



*Asesorías y Tutorías para la Investigación Científica en la Educación Puig-Salabarría S.C.
José María Pino Suárez 400-2 esq a Lerdo de Tejada, Toluca, Estado de México. 7223898475*

RFC: ATI120618V12

Revista Dilemas Contemporáneos: Educación, Política y Valores.

<http://www.dilemascontemporaneoseduccionpoliticayvalores.com/>

Año: XII

Número: Edición Especial

Artículo no.:23

Período: Diciembre del 2024

TÍTULO: Evaluación de aplicación web para la creación de juegos que fomentan el aprendizaje de la lógica de programación.

AUTORES:

1. Dr. Carlos Roberto Jaimez-González.
2. Máster. Betzabet García-Mendoza.
3. Lic. Javier Erazo-Palacios.

RESUMEN: El artículo presenta la evaluación de una aplicación web para crear juegos que fomentan el aprendizaje de la lógica de programación. El objetivo de los juegos creados es mover un personaje por un tablero utilizando programación basada en bloques para recolectar todos los objetos que se colocan en el tablero. La aplicación web fue diseñada para dos tipos de usuarios: profesores, quienes pueden crear juegos, especificando la distribución de los tableros; y alumnos, quienes juegan los juegos, especificando la secuencia de comandos para recolectar todos los objetos del tablero. Para la evaluación de la aplicación web se diseñaron instrumentos de evaluación para profesores y alumnos, tomándose en cuenta tres aspectos: el diseño estético, la usabilidad y la funcionalidad.

PALABRAS CLAVES: aprendizaje de la programación, instrumento de evaluación, aplicación web, programación basada en bloques, lógica de programación.

TITLE: Evaluation of a web application for the creation of games that promote the learning of programming logic.

AUTHORS:

1. PhD. Carlos Roberto Jaimez-González.
2. Master. Betzabet García-Mendoza.
3. Bach. Javier Erazo-Palacios.

ABSTRACT: This article presents the evaluation of a web application for creating games that encourage the learning of programming logic. The objective of the games created is to move a character around a board using block-based programming to collect all the objects placed on the board. The web application was designed for two types of users: teachers, who can create games, specifying the layout of the boards; and students, who play the games, specifying the script to collect all the objects on the board. For the evaluation of the web application, evaluation instruments were designed for teachers and students, taking into account three aspects: aesthetic design, usability and functionality.

KEY WORDS: learning of programming, evaluation instrument, web application, block-based programming, programming logic.

INTRODUCCIÓN.

La lógica de programación ha sido incluida en los planes de estudio de escuelas en diferentes niveles. Aunque algunos pueden cuestionar su utilidad, aprender programación enriquece la perspectiva del estudiante, aunque el conocimiento adquirido pueda no tener una aplicación práctica inmediata (Reichert *et al.*, 2001). Al observar los resultados del desempeño de los estudiantes, la forma en que se enseña la programación también juega un papel importante. Weintrop y Wilensky (2015) afirman que enseñar programación utilizando un lenguaje de programación visual comparado con un lenguaje de programación basado en texto tiene un impacto en cómo los estudiantes comprenden el tema.

El trabajo que se presenta en este artículo es la evaluación de una aplicación web que fue diseñada y desarrollada para crear juegos personalizados, con el objetivo de apoyar el aprendizaje de la lógica de programación de computadoras. En el resto del artículo se presentan la justificación y antecedentes de la

aplicación web; el estado del arte; el funcionamiento de la aplicación web; la evaluación de la aplicación web, incluyendo los instrumentos de evaluación desarrollados; y finalmente, las conclusiones y trabajo a futuro.

DESARROLLO.

En esta sección se proporcionan los antecedentes y la justificación de la aplicación web; se hace una revisión del estado del arte; se proporcionan detalles del funcionamiento de la aplicación web y se presenta la evaluación que se realizó a la aplicación web.

Antecedentes y justificación.

Es indiscutible que la tecnología tiene un impacto significativo en nuestras vidas y en nuestra educación. Según la Encuesta Nacional sobre Disponibilidad y Uso de Tecnologías de la Información en los Hogares (INEGI, 2019) en el año 2018, más de 50 millones de mexicanos tenían acceso a Internet, lo que constituía más del 50% de la población de seis años o más. Entre las actividades más comunes que realizaban estas personas estaba la búsqueda y adquisición de información en Internet, y más de 50 millones de mexicanos también utilizaban una computadora; por lo anterior, es crucial aprender a utilizar estas herramientas tecnológicas desde una edad temprana y aprovechar sus capacidades.

Yadin (2011) afirma, que aprender programación puede ser abrumador, especialmente si el método de enseñanza no es efectivo; no es raro que los estudiantes universitarios que toman cursos de programación tengan dificultades, y en última instancia, abandonen el curso, percibiendo la materia como difícil; sin embargo, aprender programación ofrece numerosas ventajas; un análisis de 65 estudios sobre los efectos cognitivos de la programación reveló que los estudiantes con experiencia en programación obtuvieron mejores puntajes en pruebas de habilidades cognitivas que aquellos sin dicha experiencia (Liao y Bright, 1991).

La incorporación de herramientas informáticas en la enseñanza en el aula produce diversos efectos en los estudiantes. En comparación con los métodos tradicionales, las herramientas informáticas animan a

los estudiantes a participar, aumentando el compromiso y haciendo que el proceso de aprendizaje sea más atractivo y dinámico. Además, ofrecen retroalimentación inmediata, lo que permite a los estudiantes seguir su progreso en tiempo real (Caccuri, 2013). El uso de lenguajes de programación visual, como la programación basada en bloques, se ha destacado en la enseñanza por sus beneficios sobre los lenguajes basados en texto. La simplicidad de los conceptos y la dependencia del reconocimiento visual en la programación basada en bloques hacen que sea más fácil de aprender que otros lenguajes basados en texto (Bau *et al.*, 2017).

Bain y Barnes (2014) realizaron un estudio preliminar con estudiantes universitarios en Escocia, donde se les pidió que nombraran un tema difícil y explicaran sus estrategias para abordarlo; el estudio reveló que la mitad de los estudiantes dependían únicamente de un motor de búsqueda para resolver sus problemas. Además, muchos de ellos no lograron comprender la conexión entre el tema principal y otros temas relacionados; esto indica que sus habilidades para resolver problemas son inadecuadas, ya que se centran solo en encontrar una solución rápida. Para evitar estas dificultades de aprendizaje en programación, una posible solución podría ser introducirla a los estudiantes a una edad temprana, utilizando una variedad de estrategias.

Se han realizado investigaciones a lo largo del tiempo para explorar la influencia y las consecuencias de enseñar programación a niños y jóvenes. Maloney *et al.* (2008) realizaron un estudio en personas de entre ocho y dieciocho años de edad que eran miembros de un club de computación, donde usaban el lenguaje de programación visual basado en bloques Scratch. El estudio informa que Scratch se hizo popular entre los jóvenes a medida que aprendían los conceptos básicos de programación de una manera entretenida mediante la creación de juegos y videos. El estudio demostró que la implementación de un enfoque nuevo y distintivo para la enseñanza de la programación puede tener resultados positivos.

Mladenović *et al.* (2017) llevaron a cabo otro estudio con 50 estudiantes de quinto grado de dos escuelas primarias que no tenían experiencia previa en programación y tenían diferentes niveles de habilidades

para la resolución de problemas. La investigación reveló hallazgos interesantes sobre la comprensión de los conceptos de programación, utilizando el lenguaje de programación Python y Scratch. Los estudiantes que tenían mejores habilidades para la resolución de problemas obtuvieron mejores resultados en las pruebas de Python en comparación con aquellos con una capacidad intermedia; sin embargo, cuando se trataba de Scratch, ambos grupos de estudiantes obtuvieron resultados similares.

En una investigación realizada en Croacia, Zaharija *et al.* (2013) propusieron un enfoque para enseñar conceptos de programación a niños de escuela primaria con dificultades de concentración y falta de estrategias de resolución de problemas; el enfoque utilizó un juego que implicaba navegar un robot a través de un laberinto para encontrar un premio; se utilizaron tarjetas de colores para simular las direcciones en las que el robot podía moverse. Este es un método de enseñanza que incorpora juegos, específicamente juegos de computadora, como un elemento crucial. Se ha evaluado comparativamente con los métodos de enseñanza tradicionales y se ha encontrado que mejora la motivación de los estudiantes, aunque no necesariamente da mejores resultados de aprendizaje (Egenfeldt-Nielsen, 2005).

Long (2007) realizó otro estudio sobre el juego *IBM Robocode*, un juego de programación que simula batallas entre robots y se juega en línea; el estudio reveló que más del 50% de los participantes, incluidos jóvenes y adultos, reportaron una mejora en sus habilidades de programación mientras jugaban al juego; una de las razones de esta mejora fue debido a que disfrutaban jugando. En otro estudio, Rogozhkina y Kushnirenko (2011) se centraron en un juego educativo llamado *PiktoMir*, diseñado para presentar conceptos básicos de programación a niños en edad preescolar de entre cinco y siete años. El estudio concluye, que si bien algunos niños no disfrutaron de las tareas de *PiktoMir*, la mitad de los niños querían seguir jugando, porque lo encontraban divertido.

Varios otros estudios e iniciativas han explorado enfoques similares. Una iniciativa de Keller y John (2020) fue un taller destinado a aumentar el número de mujeres en informática en el nivel medio y secundario; el taller introdujo conceptos de programación para aplicaciones robóticas mediante una

presentación con Python. La evaluación inicial del taller mostró que motivó a los estudiantes a aprender conceptos de programación. En otro estudio, Steinmaurer *et al.* (2019) utilizaron *sCool*, un juego que enseña a los niños a pensar computacionalmente y a codificar en Python. En el experimento, los estudiantes aprendieron un concepto y tuvieron que aplicarlo a una tarea práctica; dos grupos de estudiantes participaron en la actividad y el estudio mostró que los estudiantes estaban interesados en aprender a codificar, pero tenían dificultades para transferir el contenido aprendido a áreas similares. Otro estudio de Fatourou *et al.* (2018) tuvo como objetivo explorar los posibles beneficios de introducir conceptos de programación concurrente en las primeras etapas del proceso de aprendizaje. Existen numerosos enfoques que se pueden utilizar para aprender programación, pero uno de gran importancia es el factor visual.

Estado del arte.

En esta sección se examinan cinco herramientas distintas que tienen como objetivo enseñar conceptos de programación a niños y jóvenes. Las herramientas utilizan programación basada en bloques o en texto.

Code Combat (2020) es un juego en línea diseñado para que los estudiantes mayores de nueve años aprendan conceptos de programación en Python y JavaScript. El juego presenta diferentes niveles y misiones, en las que se especifican los conceptos de programación que se deben aprender. Code Combat ha tenido un impacto significativo en las escuelas, como lo demuestra la experiencia del profesor Baily (2020), quien creó una clase de codificación para que 180 estudiantes aprendieran conceptos como ciclos, funciones y lógica booleana.

Coding Adventure (2020) es un programa que enseña programación mediante el lenguaje *CoffeScript*, dirigido a niños de ocho años en adelante; forma parte de la plataforma de enseñanza *Code Monkey*. Este programa contiene un editor para escribir comandos para que un personaje los ejecute. Existe un estudio de caso (Fudale, 2016) sobre el impacto de *Coding Adventure*, el cual fue realizado por una

especialista en integración académica, quien buscaba una forma de enseñar a codificar a estudiantes de primaria; sus estudiantes desarrollaron habilidades de resolución de problemas y pensamiento crítico.

Lightbot (2020) es una aplicación de juego de rompecabezas dirigido a niños de cuatro años en adelante. El juego utiliza un lenguaje de programación basado en bloques y tiene un menú que se enfoca en enseñar conceptos de programación. Jana (2016) reporta su experiencia al utilizar *Lightbot* con sus estudiantes de quinto grado, quienes pudieron comprender conceptos básicos de programación como algoritmos y ciclos.

Code Karts (2020) es un juego de rompecabezas dirigido a niños de cuatro años en adelante, donde el objetivo es guiar un pequeño automóvil hasta la línea de meta utilizando instrucciones de bloques. La interfaz del juego consta de un cuadro en el lado izquierdo que contiene el conjunto de bloques de instrucciones que se pueden usar para lograr el objetivo y un cuadro en el lado superior donde se colocan los bloques para ejecutarlos.

Spritebox (2020) es un juego de aventuras apto para todas las edades, en el que los jugadores superan obstáculos codificando mediante programación basada en bloques o escribiendo comandos utilizando los lenguajes de programación *Swift* o *Java*. El juego tiene como objetivo enseñar conceptos de programación como secuenciación, parámetros y ciclos, a través de un personaje que resuelve obstáculos utilizando bloques.

Erazo-Palacios *et al.* (2022) realizaron un análisis comparativo de las herramientas examinadas en esta sección. Además, Pereira y Frango (2020) llevaron a cabo una comparación más amplia de los juegos educativos destinados a la enseñanza de la programación y la lógica computacional. Sosa, *et al.* (2017) también presentan una revisión de las tecnologías emergentes en el campo de la educación.

Funcionamiento de la aplicación web.

En esta sección se explican los módulos que componen a la aplicación web, los cuales se encargan de validar usuarios, mostrar mapas y niveles, ejecutar acciones para jugar juegos correspondientes a niveles

específicos, validar soluciones, generar animaciones, presentar resultados, crear mapas y niveles, entre otros. Jaimez-González *et al.* (2023) proporcionan una explicación detallada del funcionamiento de la aplicación web.

El *módulo de menú de alumnos* corresponde a la página web del menú principal para alumnos, donde pueden iniciar juegos y desbloquear mapas y niveles. La visualización de mapas, niveles y sus estados correspondientes son dinámicos.

El *módulo de niveles* permite a los estudiantes jugar un nivel específico del juego; contiene el panel de comandos con los comandos disponibles; el panel de ejecución, donde se colocan los comandos a ejecutar; el panel de función, donde se almacenan los comandos que se replicarán al utilizar un comando de función; el panel principal, que contiene el tablero del juego con varios objetos; entre otros.

El *módulo de verificación de soluciones* verifica una solución dada. Una vez que un estudiante determina su secuencia de bloques y la ejecuta, se crea una cadena que representa los comandos utilizados en el panel de ejecución, y otra que representa los comandos utilizados en el panel de funciones, para posteriormente compararlas.

El *módulo de animación del juego* genera la animación del juego. Para implementar los gráficos de los tableros de niveles se utilizaron *Sprites*, los cuales son imágenes bidimensionales que se integran a escenarios más grandes.

El *módulo de resultados del alumno* permite a los alumnos revisar el desempeño obtenido al completar los niveles. La interfaz de este módulo proporciona a los alumnos toda la información de su desempeño a través de una tabla de resultados.

El *módulo de menú del profesor* corresponde a la página web del menú principal del profesor, donde podrá ver todos los mapas y sus niveles correspondientes disponibles, así como la opción de visualizar los niveles individualmente.

El *módulo de visualización de niveles* permite a los docentes visualizar el tablero del nivel, así como algunos datos sobre el mismo, como el número de mapa al que pertenece el nivel, el número de nivel, el ancho y alto del tablero, el número total de casillas del tablero, y el número total de recursos.

El *módulo de creación de mapas* permite a los profesores crear un nuevo mapa y nuevos niveles para que los alumnos jueguen. Los profesores deben proporcionar el nombre y la categoría de dificultad del mapa. Una vez creado el mapa, el profesor puede añadir la cantidad de niveles que desee y editarlos como sea necesario.

El *módulo de creación y edición de niveles* permite a los profesores comenzar a crear y editar niveles generando un tablero, arrastrando y ordenando los objetos disponibles al tablero y guardándolo.

Evaluación de la aplicación web.

En esta sección se describe cómo se evaluó la aplicación web, la cual se llevó a cabo mediante pruebas de usuario. La evaluación se realizó a través de la observación y mediante un cuestionario, el cual se diseñó y aplicó para obtener la retroalimentación del usuario con respecto a ciertos aspectos que se consideraron importantes: diseño estético, usabilidad y funcionalidad. Para realizar las pruebas de usuario se diseñaron dos instrumentos de evaluación, uno para el usuario alumno y otro para el usuario profesor. Ambos instrumentos de evaluación se muestran más adelante.

Pruebas de usuario a alumnos.

Las pruebas se realizaron a 17 alumnos, a los cuales se les pidió resolver cinco distintos niveles de la aplicación web, los cuales aumentaban de dificultad de un nivel a otro. Los niveles se diseñaron procurando que los usuarios utilizaran todas las funcionalidades del sistema, para así poder detectar posibles errores o aspectos a mejorar.

A continuación, se muestran los niveles con los que se evaluaron a los usuarios. En la Figura 1 se muestran los primeros tres niveles que jugaron. En el primer nivel el conejito únicamente necesita avanzar para recoger la zanahoria; en el segundo nivel, el conejito debe darse vuelta, y posteriormente,

avanzar para recoger la zanahoria; y finalmente, en el tercer nivel, el conejito debe saltar un obstáculo para poder recoger la zanahoria. En estos tres niveles de prueba el tablero fue de 3x3.



Figura 1. Niveles 1, 2 y 3 de la aplicación web con tableros de 3x3.

En la Figura 2 se muestra el cuarto nivel que probaron los usuarios. En este caso es un nivel un poco más complejo, ya que se involucran más movimientos del conejito para recoger todas las zanahorias y evitar los obstáculos del tablero. El tablero fue de 4x4.

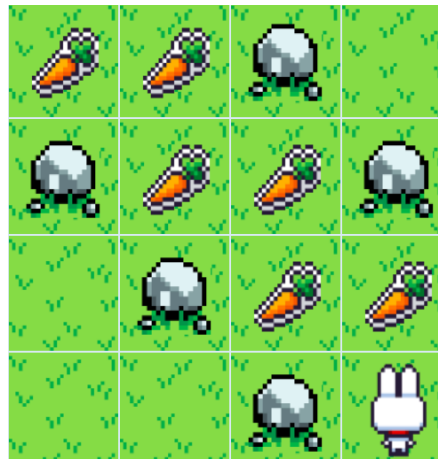


Figura 2. Nivel 4 de la aplicación web con un tablero de 4x4.

En la Figura 3 se muestra el quinto nivel que probaron los usuarios. Es un nivel más complejo que el anterior, ya que el tablero es más grande (6x6) y se involucraron más movimientos del conejito para recoger todas las zanahorias y evitar los obstáculos del tablero, que en este caso además de rocas, se incluyeron hoyos. Cabe señalar, que este nivel también puede ser resuelto por los usuarios utilizando ciclos, ya que existe un patrón que permite recoger las zanahorias con una secuencia repetitiva.



Figura 3. Nivel 5 de la aplicación web con un tablero de 6x6.

Durante la aplicación de las pruebas de usuario, se observó que los alumnos podían resolver los tres niveles iniciales sin problema alguno, mientras que los dos últimos niveles tardaron un poco más debido a su complejidad, ya que analizaban la distribución de objetos en el tablero para poder idear la secuencia de bloques de comandos correcta. A pesar de ello, lograron resolverlos correctamente, utilizando comandos más complejos como el del ciclo. Algunas funcionalidades de la interfaz de la aplicación web fueron utilizadas intuitivamente, como intercambiar de lugar dos bloques de comando o eliminarlos.

Pruebas de usuario a profesores.

Estas pruebas se realizaron a 10 profesores, a quienes se explicó el objetivo del juego, la funcionalidad de cada objeto del tablero, y se les pidió crear un mapa, así como crear y editar niveles utilizando los objetos disponibles del tablero.

A continuación, se muestran algunos niveles que fueron creados por los profesores, que probaron la aplicación web. En la Figura 4 se muestra un mapa de 6x6 creado por un profesor, en el cual se encuentran distribuidos en el tablero los diferentes objetos disponibles: zanahorias, rocas y hoyos. En

este caso el profesor utilizó 4 zanahorias, 7 rocas y 7 hoyos, y decidió colocar al conejito en la esquina inferior izquierda.

En la Figura 5 se muestra un mapa de 6x3 creado por otro profesor, en el cual se encuentran distribuidos en el tablero los diferentes objetos disponibles: zanahorias, rocas y hoyos. En este caso, el profesor utilizó 10 zanahorias, 2 rocas y 2 hoyos, y decidió colocar al conejito en la casilla correspondiente a la primera columna y segundo renglón.

Durante la aplicación de las pruebas de usuario, se observó que los profesores creaban un mapa fácilmente, y proporcionaban un nombre y una categoría de dificultad para poder crearlo. También tuvieron facilidad al editar los niveles ya creados, al arrastrar, soltar y eliminar intuitivamente los objetos del tablero, e inclusive formulaban la solución conforme iban editando el nivel.



Figura 4. Mapa de 6x6 creado por un profesor en la aplicación web.



Figura 5. Mapa de 6x3 creado por un profesor en la aplicación web.

Instrumento de evaluación para alumnos.

Este instrumento se dividió en tres secciones: diseño estético, usabilidad y funcionalidad. A continuación, se muestran las preguntas que se hicieron a los alumnos y los porcentajes que respondieron de cada una de las cinco opciones:

- a) Totalmente de acuerdo.
- b) Ligeramente de acuerdo.
- c) Ni de acuerdo ni en desacuerdo.
- d) Ligeramente en desacuerdo.
- e) Totalmente en desacuerdo.

En la Tabla 1 se muestran los resultados del instrumento de evaluación en cuanto al diseño estético; en la Tabla 2 se presentan los resultados del instrumento de evaluación de la sección de usabilidad; y finalmente, en la Tabla 3 se encuentran los resultados del instrumento de evaluación para la funcionalidad del sistema.

Tabla 1. Resultados del instrumento de evaluación en la sección de Diseño Estético.

Fuente: Elaboración propia con información de los resultados.

1. ¿Considera que es adecuada la organización de la información en la interfaz?				
a) 88%	b) 12%	c) 0%	d) 0%	e) 0%
2. ¿Considera que los botones utilizados en la interfaz son claros y fáciles de usar para navegar?				
a) 71%	b) 29%	c) 0%	d) 0%	e) 0%
3. ¿Considera que son adecuados los colores utilizados?				
a) 100%	b) 0%	c) 0%	d) 0%	e) 0%
4. ¿Considera que el tamaño y el tipo de letra utilizados le permiten leer sin dificultad los contenidos?				
a) 88%	b) 12%	c) 0%	d) 0%	e) 0%
5. ¿Considera que son adecuados los gráficos utilizados en los niveles?				
a) 94%	b) 6%	c) 0%	d) 0%	e) 0%
6. ¿Considera que es adecuada la interfaz en general?				
a) 76%	b) 12%	c) 12%	d) 0%	e) 0%

Tabla 2. Resultados del instrumento de evaluación en la sección de Usabilidad.

Fuente: Elaboración propia con información de los resultados.

1. ¿Considera que avanzar, retroceder y saltar a otra página en cualquier momento es fácil?				
a) 53%	b) 29%	c) 12%	d) 6%	e) 0%
2. ¿Considera que los botones responden de acuerdo con lo que usted esperaba?				
a) 82%	b) 18%	c) 0%	d) 0%	e) 0%
3. ¿Considera aceptable el tiempo de espera entre una acción y otra?				
a) 88%	b) 6%	c) 6%	d) 0%	e) 0%
4. ¿Considera adecuados los mensajes de error al usuario (Ej. "Usuario no disponible", etc.)?				
a) 76%	b) 18%	c) 6%	d) 0%	e) 0%
5. ¿Considera que el lenguaje utilizado en la aplicación web es entendible?				
a) 100%	b) 0%	c) 0%	d) 0%	e) 0%
6. ¿El generador web de juegos de acertijos de programación es fácil de usar y navegar?				
a) 82%	b) 6%	c) 12%	d) 0%	e) 0%

Tabla 3. Resultados del instrumento de evaluación en la sección de Funcionalidad.

Fuente: Elaboración propia con información de los resultados.

1. ¿Considera que es fácil jugar los distintos niveles de la aplicación web?				
a) 64%	b) 12%	c) 12%	d) 12%	e) 0%
2. ¿Considera que colocar los bloques de comando en la caja de ejecución y/o función es fácil?				
a) 65%	b) 29%	c) 6%	d) 0%	e) 0%
3. ¿Considera que eliminar los bloques de comando de la caja de ejecución y/o función es fácil?				
a) 64%	b) 18%	c) 6%	d) 12%	e) 0%
4. ¿Considera que es fácil intercambiar los bloques de comando en la caja de ejecución y/o función?				
a) 70%	b) 18%	c) 12%	d) 0%	e) 0%
5. ¿Considera que la lectura del registro de comandos es fácil?				
a) 88%	b) 12%	c) 0%	d) 0%	e) 0%
6. ¿Considera que es fácil manipular la animación (<i>play</i> , pausa, anterior, siguiente)?				
a) 64%	b) 12%	c) 18%	d) 6%	e) 0%
7. ¿Considera que es fácil de entender la relación entre la animación, la secuencia de bloques de comandos ejecutada y el registro de comandos?				
a) 82%	b) 12%	c) 6%	d) 0%	e) 0%
8. ¿Considera que es fácil visualizar los resultados obtenidos en la sección de resultados?				
a) 65%	b) 29%	c) 6%	d) 0%	e) 0%
9. Escribe algunos adjetivos que relaciones con los juegos de la aplicación web (ej. divertido, aburrido, interesante, simple, llamativo, entretenido, etc.)				
Divertido: 10	Entretenido: 10	Interesante: 10	Llamativo: 9	Fácil: 5
Desafiante: 1	Colorido: 1	Bonito: 1	Didáctico: 1	Simple: 1
10. Nos interesa tu opinión sobre la aplicación web, por lo que agradecemos cualquier comentario o sugerencia adicional que puedas proporcionarnos.				
Entre los comentarios destacan los siguientes:				
<ul style="list-style-type: none"> • Muy buen juego • Animaciones agradables • Fácil de entender • Bastante didáctico para aprender lo básico de programación 				
Entre las sugerencias destacan las siguientes:				
<ul style="list-style-type: none"> • Agregar botón para avanzar al siguiente nivel dentro del juego • Agregar cuadro llamativo con resultado 				

- No borrar la solución al momento de reiniciar el nivel, en caso de no superarlo

Instrumento de evaluación para profesores.

Este instrumento también se dividió en tres secciones: diseño estético, usabilidad y funcionalidad. A continuación, se muestran las preguntas que se hicieron a los profesores y los porcentajes que respondieron de cada una de las cinco opciones:

- a) Totalmente de acuerdo.
- b) Ligeramente de acuerdo.
- c) Ni de acuerdo ni en desacuerdo.
- d) Ligeramente en desacuerdo.
- e) Totalmente en desacuerdo.

En la Tabla 4 se muestran los resultados del instrumento de evaluación en cuanto al diseño estético; en la Tabla 5 se presentan los resultados del instrumento de evaluación de la sección de usabilidad; finalmente, en la Tabla 6 se encuentran los resultados del instrumento para la funcionalidad del sistema.

Tabla 4. Resultados del instrumento de evaluación en la sección de Diseño Estético.

Fuente: Elaboración propia con información de los resultados.

1. ¿Considera que es adecuada la organización de la información en la interfaz?				
a) 90%	b) 10%	c) 0%	d) 0%	e) 0%
2. ¿Considera que los botones utilizados en la interfaz son claros y fáciles de usar para navegar?				
a) 100%	b) 0%	c) 0%	d) 0%	e) 0%
3. ¿Considera que son adecuados los colores utilizados?				
a) 100%	b) 0%	c) 0%	d) 0%	e) 0%
4. ¿Considera que el tamaño y el tipo de letra utilizados le permiten leer sin dificultad los contenidos?				
a) 88%	b) 12%	c) 0%	d) 0%	e) 0%
5. ¿Considera que son adecuados los gráficos utilizados en la creación de niveles?				
a) 100%	b) 0%	c) 0%	d) 0%	e) 0%
6. ¿Considera que es adecuada la interfaz en general?				
a) 80%	b) 20%	c) 0%	d) 0%	e) 0%

Tabla 5. Resultados del instrumento de evaluación en la sección de Usabilidad.

Fuente: Elaboración propia con información de los resultados.

1. ¿Considera que es fácil avanzar, retroceder y saltar a otra página en cualquier momento?				
a) 100%	b) 0%	c) 0%	d) 0%	e) 0%
2. ¿Considera que los botones responden de acuerdo con lo que usted esperaba?				
a) 100%	b) 0%	c) 0%	d) 0%	e) 0%
3. ¿Considera aceptable el tiempo de espera entre una acción y otra?				
a) 100%	b) 0%	c) 0%	d) 0%	e) 0%
4. ¿Considera adecuados los mensajes de error al usuario? (Ej. "Usuario no disponible", "El tamaño elegido no es válido. Mínimo 3x3. Máximo 10x10", "Nivel guardado correctamente", etc.)				
a) 88%	b) 12%	c) 0%	d) 0%	e) 0%
5. ¿Considera que el lenguaje utilizado en la aplicación web es entendible?				
a) 88%	b) 12%	c) 0%	d) 0%	e) 0%
6. ¿El generador web de juegos de acertijos de programación es fácil de usar y navegar?				
a) 100%	b) 0%	c) 0%	d) 0%	e) 0%

Tabla 6. Resultados del instrumento de evaluación en la sección de Funcionalidad.

Fuente: Elaboración propia con información de los resultados.

1. ¿Considera que la creación de un mapa es fácil?				
a) 80%	b) 20%	c) 0%	d) 0%	e) 0%
2. ¿Considera que es fácil agregar un nuevo nivel al mapa?				
a) 80%	b) 20%	c) 0%	d) 0%	e) 0%
3. ¿Considera que es fácil generar un tablero?				
a) 100%	b) 0%	c) 0%	d) 0%	e) 0%
4. ¿Considera que es fácil colocar los objetos en el tablero?				
a) 80%	b) 20%	c) 0%	d) 0%	e) 0%
5. ¿Considera que es fácil eliminar los objetos del tablero?				
a) 100%	b) 0%	c) 0%	d) 0%	e) 0%
6. ¿Considera que es fácil intercambiar los objetos del tablero entre sí?				
a) 88%	b) 12%	c) 0%	d) 0%	e) 0%
7. ¿Considera que es fácil volver a editar un tablero, después de haberlo guardado?				
a) 100%	b) 0%	c) 0%	d) 0%	e) 0%
8. ¿Considera que es fácil visualizar los niveles creados previamente?				
a) 100%	b) 0%	c) 0%	d) 0%	e) 0%

CONCLUSIONES.

En este artículo se presentó la evaluación de una aplicación web que permite la creación de juegos que fomentan el aprendizaje de la lógica de programación.

Los resultados de la aplicación de los instrumentos de evaluación fueron satisfactorios, ya que la respuesta de los alumnos fue muy positiva. Algunos alumnos mencionaron que la aplicación web es divertida, y que algunos niveles de prueba a pesar de ser un poco más complejos, es entretenido resolverlos. Los adjetivos más utilizados por los alumnos para describir la aplicación web fueron, con diez ocurrencias, “divertido”, “entretenido”, “interesante”; con nueve ocurrencias, “llamativo”; y con cinco o menos ocurrencias, “fácil”, “desafiante”, “colorido”, “bonito”, “didáctico”, “simple”.

Con respecto a las pruebas de usuario del profesor, los resultados de la evaluación también fueron satisfactorios, debido a que la creación de mapas, así como la creación y edición de niveles les parecieron sencillas e intuitivas. Los profesores mencionaron que era útil el poder volver a editar un nivel en la aplicación web, inclusive después de haberlo guardado.

En cuanto al trabajo a futuro, los resultados obtenidos de las pruebas de usuario mostraron algunos aspectos a mejorar; por ejemplo, es importante agregar un botón cuya funcionalidad sea la de pasar al siguiente nivel cuando se complete exitosamente el nivel actual, esto sin la necesidad de volver a la página de niveles, además de mejorar la pantalla de resultados haciéndola más llamativa. También se puede mejorar el estilo y ubicación de botones para ejecutar la solución de bloques de comandos, reiniciar y salir del nivel, así como mantener la solución propuesta en pantalla, aunque se reinicie el nivel, esto con el objetivo de hacer la experiencia más rápida y menos complicada. Además, sería deseable trasladar este proyecto a una versión móvil, ya que los niños de nivel primaria tienen más acceso a este tipo de dispositivos y así podría llegar a más usuarios.

Finalmente, cabe destacar, que han surgido otras iniciativas para apoyar el proceso de enseñanza-aprendizaje de la programación en diferentes niveles: una aplicación web que muestra la ejecución de un programa a través de una visualización gráfica (Jaimez-González y Castillo-Cortes, 2020), una aplicación web para crear diagramas de flujo (Vazquez-Peñaloza y Jaimez-González, 2019), y un

conjunto de objetos de aprendizaje para apoyar cursos programación estructurada (Luna-Ramírez y Jaimez-González, 2014).

REFERENCIAS BIBLIOGRÁFICAS.

1. Baily, S. (2020). Bobby Duke Case Study. Bobby Duke Middle School. https://codecombat.com/images/pages/impact/pdf/CodeCombat_CaseStudy_BobbyDukeMS.pdf
2. Bain, G., Barnes, I. (2014). Why is programming so hard to learn? Proceedings of the 2014 Innovation and Technology in Computer Science Education Conference (ITICSE '14), Uppsala, Sweden. <http://dx.doi.org/10.1145/2591708.2602675>
3. Bau, D., Gray, J., Kelleher, C., Sheldon, J., Turbak, F. (2017). Learnable programming: blocks and beyond. Communications of the ACM, 6 (6). <https://doi.org/10.1145/3015455>
4. Caccuri, V. (2013). Educación con TICs. 1st ed. Buenos Aires: REDusers. <https://issuu.com/redusers/docs/educacion-con-tics>
5. Code Combat (2020). The most interesting game to learn programming. <https://codecombat.com>
6. Code Karts. A fun app to develop observation, concentration and logic. <https://montessori.edokiacademy.com/en/our-games/discovery/car-game>
7. Coding Adventure (2020). Teach text-based coding by helping a monkey to catch bananas. <https://www.codemonkey.com/coding-adventure/>
8. Egenfeldt-Nielsen, S. (2005). Beyond Edutainment Exploring the Educational Potential of Computer Games. PhD Thesis, IT-University of Copenhagen, Copenhagen, Dinamarca.
9. Erazo-Palacios, J., Jaimez-González, C. R., García-Mendoza, B. (2022). Towards a Web Generator of Programming Games for Primary School Children. International Journal of Engineering Pedagogy (IJEP), 12 (4), pp. 98-114. <https://doi.org/10.3991/ijep.v12i4.17335>
10. Fatourou, E., Zygouris, N. C., Loukopoulos, T., Stamoulis, G. I. (2018). Teaching Con-current Programming Concepts Using Scratch in Primary School: Methodology and Evaluation.

<https://doi.org/10.3991/ijep.v8i4.8216>

11. Fudale, K. (2016). Case Studies for CodeMonkey. Ohio, USA: Edsurge.
<https://www.edsurge.com/product-reviews/codemonkey/educator-case-studies>
12. INEGI. (2019). En México hay 74.3 millones de usuarios de Internet y 18.3 millones de hogares con conexión a este servicio: ENDUTIH 2018.
13. Jaimez-González, C. R., Castillo-Cortes, M. (2020). Web Application to Support the Learning of Programming Through the Graphic Visualization of Programs. *International Journal of Emerging Technologies in Learning*, 15 (6), pp. 33-49. <https://doi.org/10.3991/ijet.v15i06.12157>
14. Jaimez-González, C.R., Erazo-Palacios, J., García-Mendoza, B. (2023). BlockCode: A Web Application to Create Games that Support the Learning of Computer Programming Logic. *International Journal of Emerging Technologies in Learning (iJET)*, 18(15), pp. 240–257.
<https://doi.org/10.3991/ijet.v18i15.40901>
15. Jana, C. (2016). Teacher Review for Lightbot: Programming Puzzles. Commonsense.
<https://www.commonsense.org/app/lightbot-programming-puzzles-teacher-review/4132306>
16. Keller, L., John, I. (2020). Motivating Female Students for Computer Science by Means of Robot Workshops. *International Journal of Engineering Pedagogy (iJEP)*, 10(1), pp. 94–108.
<https://doi.org/10.3991/ijep.v10i1.11661>
17. Liao, Y. K. C., Bright, G. W. (1991). Effects of Computer Programming on Cognitive Outcomes: A Meta-Analysis. *Journal of Educational Computing Research*, 7 (3), pp. 251-268.
<https://doi.org/10.2190/E53G-HH8K-AJRR-K69M>
18. Lightbot (2020). Puzzle game based on coding. Available: <https://lightbot.com>

19. Long, J. (2007). Just for Fun: Using Programming Games in Software Programming Training and Education - A Field Study of IBM Robocode Community. *Journal of Information Technology Education*, vol. 6, pp. 279-290. <https://doi.org/10.28945/216>
20. Luna-Ramírez, W. A., Jaimez-González, C. R. (2014). Supporting Structured Programming Courses Through a Set of Learning Objects. *Proceedings of the IEEE International Conference on Information Society (i-Society 2014)*, pp. 124-128, London, UK. <https://doi.org/10.1109/i-Society.2014.7009024>
21. Maloney, J. H., Peppler, K., Kafai, Y., Resnick, M., Rusk, N. (2008). Programming by choice: urban youth learning programming with Scratch. *Proceedings of the 39th ACM Technical Symposium on Computer Science Education (SIGCSE '08)*, 40 (1), pp. 367-371. <https://doi.org/10.1145/1352322.1352260>
22. Mladenović, M., Krpan, D., Mladenović, S. (2017). Learning programming from Scratch. *Proceedings of the International Conference on New Horizons in Education 2017*, Berlin, Germany.
23. Pereira, J., Frango, I. (2020). A Systematic Review on Open Educational Games for Programming Learning and Teaching. *International Journal of Emerging Technologies in Learning*, 15 (9), pp. 156-172. <https://doi.org/10.3991/ijet.v15i09.12437>
24. Reichert, R., Nievergelt, J., Hartmann, W. (2001). Programming in schools - why, and how? In C. Pellegrini, A. Jacquesson (Eds.): *Enseigner l'informatique*, pp. 143-152.
25. Rogozhkina, I., Kushnirenko, A. (2011). PiktoMir: teaching programming concepts to preschoolers with a new tutorial environment. *Procedia - Social and Behavioral Sciences*, vol.28, pp. 601-605. <https://doi.org/10.1016/j.sbspro.2011.11.114>
26. Sosa, E., Salinas, J., Benito, B. (2017). Emerging Technologies (ETs) in Education: A Systematic Review of the Literature Published between 2006 and 2016. *International Journal of Emerging Technologies in Learning*, 12 (5), pp. 128-149. <https://doi.org/10.3991/ijet.v12i05.6939>

27. Spritebox (2020). A full-blown adventure game that gets you coding. Available: <https://spritebox.com>
28. Steinmaurer, A., Pirker, J., Gütl, C. (2019). sCool – Game-Based Learning in Computer Science Class: A Case Study in Secondary Education. *International Journal of Engineering Pedagogy (iJEP)*, 9(2), pp. 35–50. <https://doi.org/10.3991/ijep.v9i2.9942>
29. Vazquez-Peñaloza, F., Jaimez-González, C. R. (2019). Towards a Web Application to Create Flowcharts for Supporting the Teaching-Learning Process of Structured Programming Courses. *American Journal of Educational Research*, 7 (12), pp. 976-982. <http://www.sciepub.com/education/abstract/11332>
30. Weintrop, D., Wilensky, U. (2015). Using Commutative Assessments to Compare Conceptual Understanding in Blocks-based and Text-based Programs. *Proceedings of the 11th International Conference on International Computing Education Research (ICER'15)*, pp. 101-110. <https://doi.org/10.1145/2787622.2787721>
31. Yadin, A. (2011). Reducing the dropout rate in an introductory programming course. *ACM Inroads*, vol. 2 (4), pp. 71-76. <https://doi.org/10.1145/2038876.2038894>
32. Zaharija, G., Mladenović, S., Boljat, I. (2013). Introducing basic Programming Concepts to Elementary School Children. *Procedia – Social and Behavioral Sciences*, vol. 106, pp. 1575-1584. <https://doi.org/10.1016/j.sbspro.2013.12.178>

DATOS DE LOS AUTORES.

1. Carlos Roberto Jaimez-González. Doctor en Ciencias de la Computación por la Universidad de Essex, Reino Unido. Maestro en Tecnologías para Comercio Electrónico por la Universidad de Essex. Licenciado en Computación por la Universidad Autónoma Metropolitana, México. Profesor investigador de tiempo completo del Departamento de Tecnologías de la Información de la Universidad Autónoma Metropolitana, Unidad Cuajimalpa, México. Correo electrónico: cjaimez@cua.uam.mx

2. Betzabet García-Mendoza. Maestra en Diseño, Información y Comunicación por la Universidad Autónoma Metropolitana, México. Licenciada en Tecnologías y Sistemas de Información por la Universidad Autónoma Metropolitana, México. Profesora investigadora de tiempo completo del Departamento de Tecnologías de la Información de la Universidad Autónoma Metropolitana, Unidad Cuajimalpa, México. Correo electrónico: bgmendoza@cua.uam.mx

3. Javier Erazo-Palacios. Licenciado en Tecnologías y Sistemas de Información de la Universidad Autónoma Metropolitana, Unidad Cuajimalpa, México. Correo electrónico: 2163030922@cua.uam.mx

RECIBIDO: 4 de septiembre del 2024.

APROBADO: 25 de septiembre del 2024.