

Metodología para el diseño de juegos serios: Articoding, un juego para desarrollar el pensamiento computacional a través de la programación

Methodology for designing games: Articoding, a serious game to develop computational thinking through programming

Antonio Calvo-Morata

Universidad Complutense de Madrid. Madrid, España

acmorata@ucm.es

Cristina Alonso-Fernández

Universidad Complutense de Madrid. Madrid, España

calonsofernandez@ucm.es

Iván Martínez-Ortiz

Universidad Complutense de Madrid. Madrid, España

imartinez@fdi.ucm.es

Baltasar Fernández-Manjón

Universidad Complutense de Madrid. Madrid, España

balta@ucm.es

...

Resumen

El avance tecnológico ha incrementado la importancia de adquirir habilidades en pensamiento computacional y programación, viéndose reflejado en los currículos escolares. Sin embargo, en muchos casos los profesores no son expertos en la materia y hay pocas herramientas validadas que faciliten su labor. Consideramos que los videojuegos educativos son herramientas versátiles en el aprendizaje de conceptos complejos y pueden aplicarse para aprender programación. En este artículo presentamos Articoding, un juego serio para desarrollar el pensamiento computacional a través de la programación. Junto con el diseño del videojuego, se describe la metodología incremental aplicada basada en experiencias en el aula en diferentes etapas de su desarrollo. En total participaron 469 estudiantes de secundaria y 36 estudiantes universitarios de educación. Estas experiencias han servido para estudiar la aplicabilidad del juego en el aula, y su aceptación por parte de estudiantes y futuros profesores. Los cuestionarios y las analíticas recogidas mientras juegan los estudiantes muestran resultados prometedores sobre su uso y aplicabilidad en clase. Los problemas encontrados han sido mínimos y las opiniones muy positivas. Además, los datos también identifican aspectos a mejorar (p.ej., la dificultad de algunos niveles). Con estos resultados estamos elaborando una nueva versión de Articoding, mejorando el progreso de los niveles, introduciendo nuevo contenido e incluyendo analíticas de aprendizaje mejoradas para su uso como herramienta de apoyo docente en la enseñanza de programación.

Palabras clave: Pensamiento computacional, videojuegos, programación.

Abstract

Technological progress has increased the importance of acquiring skills in computational thinking and programming, which is reflected in school curricula. However, in many cases teachers are not experts in the field and there are few validated tools to facilitate their work. We believe that educational videogames are versatile tools for learning complex concepts and can be applied to learn programming. In this article we present Articoding, a serious game to develop

computational thinking through programming. Along with the design of the videogame, we describe the incremental methodology applied based on classroom experiences at different stages of its development. A total of 469 high school students and 36 university education students participated. These experiences have served to study the applicability of the game in the classroom, and its acceptance by students and future teachers. The questionnaires and analytics collected while playing the game show promising results about its use in the classroom. These data also identify areas for improvement (e.g., the difficulty of some levels). With these results, a new version of Articoding is being developed, including improved learning analytics for its use as a support tool for teaching programming.

Key words: Computational thinking, videogames, programming.

1. Introducción

Los conocimientos básicos de informática en etapas tempranas son un elemento clave para entender mejor el mundo actual y participar en la transformación digital de la sociedad y la economía. Con la generalización del uso de tecnología en todas las facetas de la vida cotidiana y la aparición de nuevas aplicaciones informáticas cada vez más sofisticadas (como aplicaciones basadas en inteligencia artificial), cada vez es más necesario tener un conocimiento básico de informática y programación. Por este motivo, el pensamiento computacional y la programación han adquirido gran relevancia y empiezan a reflejarse en los programas educativos obligatorios de los diferentes países del mundo (Bocconi et al., 2022)

Hasta ahora, la programación y el pensamiento computacional (Zapata-Ros, 2015) eran materias complementarias dentro del currículo de las escuelas, y su evaluación no era sistemática. Por ejemplo, el Programa de la OECD para la Evaluación Internacional (PISA) evaluaba el pensamiento computacional dentro del marco de las matemáticas, prestando poca atención a la programación como medio para desarrollar el pensamiento computacional y como conocimiento básico para la sociedad actual. Sin embargo, esto ha empezado a cambiar y la informática se está convirtiendo, cada vez con más frecuencia, en una asignatura troncal dentro de las escuelas. Y esta materia será cada vez más relevante, ya que la nueva evaluación PISA para 2025 incluye una prueba llamada “Aprender en el mundo digital” (Organisation for Economic Co-operation and Development, 2023), que evalúa la capacidad de los estudiantes para participar en un proceso iterativo de construcción de conocimiento y resolución de problemas utilizando herramientas computacionales y de programación. Esta prueba incluirá preguntas que deberán resolverse mediante un lenguaje de programación visual basado en bloques (similar a Scratch¹). Por lo tanto, las escuelas tendrán que mejorar el proceso de enseñanza de los conceptos básicos de programación.

La introducción de la programación y el pensamiento computacional como materias troncales en el currículo no está exenta de problemas. Entre ellos podemos encontrar: la falta de interés en la materia por parte de los estudiantes, en parte por percibirse como una materia compleja; la falta de profesorado en educación obligatoria con los conocimientos necesarios de programación; o la falta de herramientas que realmente sean efectivas y pueda justificarse su uso en el aula. Si bien podemos encontrar diferentes herramientas y plataformas que abordan la enseñanza de la programación o el desarrollo

¹ <https://scratch.mit.edu/>

del pensamiento computacional, son pocas las que presentan evidencias de su validez o aplicabilidad práctica en el aula (Varghese & Renumol, 2023). Por otro lado, muchas requieren el pago de licencias de uso o son soluciones demasiado complejas y abiertas para el docente, limitando su aplicación de forma efectiva en las aulas (Bile, 2022).

Los videojuegos pueden ser una herramienta con la que enseñar programación y desarrollar el pensamiento computacional, haciéndolo una materia atractiva y fomentando el interés hacia disciplinas STEM (Calvo-Morata et al., 2024). Precisamente, existen evidencias que apuntan a que el contacto temprano con los videojuegos y con la programación parecen ser factores que influyen a la hora de elegir estudios y un futuro profesional. Por ejemplo, (Jie Du & Hayden Wimmer, 2019) señalan que las personas que son expuestas a la programación en edades tempranas terminan mostrando mayor interés en estudiar carreras en este campo. Por su parte, (Hosein, 2019) sugiere que las chicas que juegan videojuegos tienen mayor probabilidad de estudiar carreras STEM que aquellas que no lo hacen, siendo esto especialmente relevante dada la baja inserción laboral femenina en el ámbito de trabajos relacionados con la programación y las nuevas tecnologías.

Como herramienta educativa, los videojuegos tienen ventajas ya que consiguen combinar el entretenimiento con el aprendizaje. Los juegos específicamente diseñados con fines educativos pueden ser potentes herramientas interactivas para reforzar conocimientos y desarrollar habilidades, adaptándose a distintos niveles de los estudiantes, quienes deben aplicar y poner a prueba lo aprendido en situaciones nuevas propuestas por el juego. A través de las diferentes mecánicas de juego que ofrecen, pueden lograr fomentar la resolución de problemas, el pensamiento crítico o la creatividad. Además, los videojuegos educativos permiten que los estudiantes controlen su proceso de aprendizaje al poder avanzar a su propio ritmo, facilitando así la comprensión y la retención de conceptos.

Teniendo en cuenta estos antecedentes, este trabajo presenta el diseño de Articoding, un videojuego para desarrollar el pensamiento computacional a través de la programación, así como su validación inicial a través de diversas experiencias en el aula. Estas primeras experiencias realizadas en el aula con distintos tipos de alumnos sirven para demostrar su aplicabilidad en educación secundaria y estudiar las ventajas que puede aportar una solución de este tipo.

2. Trabajo relacionado: pensamiento computacional, programación y juegos educativos para programación

La definición más globalmente aceptada sobre pensamiento computacional (PC) lo define como “los procesos de pensamiento que intervienen en la formulación de problemas y sus soluciones, de modo que éstas se representen de forma que puedan ser llevadas a cabo por un agente de tratamiento de la información” (Wing, 2006). Según (Anderson, 2016), los elementos clave del PC son: (1) descomposición de un problema, (2) reconocimiento de patrones, (3) abstracción (generalización de patrones repetidos), (4) diseño de algoritmos para la resolución del problema, y (5) evaluación de soluciones. Sin embargo, no hay un consenso global sobre cuál es la definición de PC ni sobre cómo modelarlo o evaluarlo (Shute et al., 2017).

Si bien hay matices en sus distintas definiciones, no hay duda de que el PC está cobrando cada vez más importancia en la educación y como requisito para entender mejor los cambios tecnológicos (como la Inteligencia Artificial Generativa, p. ej., ChatGPT). Analizando los datos presentados en el informe PISA de 2022, algunos autores han destacado “la necesidad de promover programas sobre pensamiento computacional en los centros educativos para fomentar la vocación de los estudiantes en las disciplinas STEM, utilizar aplicaciones para desarrollar el pensamiento computacional y reforzar la competencia digital del alumnado” (Ortega-Rodríguez, 2025). En el Estudio Internacional sobre Competencia Digital ICLIS de 2023 también se destaca que tanto la competencia digital (CD) como el PC tienen mayor presencia en la enseñanza secundaria que en la primaria y que hay mayor probabilidad de que la CD tenga carácter obligatorio frente al PC (Fraillon & Rožman, 2023).

Al igual que no existe un consenso sobre la definición de PC, tampoco hay una estrategia clara sobre cómo abordarlo. Distintos estudios han adaptado estrategias de aprendizaje basado en proyectos, aprendizaje basado en problemas, aprendizaje cooperativo y aprendizaje basado en juegos para abordarlo (Hsu et al., 2018). Para enseñar PC, la mayoría de estas iniciativas se enfocaron en el diseño de programas a través de lenguajes de programación visuales (p. ej., Scratch). También se han desarrollado múltiples herramientas para abordar aspectos tanto de programación como de PC, en muchos casos de forma relacionada como el desarrollo del PC a través del diseño y el desarrollo de distintos proyectos de programación (Basogain et al., 2018). Y si bien también podemos encontrar actividades denominadas desconectadas (*unplugged*), es decir, en las que se usan lápiz y papel (Brackmann et al., 2017), los pocos estudios existentes sugieren que a la hora de mejorar el pensamiento computacional y aprender programación son mejores aquellas actividades conectadas (Sigayret et al., 2022).

Una revisión sobre juegos para programación encontró que estos se centraban en introducción a la programación y resolución de problemas (Miljanovic & Bradbury, 2018). También se han aplicado entornos digitales inmersivos tridimensionales (EDIT) para la enseñanza de programación, en particular utilizando herramientas de realidad aumentada a nivel universitario y principalmente para aprendizaje de estructuras de control (López Solórzano & Ángel Rueda, 2023). Además, existen multitud de ejemplos de juegos y plataformas de programación, tales como: la plataforma Blockly Games² que enseña programación mediante puzzles interactivos utilizando bloques gráficos; Rabbids Coding³, un juego de Ubisoft para introducir programación a través de misiones; CodeCombat⁴ que enseña programación a través de un juego de rol donde se escribe código para avanzar en niveles; Programming Hero⁵, plataforma interactiva para aprender a programar, con minijuegos y desafíos; Toolbox Academy⁶, plataforma para aprender programación de forma práctica con proyectos y ejemplos; CodeSpark⁷, un juego educativo que enseña conceptos de programación a través de juegos de lógica y

² <https://blockly.games>

³ <https://store.ubisoft.com/es/rabbids-coding/5d96f9b05cdf9a2eacdf68cb.html>

⁴ <https://codecombat.com/>

⁵ <https://www.programming-hero.com/>

⁶ <https://toolbox.academy/>

⁷ <https://codespark.com/play>

rompecabezas; LightBot⁸, un juego de rompecabezas que enseña conceptos básicos de programación a través de la resolución de problemas con un robot; Human Resource Machine⁹, un juego de lógica y programación en el que resolver tareas de oficina utilizando conceptos de programación; o TIS-100¹⁰, un juego de simulación de programación en el que los jugadores deben reprogramar una computadora antigua utilizando ensamblador. Sin embargo, no existen evidencias de la aplicación y efectividad en el aula de muchas de estas aplicaciones.

Ya centrados específicamente en el ámbito del PC, la revisión de literatura de (Varghese & Renumol, 2024) se centra en los videojuegos como herramientas para evaluar PC. Entre sus resultados, además de la gran variedad de tiempos de uso, tipo de participantes y aspectos de PC evaluados en las aplicaciones, destacan el limitado número de estudios empíricos que demuestren que los videojuegos son herramientas útiles para evaluar las habilidades relacionadas con el PC. Algunos estudios recientes han abordado estos aspectos, como el caso de (Zhao & Shute, 2019) en el que su videojuego mejoró significativamente las habilidades de PC de los participantes, o el de (Liu, 2024) en el que evaluaron distintos aspectos que influyen en la mejora de las habilidades de PC con un juego basado en puzzles. Entre sus conclusiones, no encontraron diferencias significativas por género o curso, mientras que las acciones en el juego (duración, dificultad de los niveles, etc.) sí influyen en el aprendizaje de PC.

3. Metodología y Diseño del Videojuego

Articoding (Calvo-Morata et al., 2025) es un videojuego educativo de puzzles que tiene como objetivo mejorar el aprendizaje de los conceptos básicos de programación de manera incremental a través de sus diferentes niveles. El juego se ha diseñado para ser utilizado en el aula con alumnos de 12 a 16 años y pretende servir como herramienta de apoyo al profesor. En cada uno de los puzzles que se presentan, el jugador debe utilizar la programación visual basada en bloques para conseguir que el láser del nivel llegue a colisionar con un receptor. El lenguaje usado es Blockly¹¹ a través de una reimplementación del intérprete¹² en el entorno de juego (lo que permite ejecutar y comprobar las soluciones de los alumnos y no sólo usar soluciones previamente previstas).

En el desarrollo de Articoding se siguió un proceso incremental centrado en la opinión y análisis de expertos y en la retroalimentación obtenida de su uso en centros educativos (ver Figura 1). A la hora de diseñar la primera versión de Articoding se analizaron los juegos educativos disponibles para Scratch, así como otros enfoques basados en juegos (p. ej., creación o modificación de juegos básicos) y se observaron características comunes como una alta interacción, o el uso de animaciones y entornos atractivos y motivadores para los alumnos. Sin embargo, no estaba tan claro que los alumnos adquirieran los conceptos e ideas de programación o pensamiento computacional.

⁸ <https://lightbot.com>

⁹ <https://tomorrowcorporation.com/humanresourcemachine>

¹⁰ <https://www.zachtronics.com/tis-100>

¹¹ <https://developers.google.com/blockly>

¹² <https://github.com/imagicbell/ublockly>

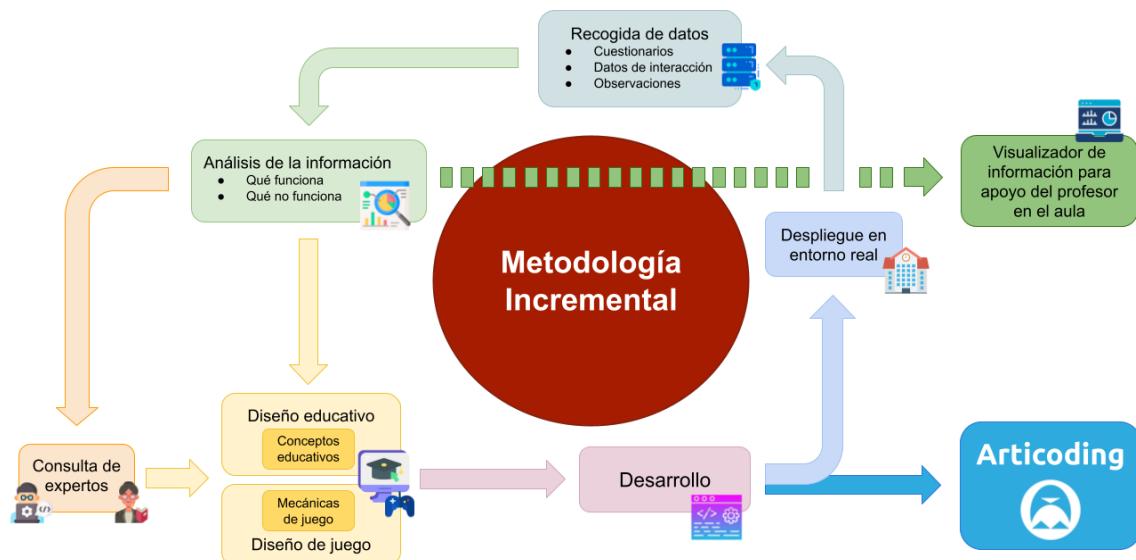
A partir de la opinión de expertos, que incluían tanto docentes de secundaria como ingenieros informáticos, el diseño de Articoding partió de los siguientes requisitos:

- El entorno de aprendizaje debía ser un juego simple basado en puzzles o desafíos, con una mecánica de juego sencilla de aprender, pero suficientemente flexible para que se pudieran ilustrar y practicar los principales conceptos de programación.
- El conjunto de puzzles o desafíos de juego debían estar dirigidos por los conceptos de programación (p. ej., variables, expresiones, condicionales, bucles, funciones) y que se pudieran ir planteando de forma incremental y con dificultad progresiva. Se debía partir siempre de un tutorial que ilustrara el concepto, de un ejercicio simple que lo ilustre y a partir de ahí con dificultad incremental se incluirían un número suficiente de niveles para que el alumno pueda practicar e interiorizar el concepto.
- El entorno de juego debía permitir soluciones diferentes de los estudiantes. Para lograrlo, se debía integrar un intérprete del código que realmente ejecuta la solución propuesta por el usuario. Esto permite una mayor flexibilidad y añade otro requisito, un sistema que permitía a los alumnos aprender de sus errores y saber si, incluso tras una solución válida, aún pueden mejorar su solución.
- La solución de los desafíos, salvo en los niveles más simples e introductorios, no debe depender únicamente del estado de un único objeto.

Cada nueva versión de Articoding fue desplegada y probada en un centro educativo, recogiendo datos de diferentes fuentes para analizar la efectividad y uso del juego en el aula y así mejorar el diseño y el propio juego de cara a obtener una herramienta efectiva para desarrollar el pensamiento computacional y aprender los conceptos básicos de programación. De manera indirecta también se ha obtenido un conjunto de análisis que pueden servir al profesor para conocer qué hacen los estudiantes durante las sesiones de juego de Articoding y qué problemas de aprendizaje tienen.

Figura 1

Diagrama de la metodología propuesta para el diseño de juegos serios



3.1 Articoding y los Conceptos de Programación

Articoding se compone de distintos niveles agrupados en categorías diferentes (ver Figura 2 que muestra un ejemplo de nivel y Figura 3, donde se muestran algunas de las 7 categorías de la versión actual del juego), cada una de ellas con un mínimo y máximo de niveles. En la versión actual del juego, cada categoría tiene un número mínimo de 3 niveles y un máximo de 9 niveles. Por otro lado, cada categoría introduce al jugador un nuevo concepto de programación y propone puzzles con los que aprender y practicar dicho concepto. Estas categorías y niveles presentan una dificultad incremental (ya que hay que comprender y utilizar los conceptos y bloques anteriores) y el jugador debe superar siempre el nivel anterior para poder avanzar al siguiente. Del mismo modo se debe completar una categoría para acceder a la siguiente. Los niveles cuentan con tutoriales iniciales para explicar el funcionamiento de los elementos según se presentan en el juego (ver Figura 4). Las 7 categorías secuenciales que se consideran en el modelo de Articoding son:

1. **Introducción:** presenta las mecánicas básicas del juego, qué son los bloques (es decir las instrucciones de este lenguaje de programación visual), cómo se enlazan para crear un programa y cómo se ejecuta dicho programa. A lo largo de esta categoría el jugador aprende a mover y girar el objeto que genera el láser y que el objetivo para completar cada nivel es conseguir que el láser llegue al receptor. En esta introducción el jugador debe utilizar sólo bloques básicos de movimiento y bloques de números que permiten indicar la cantidad de esos movimientos.
2. **Variables:** introduce el concepto de variable en programación, para ello se presentan niveles donde el jugador no tiene suficientes bloques de números y debe recurrir a asignar dichos valores en bloques de variables para reutilizarlos.
3. **Tipos de datos:** el juego presenta nuevos objetos con los que los bloques pueden interactuar. Estos son los espejos, que son capaces de reflejar el rayo láser en ángulos de 90 grados, y las puertas, cuyo estado puede ser: abierto, y deja pasar

el láser; o cerrado y bloquearlo. Además, introduce nuevos tipos de bloques. Por un lado, los de texto, que permiten, mediante un identificador, decidir sobre qué elemento presente en el nivel deben realizarse las acciones de movimiento. Por otro lado, presenta los bloques que representan booleanos, y que permiten abrir los elementos de tipo puerta que aparezcan en el nivel.

4. **Operadores básicos:** los niveles de esta categoría introducen operadores como nuevos bloques y evaluación de expresiones (resultado de usar operadores). Entre ellos, los jugadores disponen de operadores aritméticos como suma o resta, y operadores lógicos como AND y OR (y-lógica y o-lógica). Los niveles de esta categoría están diseñados para completarse con variables que deben utilizarse para operar obteniendo nuevos valores.
5. **Bucles:** en esta categoría se presenta el concepto de bucle (repeticIÓN) con nuevos bloques que es necesario usar para completar los niveles debido a las restricciones en la cantidad de bloques de movimiento simple.
6. **Condiciones:** en estos niveles se introduce un nuevo tipo de suelo con distintos colores (verde, rojo o azul). Este nuevo elemento se usa para introducir al jugador el concepto de condición y los bloques “if”, “else if” y “else” ya que debe comprobar dicho color para realizar una acción u otra.
7. **Funciones:** por último, en estos niveles se restringen los bloques de bucle y se fomenta que el jugador defina y utilice funciones para ejecutar código que se repite (abstracción funcional y encapsulamiento de comportamiento).

La progresión en la presentación de los diferentes conceptos, en el incremento de dificultad y en la cantidad y diversidad de bloques a utilizar para resolver el puzzle, pretenden contribuir a desarrollar el pensamiento computacional (y sus 5 elementos clave definidos por (Anderson, 2016)). De este modo, para resolver cada nivel los jugadores deben: (1) descomponer el puzzle y la situación que se les presenta, reconociendo los diferentes elementos que aparecen y los bloques disponibles; (2 y 3) reconocer los patrones de los niveles según progresan para abstraer y generalizar el problema resolver, lo cual es necesario para averiguar qué bloques deben usarse de los disponibles; (4) con todo lo anterior, el jugador debe diseñar e implementar un programa que dé solución al puzzle planteado y (5) debe ejecutar su solución y evaluar su calidad. Puede ser desde que la solución no sea correcta hasta que incluso resolviendo el puzzle no sea óptima (esto se le indica mediante las estrellas) y tenga que decidir si quiere mejorarla o prefiere avanzar, y así hasta llegar a completar el juego.

Figura 2

Captura de pantalla de un nivel de juego con la parte de edición de código (izquierda) y la vista del puzzle a resolver (derecha). En la parte superior aparecen, de izquierda a derecha: el nombre del nivel; y los botones de ejecución, de información útil y de salir del nivel.

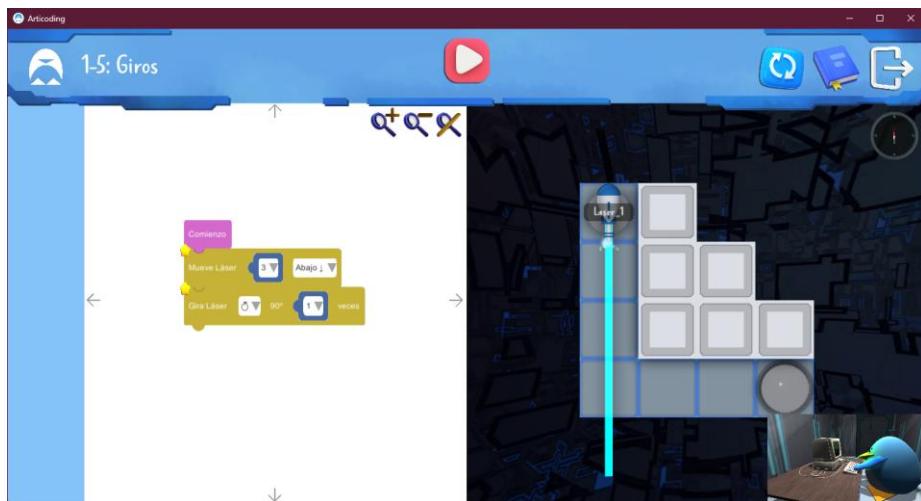
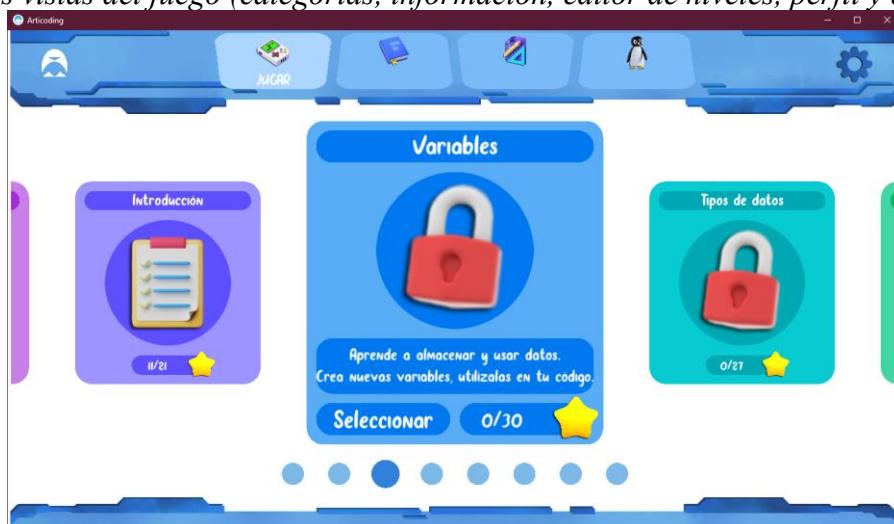


Figura 3

Captura de pantalla de la vista de categorías. La parte central contiene las categorías y su información. En la parte superior se encuentran botones de navegación para ir a las diferentes vistas del juego (categorías, información, editor de niveles, perfil y ajustes).



3.2 Rejugabilidad y Retención

Con el objetivo de aumentar la rejugabilidad y fomentar la sensación de superación, en cada uno de los niveles el jugador puede conseguir hasta 3 estrellas, que representan diferentes logros: haber usado el mínimo número de bloques de movimiento necesarios; usar el bloque especial de cada nivel; y no dejar bloques sueltos en el nivel desconectados del programa principal (ver Figura 5). El logro de usar un bloque concreto se debe a que los niveles pueden completarse de diversas formas (por ejemplo, sin usar variables), pero el bloque especial de cada nivel busca fomentar que el jugador utilice y practique los

conceptos nuevos o de interés del nivel. El juego incluye un intérprete del lenguaje de bloques lo que posibilita que los niveles admitan diferentes soluciones (no óptimas). Esto ayuda a que la dificultad sea más progresiva para aquellos estudiantes que puedan tener mayor problema en entender los conceptos que se presentan.

Figura 4

Captura de pantalla de una ventana emergente de tutorial, explicando, en este caso, donde encuentra el jugador su código y cómo se realiza la ejecución de éste.

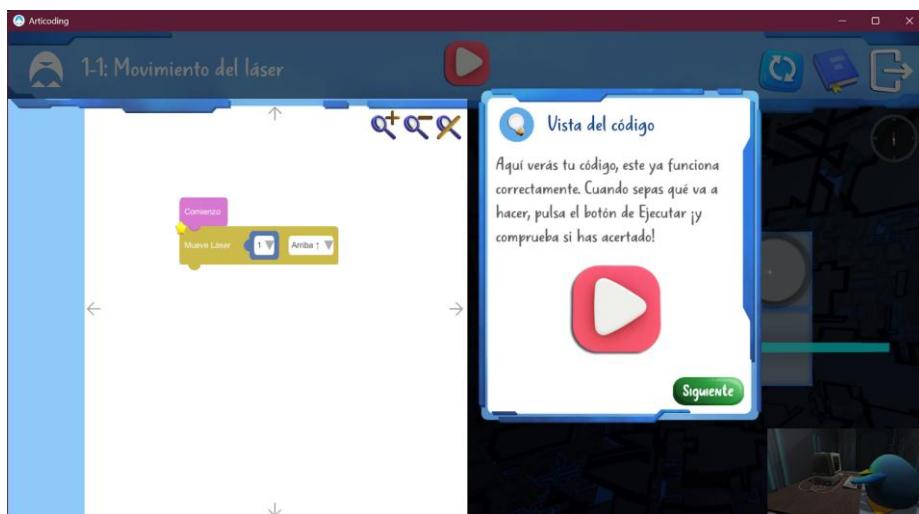


Figura 5

Captura de pantalla del panel emergente que aparece al superar el nivel y que indica las estrellas conseguidas.



Además, el propio juego va introduciendo de manera breve los nuevos conceptos para guiar al jugador (de esta forma se reduce el esfuerzo requerido por parte del profesor para desplegar el juego en la clase).

3.3 Adaptación a Distintos Niveles Educativos y Tipos de Alumnado

Junto con el juego, se desarrolló un editor (Figura 6) que permite crear nuevos niveles. Esta característica permite que profesores y educadores creen sus propios niveles, abordando mejor la diversidad de alumnos en una clase y especialmente aquellos que destacan en la actividad. Posibilita que los alumnos que mejor entienden la programación y que avanzan más rápidamente tengan una actividad desafiante y motivadora una vez que han completado la parte básica del juego. También permite adaptar el juego a las necesidades a diferentes niveles educativos, desde últimos años de educación primaria a un posible curso cero de universidad.

Asimismo, el editor permite que los alumnos demuestren los conocimientos adquiridos creando sus propios niveles, que pueden compartir con otros compañeros, permitiendo crear nuevas dinámicas en la clase según las necesidades del profesor. A la hora de editar un nivel, el jugador puede seleccionar el tamaño del mapa, colocar los elementos que quiera teniendo en cuenta que debe haber un láser y un receptor, y puede restringir los bloques que pueden usarse para resolver el nivel, así como la cantidad de bloques de cada tipo que pueden usarse. Para poder publicar un nivel creado con el editor, este debe ser completado por el jugador antes, de esta forma se asegura que todo nivel creado sea resoluble. Esta característica también permite que el profesor use mejor su tiempo y se centre en ayudar a aquellos alumnos que tienen más problemas de comprensión de los conceptos de programación.

Figura 6

Captura de pantalla del editor de niveles. El jugador puede editar el tamaño del puzzle y añadir objetos arrastrándolos desde el panel inferior al interior del tablero.



El diseño y desarrollo del juego también permite la posibilidad de añadir nuevos conceptos, permitiendo crear nuevas versiones del juego con más categorías, niveles y bloques de manera sencilla, siendo escalable a otros conceptos que quieran enseñarse o introducirse en un futuro. El desarrollo del juego fue incremental, su primera versión no contemplaba la categoría de Introducción ni la de Funciones, que fueron introducidas a posteriori. Estas características fueron introducidas en el curso 2022, junto a una primera

versión del editor. Y en 2024 se cambió el diseño de algunos niveles, se mejoró la interfaz y el editor y se creó la comunidad.

Por último, el videojuego se ha desarrolló siguiendo aspectos de accesibilidad: la interfaz utiliza combinaciones de color con suficiente contraste, para garantizar su adecuada legibilidad y reducir barreras visuales. La navegación se realiza principalmente mediante ratón, y se ha realizado una composición de elementos interactivos que facilita su uso por parte de estudiantes con distintos niveles de familiaridad tecnológica. El juego no requiere grandes requisitos técnicos, por lo que se puede ejecutar en portátiles y ordenadores de baja gama, sin tarjeta gráfica dedicada y sin requerimiento de conexión a Internet en la versión en la que no se recogen datos de interacción. Desde la perspectiva de equidad, el videojuego se distribuye de manera gratuita, está publicado en la web del grupo de investigación e-UCM, es de código abierto y ha sido diseñado para funcionar en dispositivos ampliamente disponibles en contextos educativos, priorizando la inclusión y la reducción de brechas de acceso digital.

3.4 Contexto Educativo de Uso

Articoding está diseñado para ser usado en el aula informática a lo largo de varias sesiones de una hora aproximadamente, dedicando el tiempo de cada una de ellas a una de las diferentes categorías. Estas sesiones pueden ser enfocadas por los profesores según su criterio y necesidades. Por ejemplo, algunas de esas opciones serían:

- **Como hilo conductor.** El uso del juego como herramienta para introducir los elementos básicos de programación o para desarrollar el pensamiento computacional, permitiendo que el profesor se centre en aquellos conceptos y actividades en los que más dificultades tenga la clase, o incluso cada estudiante haciendo uso de las diferentes características que proporciona Articoding, como el editor y la comunidad.
- **Como herramienta de apoyo.** El uso del juego como herramienta de ejercicios para practicar los conceptos de programación que hayan sido explicados por el profesor previamente. Como se ha demostrado en numerosas experiencias con juegos, practicar con este tipo de herramientas educativas motiva más a los estudiantes y les ayuda a romper la barrera de los 10 minutos de atención, consiguiendo que trabajen durante más tiempo los elementos introducidos por el juego.
- **Como herramienta de autoevaluación.** El uso del juego como herramienta de repaso para estudiantes que ya hayan aprendido los conceptos básicos de programación y quieran revisarlos, además de seguir desarrollando sus habilidades de pensamiento computacional.

El juego está desarrollado para ser usado en PC únicamente. Esta decisión se debe principalmente a los siguientes aspectos: (1) muchos centros escolares carecen de tabletas o tienen políticas que prohíben o restringen el uso de dispositivos móviles, siendo este último caso cada vez más común con escuelas que optan por restringir el uso de los dispositivos móviles en las aulas (Steven Saphore, 2024); (2) hay estudios que sugieren que hay diferencias en el comportamiento y el aprendizaje de los estudiantes al utilizar ordenadores frente a dispositivos móviles (Sage et al., 2022); (3) existe un mayor control de la actividad utilizando PC que dispositivos móviles, donde a los docentes les resulta

más difícil monitorear a cada estudiante, ya que pueden abrir otras aplicaciones de forma sencilla durante la sesión. Esto es más fácil de supervisar en ordenadores, donde las pantallas y su posición vertical permiten un mayor control a primera vista.

El diseño, que incluye la explicación de los conceptos básicos y la estructura progresiva del juego, permite que los alumnos que comprenden mejor la programación completen los niveles de manera independiente y sin requerir ayuda por parte del profesor. Así los alumnos adquieren o afianzan los conceptos ya que los tienen que aplicar resolviendo el desafío de cada nivel y recibiendo una realimentación automática del juego sobre la corrección y “calidad” de su solución. Esto permite que el docente use mejor su tiempo en clase y se centre en aquellos estudiantes con mayor dificultad para entender y aplicar los conceptos de programación presentes en el juego.

Como característica añadida, Articoding integra analíticas de aprendizaje, que capturan las interacciones de cada uno de los estudiantes en tiempo real, y de forma anonimizada en origen (los alumnos están identificados por un token anonimizado de 4 caracteres). Entre los datos que se pueden generar está el tiempo de juego, el código que se ejecuta en cada nivel, el número de intentos para superarlo y si la solución ejecutada es correcta o no. Esto permite mejorar la evaluación de Articoding como herramienta de aprendizaje de la programación ya que permite relacionar los datos obtenidos en los cuestionarios con los datos de interacción con el juego. Estas evidencias de las interacciones con el juego se utilizaron para estudiar la aplicabilidad del juego en clase, como se detalla en los casos de estudio descritos a continuación. Y también permitirá el desarrollo de nuevas herramientas de seguimiento que den apoyo al profesor durante el uso del juego en el aula.

3.5 Novedad de la Propuesta

Si bien existen muchas alternativas de juegos para enseñar programación, no todas ellas se enfocan al mismo rango de edad, han sido diseñadas para el mismo contexto de uso, han sido probadas en el aula o son gratuitas. Juegos como Lightbolt van orientados a rangos de edad menores a 10 años, mientras que opciones como Hacknet o TIS-100 se enfocan a usuarios mayores a 16 años y resultan más complejos.

Por otro lado, Articoding, a diferencia de Scratch, que se enfoca en proyectos, tiene categorías y niveles, presenta una consecución definida de objetivos y conocimiento que presenta poco a poco, simplificando su uso por parte del profesor. Y a diferencia de CodeCombat o Human Resource Machine, que si tienen una estructura por niveles, es gratuito, de código libre y ha sido diseñado para ser en clase, además de presentar programación visual por bloques en vez de programación textual. Además, Articoding es de código abierto, permitiendo, además, su modificación y aplicación de forma gratuita.

Además, el editor de niveles y la funcionalidad de comunidad permite, si el profesorado lo necesita, crear niveles propios, ampliando la cantidad de ejercicios y permitiendo hacer hincapié en aquellos conceptos más necesarios para la clase específica. También permite a los propios usuarios practicar su conocimiento poniendo retos a sus compañeros. Mientras que los requisitos de hardware facilitan su ejecución en cualquier ordenador que tenga Windows instalado.

3.6 Experiencias en Entorno Real

A lo largo de sus diferentes versiones, Articoding se ha utilizado con usuarios en entornos reales, tanto en institutos con estudiantes de educación secundaria (su público objetivo), como con estudiantes de estudios superiores en educación, futuros profesores que podrán utilizar el juego para sus clases. Estas experiencias tenían un objetivo doble: en primer lugar, analizar y mejorar la aplicabilidad del juego en las clases, a través de evidencias recogidas de las interacciones con el juego, sobre tiempos de juego y niveles completados; mientras, por otro lado, y mediante cuestionarios, recoger información sobre la opinión y aceptación del juego por parte de los usuarios.

A la hora de llevar a cabo la experiencia con Articoding en el aula se han seguido las siguientes preguntas de investigación:

- RQ1. ¿El juego es fácil de usar y aplicar en el aula? ¿Hay algún elemento o mecánica que les cuesta entender a los jugadores?
- RQ2. ¿Los estudiantes son capaces de interactuar con el juego y completar los niveles por sí mismos?
- RQ3. ¿Los niveles están bien diseñados y el incremento de dificultad es adecuado a lo largo del tiempo?
- RQ4. ¿El juego es atractivo para los estudiantes y les gusta?
- RQ5. ¿Los profesores consideran el juego de Articoding como una herramienta útil?
- RQ6. ¿Los profesores estarían interesados en usar Articoding en sus aulas?

En particular, las dos primeras versiones de Articoding han sido utilizadas con estudiantes de secundaria, con el objetivo de evaluar el aprendizaje y validar el juego, obteniendo resultados positivos de su aplicación, así como aspectos a mejorar en su diseño que fueron abordados en las siguientes iteraciones del juego. Además de esta validación inicial, también se evaluó la opinión de los participantes sobre el juego y su usabilidad (cuestionario detallado en el Anexo 1). Estos puntos son los que se detallan en los siguientes apartados, incluyendo los cuestionarios utilizados para recoger dicha opinión (cuestionarios completados posteriormente al juego). La tercera y última versión de Articoding ha sido utilizada con estudiantes universitarios de educación, buscando la opinión del juego por parte de futuros profesores que serán los que lo utilicen. De nuevo, esta opinión se recogió con un cuestionario específico que completaron tras jugar a Articoding (Anexo 2).

Con el fin de evaluar la aplicabilidad del juego en el contexto educativo, se recogieron, además de los datos obtenidos mediante cuestionarios, registros detallados de la interacción de los estudiantes con el juego (analíticas). Estas analíticas incluyeron el tiempo empleado en cada nivel, los niveles efectivamente completados y el número de errores cometidos en cada uno. El resumen de los datos de analíticas recogidos, incluyendo una breve indicación de los campos del formato xAPI utilizado para su recogida, se detallan en el Anexo 3. Estos datos cumplieron una doble función. Por un lado, permitieron guiar el proceso de mejora incremental del juego (Figura 1), identificando niveles con una alta tasa de errores o patrones recurrentes de aplicación incorrecta de conceptos, lo que señaló la necesidad de ajustes en el diseño pedagógico de dichos niveles. Por otro lado, los datos relativos a los niveles completados y al tiempo de

dedicación por parte de los estudiantes proporcionaron información objetiva y contrastable sobre el uso del juego en el aula y su viabilidad como herramienta educativa.

Durante las diferentes sesiones, siempre estuvieron presentes un investigador y un profesor por aula. Al principio de la sesión se explicaba la actividad a realizar, indicando que consistía en completar el mayor número de niveles posibles y en la realización en los últimos 10 minutos de un cuestionario. Durante las sesiones, el profesor y el investigador asistían a aquellos estudiantes con problemas reiterados en un nivel, es decir, si pedían ayuda y ya habían realizado varios intentos. En algunos casos, y sobre todo en la primera versión, se comprobó que algunos de estos problemas eran debido al diseño del nivel o a algún problema de usabilidad del juego.

Por último, las modificaciones específicas que se realizaron entre las distintas versiones de Articoding, así como capturas de los cambios en su interfaz se detallan en el Anexo 4.

4. Articoding en el Aula

4.1. Experiencias con estudiantes de secundaria

Las pruebas con estudiantes de 12 a 18 años se realizaron en dos iteraciones, con dos versiones del juego Articoding. La primera versión de Articoding no incluía las categorías de “Introducción” ni “Funciones”, categorías que fueron introducidas para la segunda versión del juego como mejora, teniendo en cuenta los resultados obtenidos en las primeras sesiones de prueba de Articoding. Esta primera versión (año 2021) se probó en dos centros. Por un lado, una escuela de Arte y Diseño en el área de Barcelona y, por otro lado, un colegio de educación secundaria en Madrid. En la aplicación del juego participaron un total de 162 estudiantes.

En estas experiencias en estos dos centros educativos se obtuvieron 142 respuestas válidas de los 162 estudiantes que usaron el juego. Los otros 20 usuarios no llegaron a enviar el cuestionario posterior o no llegaron a jugar un tiempo significativo (más de 20 minutos).

El cuestionario utilizado en estas pruebas puede encontrarse al completo en el Anexo 1. Entre las dos versiones del juego solo se realizaron modificaciones mínimas en el cuestionario para adaptarlo a los cambios entre dichas versiones, que se presentan junto con los resultados.

4.1.2. Experiencias con la primera versión de Articoding

En Barcelona, 28 estudiantes de un ciclo de Gráficos Interactivos completaron 6 sesiones de juego de una hora. Estos estudiantes llegaron a superar todos los niveles disponibles en la primera versión del juego (el juego era más corto al no incluir funciones). Los estudiantes tenían 16 años de media ($SD=0,44$).

Por otro lado, en Madrid se llevaron a cabo 2 pruebas diferentes:

- En la primera se realizaron 2 sesiones de una hora y la completaron 91 estudiantes de secundaria de entre 12 y 14 años. De media, los estudiantes superaron 12,20 niveles ($SD=2,83$) y jugaron durante 86,26 minutos ($SD=30,83$). La mayoría de

los estudiantes superaron la categoría de “Variables” (88 jugadores), pero sólo 25 llegaron a completar la categoría de “Tipos”

- En la segunda se llevó a cabo una única sesión de una hora que completaron 23 estudiantes de secundaria y de 1º de Bachillerato de entre 14 y 18 años. Los estudiantes jugaron durante 37,31 minutos ($SD=3,99$) y completaron 8,43 niveles ($SD=2,5$) de media.

El resumen de estos datos se presenta en la Tabla 1, incluyendo además la mediana y el intervalo de confianza del 95%.

Tabla 1

Resumen de los datos recogidos en las pruebas con la Versión 1 de Articoding.

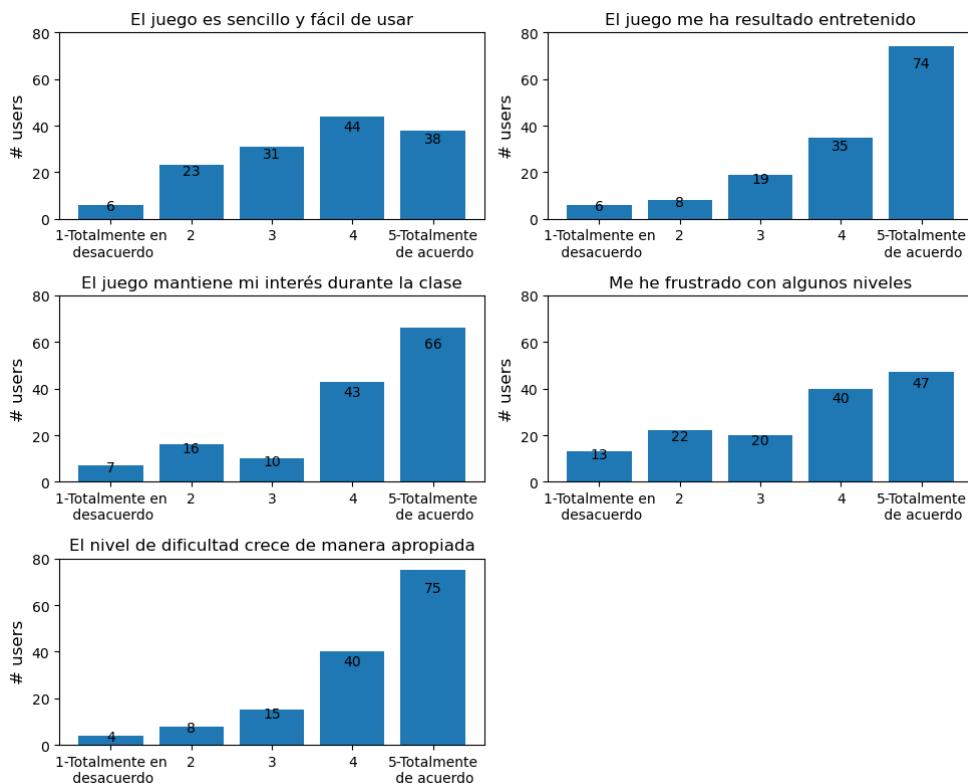
VERSIÓN 1	N	Edad	Niveles superados	Tiempo de juego (minutos)
Sesión 1	91	12-14	M= 12,20 SD=2,83 MDN= 12 IC 95%=[11,63, 12,81]	M= 86,26 SD= 30,83 MDN= 80,1 IC 95%=[80,14, 92,98]
Sesión 2	23	14-18	M= 8,43 SD=2,5 MDN= 9 IC 95%=[7,35, 9,52]	M= 37,31 SD= 3,99 MDN= 36,84 IC 95%=[35,58, 39,03]

Durante estas sesiones se comprobó que parte de los jugadores tenía problemas con el concepto de las variables y su uso. Esto incentivó la creación de una nueva categoría, separando la introducción de las mecánicas de juego y el uso de variables y sus bloques específicos, dando lugar a la categoría de “Introducción”.

Respecto a las afirmaciones planteadas en los cuestionarios aplicados (*El juego es sencillo y fácil de usar; El juego me ha resultado entretenido; El juego mantiene mi interés durante la clase; Me he frustrado con algunos niveles; El nivel de dificultad crece de manera apropiada*), la Figura 7 muestra los resultados, en escala Likert de 5 puntos (siendo 1 – totalmente en desacuerdo con la afirmación, hasta 5 – totalmente de acuerdo con la afirmación).

Figura 7

Resultados de las preguntas Likert para la versión 1 de Articoding ($N=142$).



Además, los cuestionarios incluían cuatro preguntas de múltiples opciones en las que se obtuvieron las siguientes respuestas:

- **¿Te gusta cuando se usan videojuegos en clase para enseñar?**

Para esta pregunta se obtuvieron 141 respuestas válidas, se daban 4 posibles respuestas y se obtuvieron respuestas mayoritariamente positivas: 78 (55%) “Sí, siempre me gusta”, 54 (38%) “Sí, por lo general”. 8 respuestas de “Solo a veces, de manera extraordinaria” y tan solo 1 “No me gusta que se usen en clase”.

- **¿Te gustaría que los profesores usaran más videojuegos en clase?**

Las 141 respuestas fueron mayoritariamente positivas: 124 (88%) “Sí, cualquier tipo de videojuego”, 10 “Sí, pero sólo videojuegos comerciales, que se compran en tiendas” y tan solo 7 “No”.

- **¿Te gustaría que los deberes para hacer en casa pudieran ser pasarse un videojuego o estuvieran relacionados con videojuegos?**

Los resultados fueron 110 (78%) “Sí, cualquier tipo de videojuego”, 14 (10%) “Sí, pero sólo videojuegos comerciales, que se compran en tiendas” y 17 (12%) “No”.

- **¿Quieres jugar más tiempo a este juego para poder completar todos los niveles?**

De las posibles opciones de respuesta, se obtuvieron respuestas afirmativas: 55 “Sí, no me importaría jugar más” y 37 “Sí, tengo muchas ganas” (entre ambas, 65%), y negativas: 23 “No, ya he jugado bastante” y 4 “No, no me ha gustado” (entre ambas, 19%). Las restantes 22 (16%) respuestas fueron “No sé”.

Finalmente, en las cinco preguntas de respuesta abierta, sus respuestas se resumen a continuación.

- **• ¿Has tenido dificultades para resolver los diferentes niveles del juego? ¿Cuáles?**

La mayoría de los participantes (114) contestaron afirmativamente señalando alguna dificultad en el juego. Tan solo 24 participantes respondieron no haber tenido ninguna dificultad. Entre las dificultades concretas encontradas por los jugadores, destacaron comentarios que mencionaban diversos “niveles” del juego (33 respuestas), mencionando en concreto algunos niveles de las categorías “Bucles” (10) o “Variables” (6), por ejemplo “*sí, sobre todo los bucles, han sido los niveles más complicados a los que he llegado*”. También se mencionaron algunos objetos de juego como los “espejos” (8) o las “puertas” (3), por ejemplo “*no he tenido muchas pero el de los espejos era difícil*”.

- **• ¿Qué te ha gustado más del videojuego?**

Entre los aspectos que más han gustado del videojuego, destacan: los “niveles” (36 respuestas), el hecho de ser “entretenido” (12), que haga “pensar” (8) y la “dificultad” que presenta (7) o que es “fácil” (7), con comentarios como, por ejemplo, “*La gran diversidad de niveles, y sus nuevas incorporaciones*” o “*Que los niveles van creciendo de dificultad adecuadamente, y vas aprendiendo a programar poco a poco*”. También hubo menciones a aspectos concretos del juego como el “pingüino” del logo (7) y el que sea una herramienta para “aprender a programar” (11).

- **• ¿Qué no te ha gustado y cambiarías para mejorarlo?**

24 participantes no cambiarían “nada” del juego. El resto mencionan algunos aspectos como: cambios en algunos “niveles” (19), la “velocidad” del juego que en ocasiones se hacía lento (10), la “dificultad” (7), mejoras en las “instrucciones” (9) o en las “pistas” (4), con comentarios como que “*en algunos niveles te quedabas bloqueado y las pistas no eran claras*” o “*la velocidad a la hora de colocar los bloques a veces va un poco lenta*”.

- **• ¿Has probado el modo edición? ¿Qué te parece crear tus propios niveles?**

De las 141 respuestas, 107 (76%) contestaron no haber probado el modo edición, en muchos casos por falta de tiempo. De los 34 participantes que sí respondieron afirmativamente, todos comentaron algo positivo del modo edición incluyendo que lo encontraban “divertido” y/o “entretenido” (23 respuestas), con comentarios como “*Me parece entretenido poder crear tus propios niveles*”. 4 participantes mencionaron haber encontrado algún “fallo” o “mejora” en este modo.

- **¿Cuál es tu opinión sobre el videojuego? (Puedes aprovechar para dejarnos cualquier comentario u opinión que quieras compartir).**

Finalmente, en cuanto a la opinión general sobre el videojuego, destacan adjetivos positivos como que lo encontraron “entretenido” (35 respuestas), “divertido” (14), “bueno” (10), “chulo” (8), “original” (4) o “interesante” (4). También se repiten comentarios como que les ha “gustado mucho” (24), y les ha servido para “aprender” (13) y “pensar” (8), con comentarios como *“Es un juego muy entretenido para agilizar tu mente, además puedes divertirte jugando mientras aprendes”* o *“me ha gustado mucho, me gustaría poder jugar en casa”*.

Además, en ambos casos, los profesores a cargo de las sesiones también manifestaron una opinión positiva hacia la actividad e interés por el juego una vez publicado.

4. 2. Experiencias con la segunda versión de Articoding

La segunda versión de Articoding (años 2022 y 2023) se probó en 2 colegios de la Comunidad de Madrid, y en la Universidad Complutense de Madrid durante dos ediciones del programa 4ESO+Empresa. En este programa estudiantes de 4º de la ESO participan en distintas actividades en empresas o, como fue este caso, en la universidad según sus preferencias o intereses a futuro (en cada edición 40 alumnos de distintos centros de la Comunidad de Madrid). En todos los casos, los participantes eran estudiantes de secundaria. El cuestionario utilizado en este caso puede encontrarse al completo en el Anexo 2.

310 estudiantes de Educación Secundaria Obligatoria participaron en estas experiencias de evaluación con la segunda versión del juego Articoding. Se realizaron 4 sesiones diferentes, donde se obtuvieron 291 respuestas válidas:

- La primera sesión consistió en 2 horas en el mismo centro educativo de la Comunidad de Madrid en el que se probó la versión anterior del juego. En esta ocasión, participaron las clases que no habían participado el año anterior. La edad de los estudiantes era de 14 años de media, variando entre 12 y 18 años. Los 136 usuarios válidos jugaron durante 94,58 minutos ($SD=29,77$) y completaron 21,11 niveles ($SD=5,2$).
- La segunda sesión se realizó con alumnos de diferentes institutos de la Comunidad de Madrid como parte del evento 4ESO+Empresa de 2022. Durante esta sesión de 2 horas, participaron 40 estudiantes de 4º de ESO de diferentes institutos de la Comunidad de Madrid. Los participantes superaron de media 29,6 niveles (5,89) y jugaron 95,88 minutos ($SD=12,46$).
- La tercera sesión se realizó también durante el evento 4ESO+Empresa, en este caso del año 2023 y durante una sesión de únicamente 1 hora. En total se recogieron todos los datos de 36 estudiantes de los 40 que participaron en la sesión. Los estudiantes completaron 19,27 niveles ($SD=6,8$) y jugaron durante 42,63 minutos ($SD=9,23$).
- La última sesión se realizó en otro instituto diferente de la Comunidad de Madrid, donde jugaron 86 estudiantes (81 datos válidos) de secundaria, con edades entre 14 y 16 años. Durante la sesión de 2 horas los estudiantes jugaron 72,93 minutos ($SD=20,65$) y superaron 21,53 niveles ($SD=5,83$).

El resumen de estos datos se presenta en la Tabla 2, incluyendo además la mediana y el intervalo de confianza del 95%.

Tabla 2

Resumen de los datos recogidos en las pruebas con la Versión 2 de Articoding.

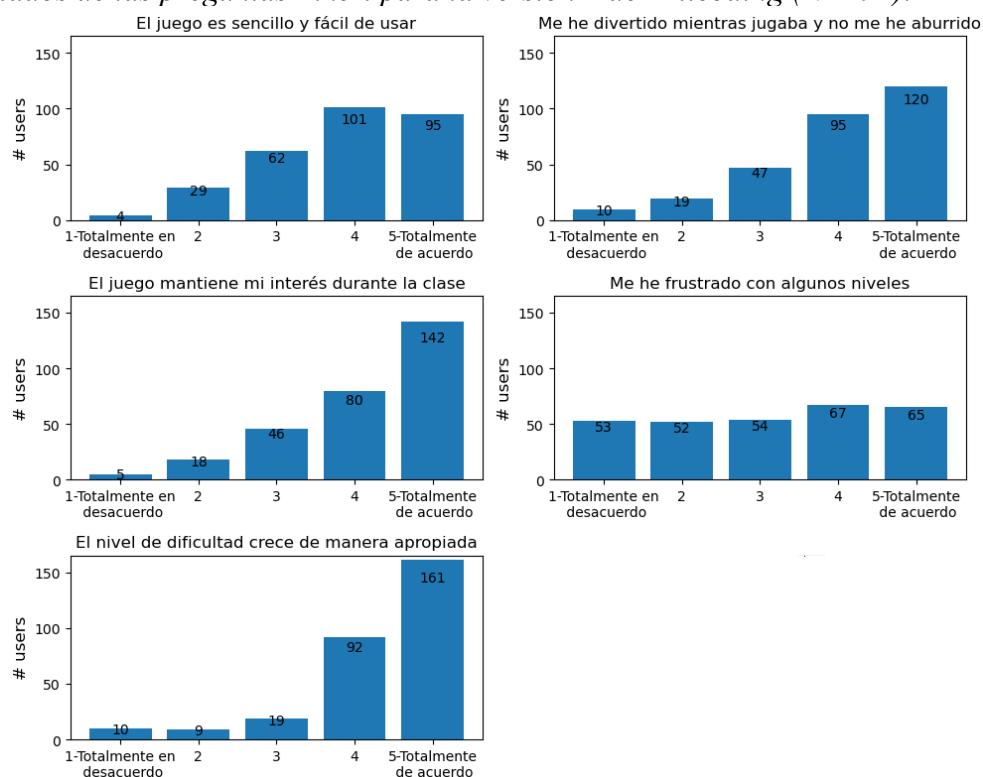
VERSIÓN 2	N	Edad	Niveles superados	Tiempo de juego (minutos)
Sesión 1	136	12-18	M= 21,11 SD= 5,2 MDN= 22 IC 95%=[20,75, 22,66]	M= 94,58 SD= 29,77 MDN= 82,98 IC 95%=[89,47, 99,68]
Sesión 2	40	15-16	M= 28,7 SD= 5,68 MDN= 30 IC 95%=[26,87, 30,60]	M= 95,88 SD= 12,46 MDN= 98,32 IC 95%=[91,79, 99,98]
Sesión 3	36	15-16	M= 19,27 SD= 5,92 MDN= 19 IC 95%=[16,1, 21,17]	M= 43,64 SD= 10,12 MDN= 49,22 IC 95%=[37,55, 46,93]
Sesión 4	81	14-16	M= 21,53 SD= 5,83 MDN= 23 IC 95%=[20,79, 23,50]	M= 72,93 SD= 20,65 MDN= 81,29 IC 95%=[67,43, 76,92]

En estas sesiones se comprobó que los estudiantes avanzaban más rápido que en la versión anterior del juego durante los niveles de variables y tipos, confirmando que había sido útil la modificación en el juego de las categorías “Introducción” y “Variables”. En el mismo tiempo, los jugadores completaban mayor cantidad total de niveles, superando las categorías “Introducción”, “Variables” y “Tipos” en las sesiones de 2 horas.

Respecto a las afirmaciones del cuestionario aplicado (*El juego es sencillo y fácil de usar; Me he divertido mientras jugaba y no me he aburrido; El juego mantiene mi interés durante la clase; Me he frustrado con algunos niveles; El nivel de dificultad crece de manera apropiada*), la Figura 8 muestra los resultados en escala Likert de 5 puntos (siendo 1 – totalmente en desacuerdo con la afirmación, hasta 5 – totalmente de acuerdo con la afirmación).

Figura 8

Resultados de las preguntas Likert para la versión 2 de Articoding ($N=291$).



En cuanto a las preguntas de respuesta abierta, sus respuestas se resumen a continuación.

- **¿Has tenido dificultades para resolver los diferentes niveles del juego? Cuéntanos qué problemas has tenido.**

166 participantes mencionaron alguna dificultad concreta, mientras que 119 contestaron “no” haber tenido ninguna dificultad para resolver los niveles del juego. Sobre las dificultades concretas, destacan algunos “niveles” del juego (65 respuestas), en particular se mencionan algunos de los primeros niveles pertenecientes a las categorías de “Variables” (18), con comentarios como “Sí, algunos niveles en los inicios de aprender a usar los espejos ya que no sabía a qué dirección girarlo para que rebotase de la manera correcta” o “He tenido al principio problemas con entender sobre las variables, en cuanto seguía pasando los niveles entendía mejor y tardaba menos en pasar los niveles”. También se mencionan dificultades para entender algunos elementos el juego como los “espejos” (7).

- **¿Qué te ha gustado más del videojuego?**

Los aspectos más mencionados han sido los “niveles” del juego (52 respuestas) y su “dificultad” (38). También se mencionan elementos concretos del juego como los “espejos” (38), el “láser” (22), el “pingüino” (12), o los “bloques” (8). 50 participantes contestaron que les había gustado “todo”. Otros participantes mencionaron que les había gustado que el juego les hiciera “pensar” (32), “aprender” (30), que fuera “entretenido” (32) y “divertido” (12). Destacan comentarios como por ejemplo “como la dificultad se

eleva con el aumento de los niveles” o “la idea de los lasers y los espejos, me parece muy buena manera de aprender a programar, a mi me ha ayudado”.

- **• ¿Qué no te ha gustado y cambiarías para mejorarlo?**

En cuanto a los aspectos a mejorar, 82 participantes contestaron que no cambiarían “nada”. El resto mencionaron cambios en algunos “niveles” (25 menciones), en particular sobre la categoría “Variables” (12). Harían cambios en cuanto a la “dificultad” del juego (7) y las “explicaciones” o “instrucciones” del mismo (11). También aumentarían su “velocidad” (6) y que fuera menos “repetitivo” (4). Algunos comentarios destacados de esta pregunta son “*algunos niveles que son un poco difíciles*” o “*que algunos niveles son un poco repetitivos, pero es normal, el juego trata de aprederte algo para que no se te olvide*”.

- **• ¿Cuál es tu opinión sobre el videojuego? (Puedes aprovechar para dejarnos cualquier comentario u opinión que quieras compartir)**

Gran parte de los participantes respondieron que el juego estaba “bien” (78) o que les había “gustado” (49). Destacaron que les había parecido “entretenido” (47), “divertido” (32), o “interesante” (13), con comentarios como “*está muy bien, aunque sea para aprender es entretenido*”. También se repitieron menciones a la parte educativa, destacando que el juego les había servido para “aprender” (29) con menciones explícitas a los aspectos de programación que contiene (30), con comentarios como “*La idea de aprender programación jugando y resolviendo niveles lógicos en lugar de hacer aburridas y largas prácticas*”.

- **• ¿Te gustaría tener más tiempo para seguir jugando?**

De las 216 respuestas a esta pregunta, se obtuvieron: 147 (68%) sí, 20 (9%) no, y 49 (23%) no están seguros.

4.3 Experiencias con estudiantes universitarios de educación

La tercera versión de Articoding (2024) se evaluó con 33 estudiantes del máster de educación de la Universidad Autónoma de Madrid (UAM) y con 3 profesores del máster.

La sesión duró 2 horas, en las que se dedicó una a Articoding y a otra a hablar del uso de juegos en el aula. En total se obtuvieron los datos de 31 de los 33 estudiantes que participaron y llegaron a completar 13,77 niveles ($SD=4,88$). Los estudiantes tenían entre 20 y 30 años, y el tiempo de juego medio fue de 41,77 minutos ($SD=8,13$). El resumen de estos datos se presenta en la Tabla 3, incluyendo además la mediana y el intervalo de confianza del 95%.

Tabla 3

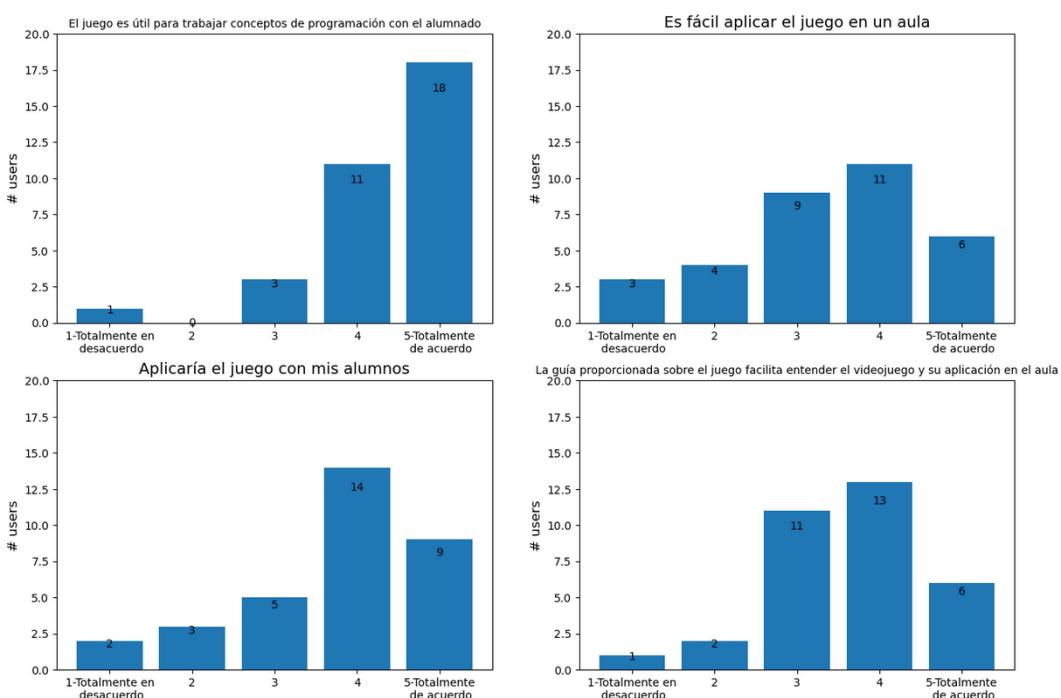
Resumen de los datos recogidos en las pruebas con la Versión 3 de Articoding.

VERSIÓN 3	N	Edad	Niveles superados	Tiempo de juego (minutos)
Sesión 1	31	20-30	M= 13,77 SD= 4,88 MDN= 13 IC 95%=[11,98, 15,56]	M= 41,77 SD= 8,13 MDN= 42,92 IC 95%=[38,79, 44,76]

Respecto a las afirmaciones del cuestionario (*El juego es útil para trabajar conceptos de programación con el alumnado; Es fácil aplicar el juego en un aula; Aplicaría el juego con mis alumnos; La guía proporcionada sobre el juego facilita entender el videojuego y su aplicación en el aula*), la Figura 9 muestra los resultados en escala Likert de 5 puntos (siendo 1 – totalmente en desacuerdo, hasta 5 – totalmente de acuerdo con la afirmación).

Figura 9

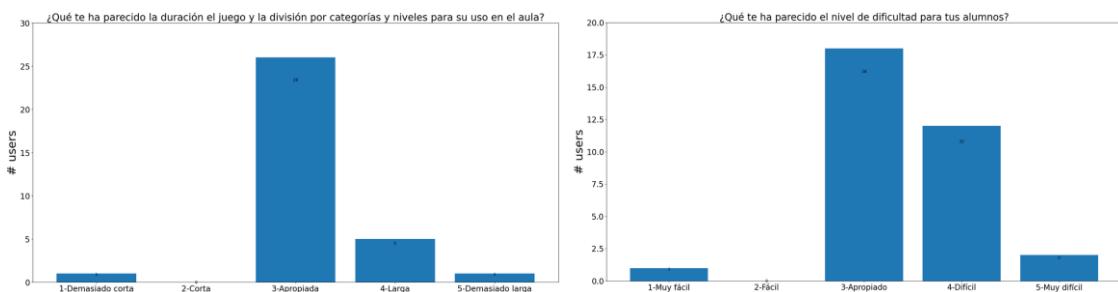
Resultados de las preguntas Likert sobre aplicabilidad de Articoding con estudiantes de educación (N=33).



En cuanto a las preguntas sobre la duración del juego y su dificultad (*¿Qué te ha parecido la duración el juego y la división por categorías y niveles para su uso en el aula? (1 - demasiado corta a 5 - demasiado larga); ¿Qué te ha parecido el nivel de dificultad para tus alumnos? (1 - muy fácil a 5 - muy difícil)*), las respuestas también en escala Likert de 5 puntos se muestran en la Figura 10.

Figura 10

Resultados de las preguntas Likert sobre duración y dificultad de Articoding con estudiantes de educación ($N=33$).



Finalmente, en las preguntas de respuesta abierta se obtuvieron 32 respuesta válidas cuyas ideas principales se resumen a continuación:

- **• ¿Hay algún tipo de bloque u objeto del juego que te haya costado entender su funcionamiento o utilidad? ¿Cuál?**

Destaca por encima del resto el bloque de “Variables” (16 respuestas) como difícil de entender al inicio, con comentarios como “*El bloque de las operaciones con variables. Saber ordenar los distintos bloques y variables para que se ejecute de forma correcta*”, si bien varias respuestas matizan que ha sido “por desconocimiento del tema” o que “sin demasiada dificultad”. En menor medida, se mencionan también algunas dudas respecto al funcionamiento de los “bloques” (3), los “comandos” (2) o los “espejos” (2). 5 participantes contestaron negativamente a esta pregunta, indicando que “no” les costó entender nada.

- **• ¿Qué no te ha gustado, cambiarías o mejorarías para poder facilitar el uso del juego en el aula?**

Sobre este punto, el tema más mencionado, en 14 de las respuestas, se refiere a las “instrucciones” del juego: los usuarios señalan que las extenderían (sobre todo al principio de los niveles), las mejorarían, o permitirían consultarlas en cualquier momento durante el juego, con comentarios como “*Haría las instrucciones más claras y más explicativas*” o añadir “*botón con instrucciones para el juego*”. Relacionado con esto, también algunos participantes mencionan más explicaciones sobre el bloque de Variables (3) y añadirían una opción de ayuda (2) en caso de no saber continuar. 4 participantes mencionaron que el juego se les hizo repetitivo o pesado, mientras que 2 participantes contestaron que no cambiarían nada.

- **• ¿Qué características del juego te ha gustado más para poder hacer uso del videojuego en el aula?**

Los aspectos del juego mencionados son muchos y muy variados, aunque podemos destacar: su “sencillez” o “facilidad” (8 participantes), que es “atractivo” para los estudiantes (5), y “aplicable” en clase (4), o que permite el “trabajo autónomo” por parte de los alumnos (2). Al respecto, aparecen comentarios como “*Que sean comandos fáciles de utilizar. A pesar de que al principio costaba un poco comprender el funcionamiento*

del juego, cuando le cogías el truco era bastante fácil.” o “Que la programación por bloques es más fácil de entender para los alumnos y es mucho más atractivo para ellos que aprender a hacer líneas y líneas de código”. Otros participantes mencionaron el “diseño” del juego, sus “mecánicas”, el “tema”, los “tutoriales”, los “reintentos”, el “progreso” o las “estrellas” obtenidas en el juego, como otros aspectos que les gustaron.

- **¿Te gustaría poder ver informes de las acciones de tus alumnos cuando juegan, los niveles que han completado y otra información relacionada? ¿Qué consideras que deben contener estos informes?**

De los 32 participantes que respondieron, tan solo 1 contestó “no estar seguro”, mientras los otros 31 contestaron afirmativamente. Sobre el contenido de los informes, las respuestas fueron bastante uniformes y destacan: saber qué “niveles han presentado una mayor dificultad” (9 participantes), sobre todo con el objetivo de ayudarles y reforzar esos conocimientos en clase, saber el “número de intentos” que han realizado en cada nivel (7), los “errores” que han cometido (7), el “tiempo empleado” en cada nivel (6) y el “progreso” o alcance total en el juego (5). Un comentario destacado en esta pregunta que engloba varias opiniones es: *“Estaría bien, sobre todo poder sacar gráficos de la evolución o los errores más recurrentes. También estaría interesante si se pudiese desarrollar un apartado en el que agrupar por clase a los alumnos y su interacción en el videojuego para extraer estadísticas de los apartados que más les cuesta para poder reforzar esos contenidos en clase o incidir en cómo hacer más eficiente la programación en el caso de que utilicen bloques de más”*.

- **• ¿Cuál es tu opinión, en resumen, sobre el videojuego y su utilidad educativa?**

En esta pregunta a modo de resumen global, la mayoría de las respuestas (24) fueron únicamente positivas destacando, entre otros aspectos, el valor “educativo” de la herramienta para aplicarla en clase y el ser una herramienta “novedosa”. Destacan comentarios como *“Creo que el videojuego es bastante útil a la hora de utilizarlo en un aula, ya que es algo cotidiano para ellos y al final es una manera entretenida para que aprendan, y muchas aprenden mejor e interiorizan mejor los conceptos jugando”* o *“Está muy entretenido, lo veo muy adecuado para el aula como forma de practicar la programación”*. 5 participantes destacaron que, al ser “profesores de primaria”, no ven el juego adecuado para sus futuros alumnos, como sí lo ven para estudiantes de secundaria. De forma minoritaria se obtuvieron algunos comentarios negativos sobre el juego: sobre su “dificultad”, mejorar las “instrucciones”, o el considerarlo únicamente como herramienta para “relajar a los alumnos”, pero no eficaz para su aprendizaje.

5. Discusión y Conclusiones

Como resultado del trabajo se ha obtenido un juego educativo que permite practicar el pensamiento computacional y aprender conceptos básicos de programación (variables, condiciones, funciones, etc.). Además, se han realizado diversas experiencias en aula con usuarios reales, permitiendo estudiar la aplicación del juego y mejorarlo en diversas iteraciones. Los estudios iniciales realizados con las primeras versiones del videojuego educativo Articoding han arrojado resultados positivos en cuanto a la aceptación del juego, tanto por parte de estudiantes de secundaria (su público objetivo), como por parte

de estudiantes universitarios de educación (futuros profesores y posibles usuarios del juego en sus clases).

En la primera versión del juego, la gran mayoría de estudiantes lo encuentran entretenido, que mantiene su interés y que su dificultad crece de forma apropiada. Se obtuvieron resultados más variados en cuanto a la sencillez del juego, aunque la mayoría también fueron positivos. También la mayoría respondieron que les gustaría continuar jugando. En general, los estudiantes tienen una actitud positiva respecto del uso de videojuegos como herramientas de aprendizaje, como se esperaba por resultados previos (Lopez-Fernandez et al., 2021). Si bien, una mayoría (70 participantes) se han frustrado con algunos niveles, como bien reflejaron en sus respuestas abiertas, donde muchos estudiantes comentaron dificultades en algunos niveles del juego. Estas dificultades venían fundamentalmente del contenido del juego (es decir, de los propios conceptos de programación) y no tanto por dificultades más directamente achacables al diseño del juego.

En la segunda versión del juego, la mayoría de los estudiantes encontraron que el juego es sencillo y fácil de usar, mantiene su interés y se han divertido jugando, y les gustaría tener más tiempo para seguir jugando (aunque más que el aspecto de la diversión lo que nos parece más relevante es que ha logrado mantener el interés de los alumnos). Se obtuvieron resultados muy variados en cuanto a la frustración con algunos niveles, sin embargo, la inmensa mayoría afirmaron que la dificultad de dichos niveles crece de forma apropiada. De nuevo, las dificultades mayoritariamente parecen venir del contenido conceptual de programación implicados en los niveles del juego, si bien los datos de analíticas de interacción combinados con la realimentación de los usuarios nos han permitido identificar algunos aspectos del diseño del juego a mejorar (p. ej., las instrucciones en algunos niveles).

Con estas dos evaluaciones, se ha respondido afirmativamente a las preguntas de investigación RQ1, RQ2, RQ3 y RQ4 ya que los resultados han sido ampliamente positivos en todos los aspectos evaluados. Sin embargo, también se han identificado algunos aspectos a mejorar en cuanto a la dificultad de algunos niveles concretos (RQ3) y algunas mecánicas cuya explicación puede mejorarse (RQ1).

Por último, en la tercera versión de Articoding, los estudiantes de educación encuentran en su mayoría que el juego es útil para trabajar programación y que lo aplicarían en sus clases. En cuanto a la sencillez para hacerlo, las respuestas no son tan claras lo que indica la necesidad de dar apoyo a los profesores, por ejemplo, a través de guías docentes o material complementario que les ayude en la aplicación de estas herramientas. Estos futuros profesores encuentran en su mayoría que tanto duración como nivel de dificultad del juego son apropiados, si bien hay más estudiantes que encuentran que el juego es algo difícil.

Esta evaluación ha servido para dar respuesta a las preguntas de investigación RQ5 y RQ6 de nuevo de forma ampliamente positiva.

Estos prometedores resultados iniciales sobre la aceptación del juego como herramienta educativa tanto por parte de alumnos como de futuros profesores motivan a realizar una validación científica más completa. Con este propósito antes de promover su aplicación

a gran escala en escenarios reales sería deseable validar el aprendizaje que promueve el juego en los alumnos. Para ello, se plantea realizar nuevos estudios con Articoding que incluyan, por un lado, la captura y el análisis de analíticas de aprendizaje para juegos (GLA) para poder estudiar las diferencias según el tipo de jugador y sus conocimientos iniciales y, por otro lado, el uso de cuestionarios previo y posterior al juego para medir el cambio en el aprendizaje motivado por el juego en cuanto a programación y pensamiento computacional. Esta evaluación más dirigida al aprendizaje concreto de los alumnos sería posible mediante alguna de las pruebas estandarizadas y previamente validadas (Román-González et al., 2017) o mediante la propia prueba que se publique en PISA 2025 (u otra similar derivada de la misma). Además, este enfoque basado en datos de aprendizaje permitiría estudiar la efectividad de Articoding en distintos escenarios (con clase previa de introducción o sin ella) y su comparación con otras herramientas existentes (p. ej., tutores de IA generativa). Por otra parte, las analíticas de aprendizaje recogidas en formato xAPI podrían utilizarse para proporcionar informes detallados al docente sobre la progresión y resultados de sus estudiantes en el juego.

Una vez validado, y desarrollada alguna herramienta complementaria de apoyo al profesorado (p. ej., una guía de uso del juego sobre su aplicabilidad en clases), se podría proceder al despliegue en abierto de Articoding como herramienta de aprendizaje a utilizar por profesores en sus clases de educación secundaria para fomentar el aprendizaje de conceptos básicos de programación y el desarrollo del pensamiento computacional.

Por último, hay que resaltar algunas de las limitaciones que tiene el juego, como son el uso de la característica de comunidad, que aún no ha sido validado y su uso depende de tener servidores dedicados para el almacenamiento y compartición de los niveles creados. Además, el juego está pensado para su uso individual, en ordenadores con Windows y los centros educativos deben tener suficientes dispositivos para su aplicación. La exportación de versiones para otras plataformas (versiones web y móvil) y para dispositivos con el sistema operativo Linux aumentaría aún más el porcentaje de dispositivos en los que podría desplegarse. Además, una versión web permitiría saltarse el paso de instalación, simplificando el proceso de uso en el aula.

Agradecimientos

Queremos agradecer el trabajo de desarrollo de los alumnos de la Facultad de Informática en general y de la asignatura de Juegos Serios en particular que han participado activamente (en algunos casos en sus trabajos de fin de grado) en el desarrollo y validación inicial (*beta-testing*) de Articoding. Sus nombres se pueden encontrar en el repositorio del código del proyecto (que tiene licencia abierta).

Presentación del artículo: 28 de marzo de 2025

Fecha de aprobación: 15 de noviembre de 2025

Fecha de publicación: 1 de enero de 2026

Calvo-Morata, A., Alonso-Fernández, C., Martínez-Ortiz, I., & Fernández-Manjón, B. (2026). Metodología para el diseño de juegos serios: Articoding, un juego para desarrollar el pensamiento computacional a través de la programación. *RED. Revista de Educación a Distancia*, 26(83). <http://dx.doi.org/10.6018/red.662331>

Financiación

Este trabajo ha sido parcialmente financiado por el Ministerio de Educación (PID2020-119620RB-I00; PID2023-149341OB-I00) y por la Cátedra Telefónica-Complutense de Educación Digital y Juegos Serios.

Declaración del autor o de los autores sobre el uso de LLM

Este artículo no ha utilizado para su redacción textos provenientes de un LLM (ChatGPT u otros).

Declaración de las contribuciones de los autores

Conceptualización, B.F. y A.C.; curación de datos, C.A. y A.C.; software, A.C. e I.M.; análisis formal, C.A. y A.C.; obtención de fondos, B.F. e I.M.; investigación, A.C. y B.F.; metodología, A.C. y B.F.; administración del proyecto, B.F.; recursos, B.F. e I.M.; supervisión, B.F. e I.M.; validación, A.C. y C.A.; visualización, C.A.; redacción – borrador original, C.A. y A.C.; redacción – revisión y edición, C.A., A.C. y B.F.

Referencias

- Anderson, N. D. (2016). A Call for Computational Thinking in Undergraduate Psychology. *Psychology Learning & Teaching*, 15(3), 226–234. <https://doi.org/10.1177/1475725716659252>
- Basogain, X., Olabe, M. Á., Olabe, J. C., & Rico, M. J. (2018). Computational Thinking in pre-university Blended Learning classrooms. *Computers in Human Behavior*, 80, 412–419. <https://doi.org/10.1016/J.CHB.2017.04.058>
- Bile, A. (2022). Development of intellectual and scientific abilities through game-programming in Minecraft. *Education and Information Technologies*, 27(5), 7241–7256. <https://doi.org/10.1007/s10639-022-10894-z>
- Bocconi, S., Chiocciello, A., Kampylis, P., Dagienè, V., Wastiau, P., Engelhardt, K., Earp, J., Horvath, M., Jasutè, E., Malagoli, C., Masiulionytè-Dagienè, V., & Stupurienè, G. (2022). *State of play and practices from computing education REVIEWING COMPUTATIONAL THINKING IN COMPULSORY EDUCATION*.
- Brackmann, C. P., Román-González, M., Robles, G., Moreno-León, J., Casali, A., & Barone, D. (2017). Development of Computational Thinking Skills through Unplugged Activities in Primary School. *Proceedings of the 12th Workshop on Primary and Secondary Computing Education*, 65–72. <https://doi.org/10.1145/3137065.3137069>
- Calvo-Morata, A., Alonso-Fernández, C., Santilario-Berthilier, J., Martínez-Ortiz, I., & Fernández-Manjón, B. (2025). Learning Analytics to Guide Serious Game Development: A Case Study Using Articoding. *Computers*, 14(4), 122. <https://doi.org/10.3390/computers14040122>
- Calvo-Morata, A., Humble, N., Mozelius, P., Pechuel, R., & Fernández-Manjón, B. (2024). Games for Coding to Attract New Students to STEM. *2024 IEEE Global Engineering Education Conference (EDUCON)*, 01–08. <https://doi.org/10.1109/EDUCON60312.2024.10578626>
- Fraillon, J., & Rožman, M. (2023). *IEA International Computer and Information Literacy Study 2023 - ASSESSMENT FRAMEWORK*.

- Hosein, A. (2019). Girls' video gaming behaviour and undergraduate degree selection: A secondary data analysis approach. *Computers in Human Behavior*, 91, 226–235. <https://doi.org/10.1016/j.chb.2018.10.001>
- Hsu, T. C., Chang, S. C., & Hung, Y. T. (2018). How to learn and how to teach computational thinking: Suggestions based on a review of the literature. *Computers & Education*, 126, 296–310. <https://doi.org/10.1016/J.COMPEDU.2018.07.004>
- Jie Du, & Hayden Wimmer. (2019). Hour of Code: A Study of Gender Differences in Computing. *Information Systems Education Journal*. <http://www.edsigcon.org>
- Liu, T. (2024). Assessing implicit computational thinking in game-based learning: A logical puzzle game study. *British Journal of Educational Technology*, 55(5), 2357–2382. <https://doi.org/10.1111/bjet.13443>
- López Solórzano, J. G., & Ángel Rueda, C. J. (2023). Revisión sistemática de los entornos digitales inmersivos tridimensionales en la enseñanza de la programación. *Revista de Educación a Distancia (RED)*, 23(73). <https://doi.org/10.6018/red.540731>
- Lopez-Fernandez, D., Gordillo, A., Alarcon, P. P., & Tovar, E. (2021). Comparing Traditional Teaching and Game-Based Learning Using Teacher-Authored Games on Computer Science Education. *IEEE Transactions on Education*, 64(4), 367–373. <https://doi.org/10.1109/TE.2021.3057849>
- Miljanovic, M. A., & Bradbury, J. S. (2018). A review of serious games for programming. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 11243 LNCS, 204–216. https://doi.org/10.1007/978-3-030-02762-9_21/COVER
- Organisation for Economic Co-operation and Development. (2023). *PISA 2025 Learning in the Digital World Assessment Framework - Second Draft*.
- Ortega-Rodríguez, P. J. (2025). PISA 2022. Predictores del rendimiento en pensamiento computacional en Educación Secundaria en España. *Revista de Educación a Distancia (RED)*, 25(81). <https://doi.org/10.6018/red.600641>
- Román-González, M., Pérez-González, J.-C., & Jiménez-Fernández, C. (2017). Which cognitive abilities underlie computational thinking? Criterion validity of the Computational Thinking Test. *Computers in Human Behavior*, 72, 678–691. <https://doi.org/10.1016/j.chb.2016.08.047>
- Sage, K., Jackson, S., Mauer, L., & Stockdale, K. (2022). Equal in effectiveness but not yet perception: smartphones and laptops for completing brief academic tasks. *Educational Media International*, 59(2), 112–130. <https://doi.org/10.1080/09523987.2022.2101203>
- Shute, V. J., Sun, C., & Asbell-Clarke, J. (2017). Demystifying computational thinking. *Educational Research Review*, 22, 142–158. <https://doi.org/10.1016/J.EDUREV.2017.09.003>
- Sigayret, K., Tricot, A., & Blanc, N. (2022). Unplugged or plugged-in programming learning: A comparative experimental study. *Computers & Education*, 184, 104505. <https://doi.org/10.1016/j.compedu.2022.104505>
- Steven Saphore. (2024). We looked at all the recent evidence on mobile phone bans in schools – this is what we found. *The Conversation*.
- Varghese, V. V. V., & Renumol, V. G. (2023). Video games for assessing computational thinking: a systematic literature review. *Journal of Computers in Education*, 1–46. <https://doi.org/10.1007/S40692-023-00284-W/FIGURES/2>
- Varghese, V. V. V., & Renumol, V. G. (2024). Video games for assessing computational thinking: a systematic literature review. *Journal of Computers in Education*, 11(3), 921–966. <https://doi.org/10.1007/s40692-023-00284-w>

- Wing, J. M. (2006). Computational thinking. *Communications of the ACM*, 49(3), 33–35.
<https://doi.org/10.1145/1118178.1118215>
- Zapata-Ros, M. (2015). Pensamiento computacional: Una nueva alfabetización digital
Computational Thinking: A New Digital Literacy. In RED. Revista de Educación a
Distancia. Núm (Vol. 46). <http://www.um.es/ead/red/46>
- Zhao, W., & Shute, V. J. (2019). Can playing a video game foster computational thinking
skills? *Computers & Education*, 141, 103633.
<https://doi.org/10.1016/J.COMPEDU.2019.103633>

Anexo 1. Cuestionario para estudiantes

Pregunta	Posibles valores de respuesta
El juego es sencillo y fácil de usar	Escala Likert de 5 puntos (1 totalmente en desacuerdo a 5 totalmente de acuerdo)
El juego me ha resultado entretenido (Me he divertido mientras jugaba y no me he aburrido)	Escala Likert de 5 puntos (1 totalmente en desacuerdo a 5 totalmente de acuerdo)
El juego mantiene mi interés durante la clase	Escala Likert de 5 puntos (1 totalmente en desacuerdo a 5 totalmente de acuerdo)
Me he frustrado con algunos niveles	Escala Likert de 5 puntos (1 totalmente en desacuerdo a 5 totalmente de acuerdo)
El nivel de dificultad crece de manera apropiada	Escala Likert de 5 puntos (1 totalmente en desacuerdo a 5 totalmente de acuerdo)
¿Te gusta cuando se usan videojuegos en clase para enseñar?	“Sí, siempre me gusta”, “Sí, por lo general”, “Solo a veces, de manera extraordinaria”, “No me gusta que se usen en clase”
¿Te gustaría que los profesores usaran más videojuegos en clase?	“Sí, cualquier tipo de videojuego”, “Sí, pero sólo videojuegos comerciales, que se compran en tiendas”, “No”
¿Te gustaría que los deberes para hacer en casa pudieran ser pasarse un videojuego o estuvieran relacionados con videojuegos?	“Sí, cualquier tipo de videojuego”, “Sí, pero sólo videojuegos comerciales, que se compran en tiendas”, “No”
¿Quieres jugar más tiempo a este juego para poder completar todos los niveles?	“Sí, tengo muchas ganas”, “Sí, no me importaría jugar más”, “No, ya he jugado bastante”, “No, no me ha gustado”, “No sé”
¿Has tenido dificultades para resolver los diferentes niveles del juego? ¿Cuáles?	Respuesta abierta
¿Qué te ha gustado más del videojuego?	Respuesta abierta
¿Qué no te ha gustado y cambiarías para mejorarlo?	Respuesta abierta
¿Has probado el modo edición? ¿Qué te parece crear tus propios niveles?	Respuesta abierta
¿Cuál es tu opinión sobre el videojuego? (Puedes aprovechar para dejarnos cualquier comentario u opinión que quieras compartir).	Respuesta abierta
(En los casos que tenga sentido) ¿Te gustaría tener más tiempo para seguir jugando?	Respuesta abierta

Anexo 2. Cuestionario para profesores

Pregunta	Posibles valores de respuesta
El juego es útil para trabajar conceptos de programación con el alumnado	Escala Likert de 5 puntos (1 totalmente en desacuerdo a 5 totalmente de acuerdo)
Es fácil aplicar el juego en un aula	Escala Likert de 5 puntos (1 totalmente en desacuerdo a 5 totalmente de acuerdo)
Aplicaría el juego con mis alumnos	Escala Likert de 5 puntos (1 totalmente en desacuerdo a 5 totalmente de acuerdo)
La guía proporcionada sobre el juego facilita entender el videojuego y su aplicación en el aula	Escala Likert de 5 puntos (1 totalmente en desacuerdo a 5 totalmente de acuerdo)
¿Qué te ha parecido la duración el juego y la división por categorías y niveles para su uso en el aula?	Escala Likert de 5 puntos (1 demasiado corta a 5 demasiado larga)
¿Qué te ha parecido el nivel de dificultad para tus alumnos?	Escala Likert de 5 puntos (1 muy fácil a 5 muy difícil)
¿Hay algún tipo de bloque u objeto del juego que te haya costado entender su funcionamiento o utilidad? ¿Cuál?	Respuesta abierta
¿Qué no te ha gustado, cambiarías o mejorarías para poder facilitar el uso del juego en el aula?	Respuesta abierta
¿Qué características del juego te ha gustado más para poder hacer uso del videojuego en el aula?	Respuesta abierta
¿Te gustaría poder ver informes de las acciones de tus alumnos cuando juegan, los niveles que han completado y otra información relacionada? ¿Qué consideras que deben contener estos informes?	Respuesta abierta
¿Cuál es tu opinión, en resumen, sobre el videojuego y su utilidad educativa?	Respuesta abierta

Anexo 3. Resumen de datos de interacción recogidos

Interacción recogida	Detalles del formato xAPI	Información que proporciona
El jugador inicia el juego	<i>initialized serious-game</i> con marca de tiempo (<i>timestamp</i>)	Tiempo total empleado en el juego por el jugador
El jugador finaliza el juego	<i>completed serious-game</i> con marca de tiempo (<i>timestamp</i>)	
El jugador interactúa con un botón	<i>Interacted</i> con identificador del botón y marca de tiempo (<i>timestamp</i>)	Interacciones con los menús de juego y si entra o sale de los diferentes modos de juego
El jugador inicia un nivel	<i>initialized level</i> con identificador del nivel y marca de tiempo (<i>timestamp</i>)	Tiempo total empleado en cada nivel por el jugador
El jugador finaliza un nivel	<i>completed level</i> con identificador del nivel y marca de tiempo (<i>timestamp</i>)	
El jugador ejecuta el código en un nivel	<i>completed level</i> con identificador del nivel y extensión <i>result</i> con campo <i>success</i> (verdadero o falso) y campo <i>code</i> con el código en XML que ha ejecutado	Soluciones ejecutadas con o sin éxito, código y errores cometidos por cada nivel para cada jugador