez, L., & Zapata, P. (2019). NASA-TLX for predictability and measurability of instructional design models: case study in design methods. *Educational Technology Research and Development*, *67*, 467-493

[29] Lazzaro, N. (2009). Why we play games: Four keys to more emotion without story. XEODesign Inc. [White Paper].

[30] Ministerio de Educación de Chile. (s.f.). Objetivo de aprendizaje MA07 OA 14: Identificar puntos en el plano cartesiano, usando pares ordenados y vectores de forma concreta (juegos) y pictórica. Currículum Nacional. Recuperado de Currículum Nacional – MA07 OA 14

[31] Chen, O., Kalyuga, S., & Sweller, J. (2023). *A Cognitive Load Theory Approach to Defining and Measuring Task Complexity Through Element Interactivity*. *Educational Psychology Review, 35*, 63. https://doi.org/10.1007/s10648-023-09782-w.

[32] Hutson, J. W., Franklin, A. E., Rogers, B. A., & Walker, D. (2024). *Psychometric Testing of NASA-TLX to Measure Learners' Cognitive Load in Individual and Group Nursing Simulations.* Clinical Simulation in Nursing, 95, 101607.

[33] Verganti, R. (2009). Design-driven innovation: Changing the rules of competition by radically innovating what things mean. Harvard Business Press.

[34] Schunk, D. H., Pintrich, P. R., & Meece, J. L. (2014). Motivación en el aula: Teoría, investigación y práctica (3ª ed.). Pearson Educación.

[35] Vera, P., Nikulin, C., Lopez-Campos, M., & Gonzalez Ramirez, R. G. G. (2019). Prospective study using archetypes and system dynamics. *Academia Revista Latinoamericana de Administración*, *32*(2), 181-202.

[36] Becattini, N., Cascini, G., & Nikulin, C. (2015). Modelling the dynamics of products and processes requirements. *Procedia Engineering*, *131*, 661-671.