

Enseñanza de las Matemáticas a través del Uso de Scratch (Transversalidad STEM)

Antonio de la Hoz Ruiz, Raquel Hijón Neira

Universidad Rey Juan Carlos
España

a.delahoz.2021@alumnos.urjc.es, raquel.hijon@urjc.es

Resumen: En este artículo se lleva a cabo una revisión sistemática cualitativa para mostrar el estado de la cuestión sobre el uso de Scratch para el aprendizaje de las Matemáticas y describir el impacto que tiene tanto en el rendimiento académico en la enseñanza de las Matemáticas como en la actitud que presentan los alumnos a la hora de afrontar la enseñanza de las Matemáticas.

Para ello se analizan un total de 112 artículos sobre investigación en la aplicación o uso de Scratch en la enseñanza de la base de datos científicos recogiendo 81 documentos de la Colección Principal de WOS y 31 de la base de datos de SCOPUS y acotando los criterios de búsqueda a los últimos diez años (2012-2021) y centrándonos en las etapas educativas correspondientes a educación primaria y educación secundaria.

Con el fin de conseguir abarcar las principales características de una revisión sistemática, se seguirán las distintas secuencias propuestas en el protocolo PRISMA.

Los resultados permiten fundamentar el impacto educativo de lenguajes de programación basados en bloques en la enseñanza de las Matemáticas y muestran los beneficios que presentan programar actividades con Scratch en el campo de la resolución de problemas matemáticos abriendo la puerta a enfoques interdisciplinarios entre conceptos computacionales y matemáticos. Adicionalmente, se muestran las limitaciones que deben tenerse en cuenta en futuros estudios para optimizar la integración de Scratch en la enseñanza de Matemáticas.

Palabras clave: aplicación de Scratch, Enseñanza de Matemáticas, Pensamiento Computacional (PC), Tecnología Información y Comunicación (TIC), Ciencia Tecnología Ingeniería Matemáticas (CTIM), Innovación Educativa.

Abstract: In this article, a systematic qualitative review is carried out to show the status of the issue regarding the use of Scratch for Mathematics learning and to describe the impact it has on academic performance in the teaching of Mathematics as well as on the attitude of students to mathematics at the time of facing the learning process.

For this purpose, a total of 112 articles on research into the application or use of Scratch in the teaching of the scientific database are analysed, collecting 81 documents from the Main Collection of WOS and 31 from the database of SCOPUS and limiting the search criteria to the last ten years (2012-2021) and focusing on the educational stages corresponding to primary and secondary education.

In order to cover the main features of a systematic review, the different sequences proposed in the PRISMA protocol will be followed.

The results allow us to substantiate the educational impact of block-based programming languages in the teaching of Mathematics and show the benefits of programming activities with Scratch in the field of mathematical problem solving, opening the door to interdisciplinary approaches between computational and mathematical concepts. In addition, the limitations to be considered in future studies to optimize the integration of Scratch in Mathematics teaching are shown.

Key words: Scratch application, Mathematics Teaching, Computational Thinking (PC), Information and Communication Technology (ICT), Science Technology Mathematical Engineering (CTIM), Educational Innovation.

1. Introducción

En el siglo XXI, la integración de la tecnología de la información y las comunicaciones en los planes de estudios educativos se ha convertido en una tendencia importante y no cabe duda de que los conocimientos y habilidades tecnológicas se han asentado como un elemento fundamental en el desarrollo de las carreras de los estudiantes.

La sociedad actual solicita una ciudadanía con extensos conocimientos tecnológicos en áreas que evolucionan a gran velocidad cómo, por ejemplo, es el caso de las Matemáticas.

En consecuencia, aprender habilidades para resolver problemas y desarrollar el pensamiento lógico, abordar problemas de la vida real y **familiarizarse** con tareas matemáticas abstractas, creando así una base matemática sólida es un objetivo fundamental que cubrir en la formación de los estudiantes de hoy en día.

Hoy en día, la competencia digital está convirtiéndose en una de las competencias esenciales a cubrir en las aulas y una de las herramientas más importantes que favorecen el desarrollo de esta capacidad es la programación, entendiéndose como un recurso para diseñar o crear cosas nuevas en el mundo digital, solucionar problemas, desarrollar nuevas ideas y usar la creatividad.

La programación aporta beneficios en diferentes campos de la educación secundaria, en cuanto a

motivación, competencia digital y habilidades para crear, pudiéndose introducir en la enseñanza de distintas áreas a través de lenguajes visuales basados en bloques como Scratch.

Por todo esto, la inclusión de los lenguajes de programación es fundamental para la completa adquisición de la competencia digital y al pensamiento computacional (PC) y el uso adecuado de las tecnologías de la información y comunicación.

Sin embargo, se observa una carencia de investigaciones sobre la aplicación de lenguajes de programación para la resolución de problemas matemáticos y juegos de programación que mejoren la motivación de los estudiantes en el aprendizaje de las matemáticas.

1.1. Justificación

La comunidad científica destaca la relevancia de los análisis bibliográficos para conocer el estado de la cuestión en diferentes áreas de conocimiento y su divulgación con índices de calidad establecidos.

En el ámbito de la educación de las Matemáticas, el proceso de implantación de las Tecnologías de la Información y de la Comunicación (TIC) y más en particular, el uso de un lenguaje de programación en la enseñanza de la resolución de problemas matemáticos está siendo largo, pero se está logrando implementar y,

así fortalecer una relación que conlleva a “entender Matemáticas” por encima de “hacer matemáticas”.

En la actualidad, educar no significa únicamente mirar la legislación vigente en educación y desarrollar con ello ejercicios, tareas y exámenes sin ninguna clase de fundamento. Educar va un paso más allá y abarca, además, la integración de recursos digitales en los centros. Entre ellos se puede encontrar el uso de software de matemáticas, que favorecen el aprendizaje significativo y la proactividad de los alumnos hacia el uso de las nuevas tecnologías.

Estas metodologías dinámicas están relacionadas con el proceso de aprender a aprender que a su vez posee contacto estrecho con la competencia denominada por el mismo nombre y que es encargada de promover la autonomía responsable del futuro profesional, además de dotar al individuo de habilidades que le servirán para una óptima adaptación a la sociedad y a los cambios que en ella se produzcan (Rodríguez et al., 2019).

Esto, aplicado al ámbito de las matemáticas abre un gran abanico de posibilidades y de formas para enriquecer y facilitar el aprendizaje de conocimientos en los alumnos. Pudiéndose constatar mediante diferentes estudios los beneficios que esta posee en aquellos estudiantes que han presentado más dificultad en el aprendizaje matemático.

Teniendo en cuenta este contexto, se plantea conocer cuál es el estado de la cuestión de la investigación sobre el uso de Scratch para el aprendizaje de las

Matemáticas. En definitiva, el objetivo de este estudio es evaluar el impacto de un programa educativo basado en el uso de Scratch tanto en la enseñanza de las matemáticas como en las actitudes que presentan los alumnos a la hora de afrontar la enseñanza de las matemáticas.

1.2. Pensamiento Computacional

Se podría definir el pensamiento computacional como “el proceso de pensamiento involucrado en la formulación de problemas y sus soluciones, de modo que las soluciones se representan en una forma que pueda llevarse a cabo eficazmente por un agente de procesamiento de información” (Cuny, Snider y Wing, 2010). En otro artículo se propuso que la tecnología se debería utilizar como una herramienta de construcción de conocimiento y los alumnos deberían aprender con la tecnología en lugar de aprender de ella (Jonassen, 1995).

Más específicamente, el pensamiento computacional se define como la capacidad para resolver problemas, diseñar sistemas y comprender el comportamiento humano a partir de diferentes conceptos y procesos informáticas (Wing, 2008; Disessa 2000; Asbell-Clarke, 2017).

A comienzos del siglo XXI, ya se comenzó a predecir que el pensamiento computacional sería fundamental en las próximas competencias de los estudiantes (Wing, 2008). Aunque el pensamiento computacional es un concepto en auge, la idea de desarrollar

pensamiento computacional en la educación fue introducida por primera vez por Papert en uno de sus artículos (Papert, 1980).

Estudios anteriores han demostrado que las actividades basadas en computación tienen efectos positivos en las actitudes de aprendizaje y en el rendimiento de los estudiantes en el campo de las matemáticas. (Kenneth, 1996).

Independientemente de las dificultades para integrar adecuadamente el pensamiento computacional en las secuencias de enseñanza, diferentes estudios han evidenciado los resultados beneficiosos que puede conllevar el aprendizaje de conceptos computacionales (e.g. Garneli & Chorianopoulos, 2018; Lye & Koh, 2014).

Así, la adquisición de pensamiento computacional se asocia a una mejora en la capacidad de entender y resolver problemas cotidianos relacionados con todas las áreas de aprendizaje (Chen et al., 2017). Además, Wing destaca los beneficios educativos de poder pensar computacionalmente debido al uso de abstracciones y habilidades de razonamiento, que potencian y refuerza las habilidades intelectuales, y por tanto, son transferibles a diferentes dominios (Wing, 2014).

Pensamiento Computacional & Matemáticas

A lo largo de los años, ha habido un aumento en el uso de software computacional y de simulación en todas las profesiones de ciencia, tecnología, ingenierías y matemáticas (STEM) (Wing, 2008).

Los estudios relacionados con el pensamiento computacional en la educación se remontan a la década de 1960, cuando se desarrolló, por primera vez, el lenguaje de programación Logo (Feurzeig & Papert, 2011). El objetivo principal del lenguaje era ayudar a los estudiantes a aprender mediante pensamientos matemáticos y lógicos. Aunque el lenguaje Logo fue visto como un vehículo potencial para transformar la educación, no tuvo éxito en el cumplimiento de este objetivo y, finalmente, perdió relevancia en las escuelas convencionales de Estados Unidos y Reino Unido (Agalianos, Noss y Whitty, 2001).

De hecho, numerosos artículos han estudiado la relación entre los juegos basados en computación y el aprendizaje de las matemáticas. Algunos de ellos, demostraron que los estudiantes que jugaban a video juegos educativos pre-construidos presentaban una mejora en las actitudes y habilidades relacionadas con el aprendizaje de las matemáticas. (p. ej. Kebritchi, Hirumi y Bai, 2010; Shin, Sutherland, Norris y Soloway, 2012).

En un estudio con estudiantes de nueve años, se identificó una asociación entre las actividades destinadas a desarrollar el pensamiento computacional en los estudiantes y una mejora relevante en la capacidad de los alumnos para dividir un problema en pasos más sencillos, o subproblemas, y también en la capacidad de crear algoritmos de niveles más complejos (Mannila et al., 2014).

1.3. Lenguajes de Programación para la enseñanza

Entre otros, el libro “Piensa y Programa con Scratch en casa y en clase” (Bacelo Polo A., Hijón Neira R., Pérez Marín D., 2018) nos muestra la existencia sobre el uso de los lenguajes de programación en la aplicación de la enseñanza educativa.

El primer lenguaje de programación utilizado con fines educativos en los niveles de primaria fue LOGO (Feurzeig, Papert, & Lawyer, 2011). Se consideraba que el uso de LOGO ayudaría a los estudiantes a desarrollar su comprensión al usar este lenguaje de programación como “un instrumento diseñado para ayudar a cambiar la forma de percibir las matemáticas” (Papert, 1990).

Después de la aparición de LOGO, surgieron una gran cantidad de lenguajes de programación, muchos de ellos inspirados en LOGO.

Como en el caso de Logo, el lenguaje SET Interactivo se ha utilizado como recursos de aprendizaje en la asignatura de Matemáticas. Adicionalmente, se argumentó que un gran número de personas están dispuestas a aprender a programar, mientras que un número mucho menor tiene éxito en el aprendizaje de las Matemáticas (Dubinsky, 1995). En base a esto, decidió utilizar el poder del pensamiento computacional para enseñar Matemáticas, de modo que la instrucción que utiliza el lenguaje SET Interactivo se desarrolló con base en un enfoque constructivista del aprendizaje matemático: la teoría APOS (Acción, Proceso, Objeto y Esquema).

Como resultado de las lecciones aprendidas del lenguaje Logo, el entorno de programación Scratch se desarrolló en el MIT (Resnick, 2012).

Scratch

Scratch es un entorno de programación que se supone que tiene “piso bajo” (fácil de aprender) y un “techo alto” (permite a los usuarios construir proyectos relativamente complejos) y que permite a los alumnos escribir código uniendo bloques de codificación visual (Papert, 1993).

Además, Scratch es un entorno de programación visual destinado a mejorar el desarrollo de la fluidez tecnológica de los alumnos y fomentar el pensamiento computacional. Fue diseñado para estudiantes de todas las edades, orígenes e intereses (Maloney, Resnick, Rusk, Silverman y Eastmond, 2010).

Algunas investigaciones realizadas en los últimos años han propuesto que el entorno de aprendizaje de los estudiantes debe ser diseñado y organizado teniendo en cuenta las matemáticas en las actividades de programación de Scratch (Grover, Pea, & Cooper, 2015; Han, Bae, & Park, 2016) incluyendo un lenguaje de programación procedimental para la resolución de problemas matemáticos, como Scratch y juegos de programación para mejorar la motivación de los estudiantes en el aprendizaje de las matemáticas (Mladenovic, Krpan, & Meladenovic, 2016).

En otro artículo, se analizó los impactos de Scratch en el aprendizaje y la motivación de los alumnos de

primaria en relación con las coordenadas cartesianas. Los hallazgos del estudio sugirieron que la motivación en el aprendizaje y los logros de estudiantes de primaria se podrían mejorar con el uso de Scratch (Jianzhi, 2011).

Mediante un experimento con alumnos, se encontró que Scratch podría alentar a los estudiantes a estudiar de manera cooperativa y utilizar procesos de resolución de problemas a la hora de aprender matemáticas (Taylor, Harlow y Forret, 2010).

Varios estudios apuntan al beneficio en el desarrollo del pensamiento matemático como resultado del trabajo con Scratch (Calao, Moreno – León, Correa & Robles, 2015; Calder, 2010; Marmolejo & Campos, 2013). Según de Morais, Basso y Fagundes, el PC es un campo especialmente apropiado para fomentar el aprendizaje matemático en los estudiantes debido a la superposición entre ambos dominios (de Morais, Basso y Fagundes, (2017).

Además, el entorno de codificación Scratch puede alentar a los estudiantes jóvenes a buscar nuevas representaciones de ideas y relaciones matemáticas (Hughes, Gadanidis y Yiu, 2017). De la misma manera, se publicó un artículo donde se describía cómo los estudiantes mejoraban las competencias matemáticas durante la construcción de aplicaciones en Scratch (Calder, 2010). En particular, el uso de Scratch promovió el empleo de habilidades críticas, metacognitivas y reflexivas, que están estrechamente relacionadas con las matemáticas.

Como consecuencia de todo ello, Scratch destaca como uno de los lenguajes de programación más populares y estudiados. Fue creado para ayudar a los estudiantes a aprender a pensar creativamente, razonar sistemáticamente y trabajar en colaboración (Brennan, Balch y Chung, 2014).

En relación con esto, se realizó un estudio con estudiantes y observaron que todos los participantes pudieron aprender programación básica mediante el uso de Scratch en un período de tiempo muy corto (Funke, Geldreich y Hub Wieser, 2017). Adicionalmente, existen varios estudios que apuntan al beneficio en el desarrollo del pensamiento matemático como resultado del aprendizaje basado en Scratch (Cala, Moreno – León, Correa, & Robles, 2015; Calder, 2010; Marmolejo & Campos, 2013). El pensamiento computacional es un campo muy apropiado para fomentar el aprendizaje matemático de los estudiantes debido a la superposición entre ambos dominios (de Morais, Basso y Fagundes, 2017), Por otro lado, el entorno de codificación de Scratch puede animar a los jóvenes estudiantes a buscar nuevas representaciones de ideas y relaciones matemáticas (Hughes, Gadanidis, & Yiu, 2017).

Por último, las actividades matemáticas con Scratch permiten a los profesores a adaptar su estilo de enseñanza a las características individuales de los alumnos y, al mismo tiempo, tratar no solo conocimientos matemáticos sino también el pensamiento computacional (Benton, Hoyles, Kalas y Noss, 2017).

2. Objetivos

Los objetivos de este trabajo serán divididos según sean objetivos generales u objetivos específicos. Los objetivos deben ser expresados con claridad, para así evitar posibles desviaciones en el transcurso de la investigación (Gómez, 2015).

En esta investigación se plantea, como objetivo general, conocer cuál es el estado de la cuestión de la investigación sobre el uso de Scratch para el aprendizaje de las Matemáticas en las etapas de Educación Primaria y Educación Secundaria. En definitiva, el objetivo de este estudio es evaluar el impacto de un programa educativo basado en el uso de Scratch tanto en la enseñanza de las matemáticas como en las actitudes que presentan los alumnos a la hora de afrontar la enseñanza de las matemáticas.

De esta manera, una vez establecido el objetivo general, se pretende como objetivos específicos dar respuesta a las siguientes preguntas de investigación:

RQ1: ¿Cuál es el efecto de la enseñanza basada en juegos de Scratch en el rendimiento de los alumnos en Matemáticas?

RQ2: ¿Cuál es el efecto del uso de Scratch para la enseñanza de Matemáticas en las actitudes que presentan los alumnos hacia esta asignatura?

RQ3: ¿Hay suficientes recursos para aplicar Scratch en la enseñanza de las Matemáticas en los recursos literarios ya existentes?

3. Metodología

Una revisión sistemática se define como una metodología eficaz que permite un análisis de manera sistemática y objetiva de los estudios científicos más relevantes sobre un objeto de estudio (Sánchez-Meca y Botella, 2010). Es por esto por lo que nuestra revisión propone obtener tanto una panorámica descriptiva del campo objeto de estudio, como datos bibliográficos que nos ayuden a identificar aspectos clave de interés en torno a nuestro objeto de análisis.

Adicional a esta definición, Gómez-Luna proporciona unas directrices en su artículo Metodología para la revisión bibliográfica y la gestión de información de temas científicos, a través de su estructuración y sistematización. En él la revisión bibliográfica es concretada como la descripción detallada de un tema concreto, pero sin incluir tendencias que lleven a plantear diferentes escenarios a cerca de la tecnología en cuestión (Gómez-Luna et al., 2014).

Partiendo de ambas definiciones, constatamos la importancia que adquiere la revisión bibliográfica para adquirir un campo de visión mayor sobre el tema en cuestión y sobre los diferentes autores que lo abordan.

3.1. Unidades de Análisis y Fuentes

Este estudio consiste en una revisión sistemática de la literatura internacional actual relacionada con el uso de Scratch en la enseñanza de las Matemáticas, en dos de

las bases de datos más relevantes en el panorama educativo: WOS y SCOPUS.

En este trabajo se analizan un total de 112 artículos sobre investigación en la aplicación o uso de Scratch en la enseñanza de la base de datos científicos. Por un lado, 81 documentos de la Colección Principal de WOS y 31 de la base de datos de SCOPUS.

El análisis de los artículos lo realiza un investigador de forma individual, a través de todas y cada una de las publicaciones extraídas de WOS y SCOPUS desde el año 2012 al 2021. Este estudio se lleva a cabo a principios del año 2022, por lo que no se tuvieron en cuenta los artículos publicados durante este año.

3.2. Procedimientos

Una vez establecidas las bases de datos a utilizar para dicha investigación se procede a la búsqueda de los artículos, determinando su selección con indicadores tales como: título que contenga la palabra “Scratch”, categoría establecida en WOS, tipo de acceso de los documentos y producción cronológica.

Para responder de la manera más adaptada y acertada a los objetivos propuestos, en ambas bases de datos se sigue un criterio de selección basado en tres procedimientos a seguir: en primer lugar, se efectúa la búsqueda de información en las bases de datos anteriormente indicadas, ojeando para ello los títulos y resúmenes de cada uno de los artículos. En segundo lugar, se descartan aquellas obras cuyo título o

resumen no cumpla con los criterios establecidos. Y por último se procede a examinar el texto completo de aquellos documentos seleccionados.

Las principales características que debe contemplar una revisión son: sistemática, explícita y reproducible (Codina, 2015). Con el fin de conseguirlas, se seguirán las distintas secuencias propuestas en el protocolo PRISMA (Moher, Liberati, Tetzlaff, y Altman, 2009).

La tabla 1 expone la ecuación de búsqueda empleada para la selección de artículos incluidos en la revisión y sus resultados.

Tabla 1. Estrategia de búsqueda empleada en cada base de datos.

Base de Datos	Ecuación de búsqueda	Resultados
WOS	TI = "SCRATCH" AND (PY = "2021" OR PY = "2020" OR PY = "2019" OR PY = "2018" OR PY = "2017" OR PY = "2016" OR PY = "2015" OR PY = "2014" OR PY = "2013" OR PY = "2012") AND (SU = "EDUCATION EDUCATIONAL RESEARCH" OR SU = "MATHEMATICS") AND DT = "Article"	81
SCOPUS	TITLE ("SCRATCH") AND (LIMIT-TO (DOCTYPE, "ar")) AND (LIMIT-TO (SUBJAREA, "SOC") OR LIMIT-TO (SUBJAREA, "MATH")) AND (LIMIT-TO (PUBYEAR, 2021) OR LIMIT-TO (PUBYEAR, 2020) OR LIMIT-TO (PUBYEAR, 2019) OR LIMIT-TO (PUBYEAR, 2018) OR LIMIT-TO (PUBYEAR, 2017) OR LIMIT-TO (PUBYEAR, 2016)	59

OR LIMIT
TO (PUBYEAR , 2015)
OR LIMIT-
TO (PUBYEAR , 2014)
OR LIMIT-
TO (PUBYEAR , 2013)
OR LIMIT-
TO (PUBYEAR , 2012))
AND (LIMIT-
TO (EXACTKEYWORD ,
"Scratch") OR LIMIT-
TO (EXACTKEYWORD ,
"Education"))

Para la revisión sistemática, en primer lugar, se establecen las bases de datos que se van a utilizar y en este caso son Web of Science y Scopus.

A partir de aquí se formula la ecuación de búsqueda teniendo en cuenta indicadores como: Scratch contenido en el título, producción cronológica (últimos diez años), tipo de documento (artículos) y áreas de investigación.

Por un lado, el número de artículos que se obtuvieron tras aplicar dicho criterio de búsqueda fue de 81 en WOS y 59 en Scopus, con un total de 140 artículos.

Esta primera parte compone la etapa de identificación y una vez finalizada, se pasa a la etapa de cribado donde se identifican el número de artículos duplicados entre las distintas bases de datos que se han utilizado, así como una evaluación de los artículos tras leer el título y resumen para cada uno.

Una vez constatados el número de artículos duplicados entre WOS y Scopus, se reduce el número de artículos a 112 por lo que había 28 duplicados.

Con estos 112, se realiza un análisis preciso del título y el resumen para valorar qué artículo se adecúa a la

investigación que se quiere realizar y cuál se puede depurar. Tras esta verificación, se eliminan 91 artículos y se tienen 21 estudios.

Como tercer paso, se encuentra la idoneidad cuyo objetivo es depurar el número de artículos tras leer el texto completo de los obtenidos en la etapa de cribado, y de aquí, se tienen 21 artículos, para los cuáles se procede a leer el texto completo.

Así, se reduce el número de artículos a 12 para, finalmente, en la etapa de Inclusión evaluar metodológicamente los últimos estudios seleccionados y decidir cuáles se ajustan a los criterios e indicadores establecidos. Tras la aplicación de todas las fases, se incluyen 12 artículos en esta revisión sistemática.

A continuación (Figura 1), se expone el proceso de selección de los estudios resultantes, en distintas fases propuestas en la declaración PRISMA (Gerard Urrútia, 2009):

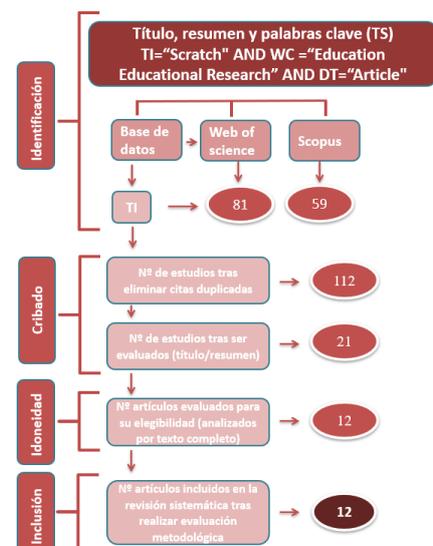


Figura 1. Diagrama de Flujo de PRISMA para el análisis de los artículos seleccionados.

4. Resultados

A continuación, se presentan los resultados obtenidos de los previos análisis efectuados, clasificados de la siguiente manera para dar respuesta a las cuestiones planteadas:

4.1. Documentos de acceso abierto y cerrado

Tras los resultados, se pone de manifiesto que, en relación al tipo de artículo, se obtiene que de los 112 artículos examinados el 34% disponía de acceso abierto y el otro 66% se correspondía con acceso cerrado (Figura 2).

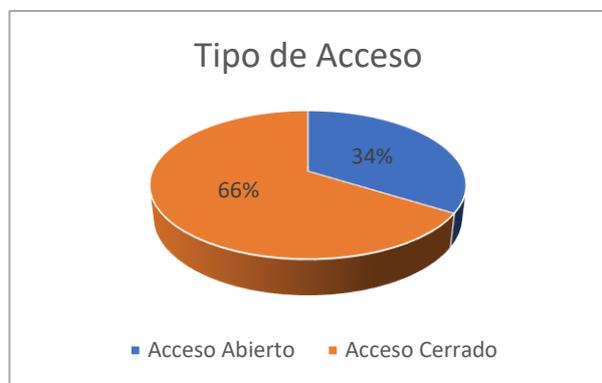


Figura 2. Porcentaje de Artículos por Tipo de Acceso.

Como punto de partida, se ha incluido la diferencia entre los documentos de acceso abierto y acceso cerrado de los 112 artículos donde nuestros resultados manifiestan una mayoría en el caso de los de tipo cerrado. Esto se puede explicar debido a la muestra que hemos seleccionado ya que, gran parte de nuestros artículos corresponde a la base de datos Web of

Science, la cual destaca por almacenar un gran volumen de documentos de tipo cerrado.

4.2. Producción cronológica

Esta revisión recoge distintos indicadores que permiten visualizar el estado actual sobre el uso de Scratch en la educación, así como su evolución en los últimos 10 años.

Tras revisar la gráfica obtenida respecto al número de artículos publicados en los últimos diez años (Figura 3), se pone de manifiesto que el año con un mayor número de publicaciones es 2020.

Destacar la diferencia de artículos publicados en los dos últimos años donde, en 2020, se presentaba un escalón muy importante en cuanto a publicaciones pero que descendió pronunciadamente en el año 2021.

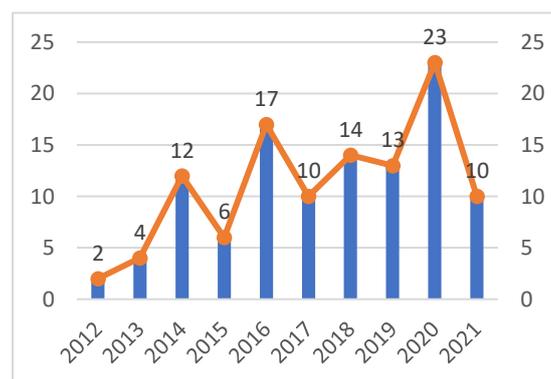


Figura 3. Número de Artículos publicados por Año

Respecto a la evolución con el paso de los años, observamos que ha habido un aumento de publicaciones de artículos relacionados con este tema, pero, ese crecimiento no ha sido progresivo de un año

para otro ya que, podemos observar que de 2014 a 2015 hubo una disminución de publicaciones, así como de 2016 a 2017. Sin embargo, basándonos en el objetivo de medir la progresión en los últimos 10 años, podemos interpretar la importancia que está abarcando el uso de Scratch en la enseñanza valorando la diferencia significativa entre el número de artículos publicados en 2012 respecto a 2020.

4.3. Tipo de Muestra

Para analizar la muestra de los artículos elegidos se hace una categorización de distintos escenarios:

- De 3 a 5 años: Educación Infantil
- De 6 a 12 años: Educación Primaria
- De 13 a 16 años: Educación Secundaria
- Universidad
- Formación Profesional
- No Académico
- Desconocido

En el diagrama de barras (Figura 4) podemos observar que, mayoritariamente, los artículos realizan el experimento utilizando alumnos que se encuentran en la etapa de educación primaria (38 de los 112 totales). Mencionar que, para el bloque “No académico” la mayoría de los artículos se aplicaban en prisiones por lo que podemos ver la gran expansión que puede tener el uso de Scratch en cuanto a diversidad. Otro aspecto a subrayar es el número de publicaciones aplicadas en profesores ya que, no solamente depende de los alumnos si no también hay que tener en cuenta la

formación que los profesores poseen para poder aplicar Scratch en el aula.

Tabla 2. Distribución de los artículos en función del tipo de muestra

Tipo de Muestra	N ^a de artículos	%
Educación Infantil	11	9,9%
Educación Primaria	38	34%
Educación Secundaria	23	20,5%
Universidad	16	14,2%
Formación Profesional	5	4,3%
Profesores	8	7,2%
No Académico	3	2,7%
Desconocido	8	7,2%

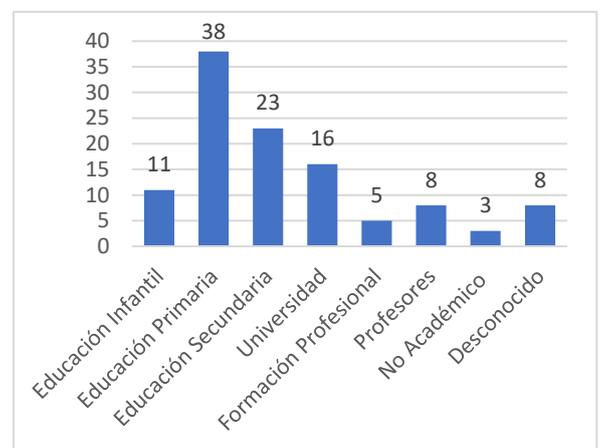


Figura 4. Número de Artículos publicados en función del tipo de muestra

En cuanto al tipo de muestra donde se aplica un mayor número de investigaciones relacionadas con nuestro tema, se focaliza en los estudiantes de Educación Primaria, seguidos por lo de Educación Secundaria. Esto se puede explicar aludiendo a que en la etapa de

primaria es donde los niños desarrollan una gran cantidad de habilidades, empiezan a enfrentarse a asignaturas básicas de la educación y, con Scratch, se les podría ayudar a esta iniciación y a una mejor comprensión de los principales conceptos. Para Educación Secundaria, los conocimientos comienzan a ser más exigentes y el uso de recursos relacionados con Tecnologías de Información y Comunicación favorecen la comprensión de los nuevos temas a los que enfrentan.

Especial hincapié, en los artículos donde la muestra está relacionada con el profesorado. ¿Es eficiente aplicar el uso de Scratch en la enseñanza de las Matemáticas si el profesor no está bien formado sobre este lenguaje de programación? Para dar respuesta a esta cuestión, nos vamos al artículo “Students' interest in Scratch coding in lower secondary mathematics” (Dohn, Niels Bonderup, 2020), donde los resultados muestran un efecto negativo en la actitud de los alumnos frente a la relación entre programación y matemáticas atribuyéndolo al nivel de dificultad experimentado y al tedioso o desorganizado flujo de trabajo. A través de esta idea, descubrimos que la integración de Scratch en la enseñanza debe venir acompañada de una formación del profesorado en este lenguaje de programación.

4.4. Materias de Aplicación de Scratch

Uno de los criterios aplicados en la fórmula de búsqueda de artículos es que cada uno de ellos contuviera la palabra “Scratch” en el título. Con esto,

el campo de aplicación resultante ha sido muy variado por lo que, para aumentar el uso de Scratch en las Matemáticas, es importante conocer donde se está aplicando actualmente con mayor relevancia y, así poder tener un punto de partida para aplicarlo en las Matemáticas. Tras representar los resultados correspondientes a esta temática, podemos visualizar (Figura 5), que el mayor uso de Scratch se realiza en la enseñanza de lenguajes de programación y pensamiento computacional con un 73% del total, seguido del uso de Scratch en las Matemáticas con un 12% y, finalmente, se aplica en muchas de las materias que se llevan a cabo en la educación con un porcentaje similar, como por ejemplo, Idiomas (6%), Física 4(%) y el grupo de otras que abarca Lengua, Historia y Biología (5%).

En este punto cabe destacar la diferencia que hay del uso de Scratch en la enseñanza de lenguajes de programación con la aplicación en el área de las Matemáticas y que nos lleva a entender una importante brecha actual en la aplicación de Scratch en las Matemáticas, así como con las demás materias de enseñanza.



Figura 5. Porcentaje de materias donde se usa Scratch como recurso de enseñanza

Se remarca una importante diversidad de materias en la que se está aplicando Scratch como complemento o recurso. Cabe poner de manifiesto la gran aplicación en la enseñanza de la programación donde Scratch está altamente capacitado para ser un recurso potencial con un gran impacto en la mejora de habilidades de codificación, entendimiento de los conceptos de pensamiento computacional, así como en la creación de algoritmos.

Por otro lado, puede remarcarse la relación entre Scratch y la asignatura de Matemáticas, donde el porcentaje destaca por encima del resto de materias y esto puede abrir una importante vía de mejora en la enseñanza de esta asignatura mediante este lenguaje de programación.

No menos importante es observar que, aunque con un porcentaje menor, existen artículos donde se experimenta con Scratch en otras materias que no son del área científica como puede ser Idiomas, Historia o Lengua y que muestra el gran potencial que puede ofrecer Scratch como recurso de enseñanza.

4.5. País de los artículos

En lo que respecta al país de publicación del artículo, la Tabla 5 muestra que el 21% de los documentos se encuentran presentados en España, seguido de un 15% en Estados Unidos y un 11% en China. Le siguen países como Turquía, Grecia, Inglaterra y Taiwan.

País	N.º de Autores	% de Autores
España	23	21%
Estados Unidos	17	15%
China	12	11%
Turquía	8	7%
Grecia	6	5%
Inglaterra	5	4%
Taiwan	3	3%
Colombia	2	2%
Israel	2	2%
Jaón	2	2%
Otros países	31	28%

Tabla 3. Distribución de los artículos en función del país

Subrayar que España se encuentra a la cabeza en la producción de documentos de la temática tratada, mostrando un compromiso sobre la evolución digital como complemento en la enseñanza, así como en el uso de nuevas prácticas innovadoras en el contexto escolar.

5. Estudio de los Resultados

En el bloque de metodología (Capítulo 4) se ha presentado el número total de artículos que componen esta revisión sistemática, siendo un total de 12. Por ello, se realiza un análisis en profundidad en cada uno de estos artículos y, a continuación, se presentan los resultados específicos para estos estudios.

El tipo de muestra donde se aplican estos estudios es uno de los indicadores más importantes que se contempla en este estudio. Por ello, la muestra se

concentra en los alumnos que se encuentran en la etapa de Educación Primaria como los estudiantes de Educación Secundaria, exceptuando un artículo donde la muestra son 18 futuros maestros (Shahbari, Daher, Baya y Jaber, 2020).

Destacar que este artículo se incluye por el interesante atractivo de su temática donde se examina el uso de funciones metacognitivas para resolver problemas de programación basados en matemáticas mediante el uso de Scratch, así como por sus importantes resultados que demuestran un hallazgo en el desarrollo de funciones metacognitivas como solucionadores de problemas relacionados con aspectos matemáticos y una mayor evolución en las funciones metacognitivas de regulación en aspectos matemáticos.

En la Figura 6, podemos ver como se distribuye el tipo de muestra.

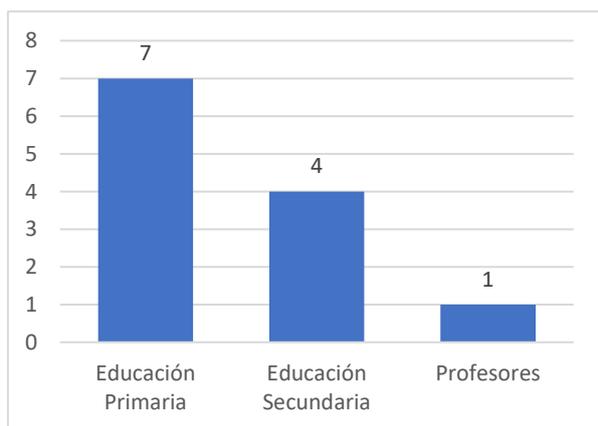


Figura 6. Número de Artículos publicados en función del tipo de muestra

En relación a los recursos presentados en los artículos se identifica una importante carencia de casos prácticos de Scratch que puedan ser utilizados y

aplicados en la enseñanza de las Matemáticas. Únicamente 3 de los 12 artículos (Martínez Zarzuelo, Rodríguez Mantilla, Roanes Lozano y Fernández Díaz, 2020), (Falloon, 2016) y (Rodríguez – Martínez, González-Calero, Sáez-López, 2019), presentan material y recursos educativos basados en Scratch que pueden ser aplicados en las aulas.

Los resultados indican que se descubrió que el uso del software Scratch en la educación tiene efectos positivos en la motivación, la autoeficacia, la actitud, el pensamiento de alto nivel y el éxito académico, además de que materializar la enseñanza del software al liberarlo de su estructura abstracta y compleja, permite que los estudiantes adquieran habilidades del siglo XXI a una edad temprana, se alienta a los estudiantes a confiar en sí mismos (Talan, 2020).

Así, los resultados indican que Scratch puede favorecer el desarrollo tanto de ideas matemáticas como pensamiento computacional (Martínez Zarzuelo, Rodríguez Mantilla, Roanes Lozano y Fernández Díaz, 2020) a la vez que posibilita la creatividad y el desarrollo del razonamiento lógico para dar respuestas a diversas situaciones (de Lima, Silva Souza Ferrete, Vasconcelos, 2021).

Además, indicaron mejoras significativas en el desempeño de los estudiantes en la resolución de ecuaciones y en sus actitudes hacia el aprendizaje de las matemáticas con la ayuda de la tecnología (Chiang y Qin, 2018) y revelaron logros de aprendizaje en los campos de números, geometría y resolución de

problemas matemáticos. (Temperman, Anthoons, De Lievre y De Stercke, 2014).

En estudios anteriores, se sugirió que diseñar juegos podía mejorar las actitudes de aprendizaje matemático de los estudiantes (Ke, 2014), mientras que en otros se indicaron que los juegos de computación podían influir significativamente en los logros de los estudiantes en matemáticas (Kebritchi et al., 2010).

Este estudio (Taylor et al., 2010) también demostró que los estudiantes, tras utilizar Scratch en Matemáticas, estaban más dispuestos a pedir ayuda a los profesores y compañeros de clase, o discutir problemas entre ellos, cooperando de mejor manera en la resolución de problemas.

Se confirma que las intervenciones habilitadas por Scratch tienen un efecto positivo general en el rendimiento académico STEM de los estudiantes (Fidai, Capraro y Caprano, 2020).

En definitiva, los resultados muestran resultados positivos utilizando una metodología basada en lenguaje en bloques mediante Scratch y una implementación útil del pensamiento computacional como recurso transversal en Matemáticas, mejorando no solo la competencia matemática sino también la lingüística y aportando una mejora en la lectura y comprensión del enunciado del problema. (Molina Ayuso, Adamuz Povedano y Bracho López, 2020).

6. Conclusiones

El principal objetivo de esta investigación ha sido analizar el impacto de las actividades de programación o juegos utilizando Scratch tanto en el aprendizaje de ideas matemáticas como en las actitudes presentadas por los alumnos hacia las matemáticas en estudiantes de Educación Primaria y Educación Secundaria, todo ello basándonos en el rendimiento a la hora de la resolución de problemas y las actitudes presentadas a la hora de aprender matemáticas con la ayuda de la tecnología.

En la sección objetivos (Capítulo 3) se planteó un objetivo general acompañado de unos interrogantes a los que dar respuesta con este estudio, centrándonos en las etapas educativas de Educación Primaria y Educación Secundaria.

En primer lugar, para la RQ1 cuyo interrogante es conocer el efecto de la enseñanza basada en juegos de Scratch en el rendimiento de los alumnos en Matemáticas, se confirma que los alumnos presentan mejoras significativas en el desempeño en la resolución de actividades relacionadas con las matemáticas y en sus actitudes hacia el aprendizaje de las matemáticas con la ayuda de Scratch.

El objetivo principal es analizar el potencial de las actividades de programación mediante Scratch en el aprendizaje de ideas matemáticas y, en base a los resultados presentados anteriormente se deja claro que el desarrollo de los conceptos computacionales favorece la resolución de problemas matemáticos y, en

específico, Scratch se encuentra en un momento de vital importancia en la integración del campo matemático.

Para la RQ2, ¿cuál es el efecto del uso de Scratch para la enseñanza de Matemáticas en las actitudes que presentan los alumnos hacia esta asignatura?

Hoy en día, la mayoría de los estudiantes se sienten presionados y aburridos a la hora de afrontar las clases de matemáticas y vemos que con la inclusión de juegos basados en Scratch se abre una oportunidad para promover nuevos sentimientos positivos como pueden ser la relajación y la curiosidad así como hacerles sentir más seguros de sí mismos después de haber hecho un juego y obtener la resolución de problemas matemáticos por una vía distinta a la tradicional que le hacen enriquecer varias de sus competencias educativas.

También, se proporciona la ocasión a compartir sus juegos con el resto de los compañeros ya que Scratch facilita el trabajo dinámico y cooperativo. Al compartir su trabajo, los estudiantes pueden identificar sus errores y aprender métodos de solución más eficaces o mejores.

Por último, para la RQ3, ¿hay suficientes recursos para aplicar Scratch en la enseñanza de las Matemáticas en los recursos literarios ya existentes?

Los resultados presentan una progresión positiva en el estudio de esta temática con la aparición y crecimiento de artículos relacionados con este campo, pero se

puede observar, en el bloque de análisis de resultados (capítulo 6), que muy pocos de ellos presentan recursos, por lo que existen importantes limitaciones que llevan a continuar con futuros estudios para, por ejemplo, poder tratar conceptos y habilidades de todas las áreas matemáticas que compone el currículo (álgebra, geometría, estadística, etc).

Para el objetivo general, se ha llevado a cabo un estudio de los diferentes artículos e investigaciones seleccionadas para tratar de obtener información relevante de este tema con cada uno de los indicadores a analizar y, adicionalmente, mediante la interpretación de resultados y conclusiones de las investigaciones donde la metodología era experimental. No solamente se han representado los beneficios que genera el uso de Scratch, si no también presentamos las limitaciones que se nos presentan en este estudio.

En resumen, Scratch se encuentra en ebullición en lo que respecta a su uso en distintos campos de la educación ya que los estudios que existen hasta el momento presentan resultados positivos en el aprendizaje de los alumnos una vez que se incluye una metodología basada en juegos en la enseñanza de la asignatura. Hay que comentar que no va a ser una tarea fácil y habrá muchas piedras en el camino para implantar este método de innovación y cambio pero, con la formación de los docentes y de los propios alumnos, se podrá afrontar un horizonte de aprendizaje nuevo que ofrecerá la posibilidad de “aprender a aprender”.

7. Limitaciones

En este estudio se detectan un conjunto de limitaciones. En cuanto al número de artículos se puede señalar la selección de únicamente dos bases de datos por lo que no se han tenido en cuenta todas las bases de datos existentes. También, el concepto de Scratch es muy amplio y nos hemos encontrado con el problema de obtener muchos artículos relacionados con el pensamiento computacional y su aplicación en el campo de la programación por lo que el número de artículos relacionado con Matemáticas no ha sido tan amplio como se hubiera deseado ya que se ha obtenido un total de 12 artículos relacionados con el campo de las Matemáticas.

Una brecha importante es la relación de artículos que se centran en medir el impacto de Scratch en los aspectos cognitivos de los estudiantes y los que aportan recursos prácticos de Scratch para aplicarlos en la enseñanza de las Matemáticas en las aulas, ya que el 90% de los artículos recogidos no ofrecen material didáctico, pero sí habla de las ventajas en la actitud de los estudiantes que ofrece la integración de Scratch en las aulas.

También, se identifica una confusión a la hora de definir y poner en práctica de forma concreta y exhaustiva qué enseñar, aprender y evaluar sobre pensamiento computacional en los niveles postobligatorios, incluso con programas populares similares a Scratch.

8. Agradecimientos

Este trabajo ha sido realizado gracias a Ayuda Puente 2021, URJC, Ref. M2614

9. Referencias

Aamir Fidai, Mary Margaret Capraro, Robert M. Capraro (2020). “Scratch”-ing computational thinking with Arduino: A meta-analysis, *Thinking Skills and Creativity*, Volume 38, 100726, ISSN 1871-1871, <https://doi.org/10.1016/j.tsc.2020.100726>.

Agalianos, A., Noss, R., Whitty, G. (2001) Logo in mainstream schools: *The struggle over the soul of an educational innovation*. *British Journal of Sociology of Education* 22(4): 479–500.

Álvaro Molina Ayuso, Natividad Adamuz Povedano y Rafael Bracho López (2020). La resolución de problemas basada en el método de Polya usando el pensamiento computacional y Scratch con estudiantes de Educación Secundaria. *Aula Abierta*, Volumen 49, número 1, enero-marzo, 2020/págs. 83-90

Assessing Scientific, Reading and Mathematical Literacy. A Framework for PISA 2006. OECD Publishing, 2006.

Benton, L., Hoyles, C., Kalas, I., & Noss, R. (2017). Bridging primary programming and mathematics: Some findings of design research in England. *Digital*

Experiences in Mathematics Education, 3(2), 115–138. doi:10.1007/s40751-017-0028-x.

Brennan, K., Balch, C., & Chung, M. (2014). Creative computing. Retrieved from <http://scratched.gse.harvard.edu/guide/>

Calao, L. A., Moreno-León, J., Correa, H. E., & Robles, G. (2015). Developing mathematical thinking with *Scratch*. In G. Conole, T. Klobučar, C. Rensing, J. Konert, & E. Lavoué (Eds.), *Design for Teaching and Learning in a Networked World. Lecture Notes in Computer Science*, vol 9307 (pp. 17–27). Cham: Springerdoi:10.1007/978-3-319-24258-3_2.

Calder, N. (2010). Using Scratch: An integrated problem-solving approach to mathematical thinking. *Australian Primary Mathematics Classroom*, 15(4), 9–14. doi:10.1007/s10857-012-9226-z.

Cetin I. Preservice Teachers' Introduction to Computing: Exploring Utilization of Scratch. *Journal of Educational Computing Research*. 2016;54(7):997-1021. DOI:10.1177/0735633116642774.

Chen, G., Shen, J., Barth-Cohen, L., Jiang, S., Huang, X., & Eltoukhy, M. (2017). Assessing elementary students' computational thinking in everyday reasoning and robotics programming. *Computers and Education*, 109, 162–175. doi:10.1016/j.compedu.2017.03.00

Chiang, F., & Qin, L. (2018). A Pilot study to assess the impacts of game-based construction learning,

using scratch, on students' multi-step equation-solving performance. *Interactive Learning Environments*, 26(6), 803–814. doi:10.1080/10494820.2017.1412990.

Codina, L. (2015). *No lo llame Análisis Bibliográfico, llámelo Revisión Sistematizada. Y cómo llevarla a cabo con garantías*. Systematized Reviews: SALSA Framework.

de Moraes, A. D., Basso, M. V. d. A., & Fagundes, L. d. C. (2017). *Educação Matemática & Ciência da Computação na escola: aprender a programar fomenta a aprendizagem de matemática? Ciência & Educação* (Bauru), 23(2), 455–473. doi:10.1590/1516-731320170020011.

DiSessa, A. A. (2000). *Changing minds: Computers, learning, and literacy*. Cambridge: MIT Press.

Dubinsky, E. (Ed.). (1995). ISETL: A programming language for learning mathematics. *Communications on Pure and Applied Mathematics*, 48(9), 1027–1051.

Erol, Osman & Çırak, Neşe. (2022). The effect of a programming tool scratch on the problem-solving skills of middle school students. *Education and Information Technologies*. 27. 10.1007/s10639-021-10776-w.

Falloon, G.. (2016). An analysis of young students' thinking when completing basic coding tasks using Scratch Jnr. On the iPad: General thinking and

computational work. *Journal of Computer Assisted Learning*, 32. 10.1111/jcal.12155.

Feng-kuang Chiang & Lian Qin (2018): A Pilot study to assess the impacts of game-based construction learning, using scratch, on students' multi-step equation-solving performance, *Interactive Learning Environments*, DOI: 10.1080/10494820.2017.1412990

Feurzeig, W., Papert, S. A. (2011) *Programming-languages as a conceptual framework for teaching mathematics*. *Interactive Learning Environments* 19(5): 487–501.

Funke, A., Geldreich, K., & Hubwieser, P. (2017). Analysis of Scratch projects of an introductory programming course for primary school students. In 2017 IEEE global engineering education conference (EDUCON) (pp. 1229–1236). IEEE. doi:10.1109/EDUCON.2017.7943005

Gäetan Temperman, Caroline Anthoos, Bruno De Lièvre, Joachim De Stercke. Tâches de programmation avec Scratch `a l`écolé primaire : Observation et analyse du développement des compétences en mathématique. Frantice.net, Res@TICE/Université de Limoges, 2014, TICE : scolarisation et ´évolution des pratiques, <http://www.frantice.net/document.php?id=1013>.

Garneli, V., & Chorionopoulos, K. (2018). Programming video games and simulations in science education: Exploring computational thinking through

code analysis. *Interactive Learning Environments*, 26(3), 386–401. doi:10.1080/10494820.2017.1337036

Gómez-Luna, E., Fernando-Navas, D., Aponte-Mayor, G., & Betancourt-Buitrago, L. A. (2014). Metodología para la revisión bibliográfica y la gestión de información de temas científicos, a través de su estructuración y sistematización. *Dyna*, 81(184), 158–163.

Grover, S., Pea, R., & Cooper, S. (2015). Designing for deeper learning in a blended computer science course for middle school students. *Computer Science Education*, 25(2), 199–237.

Hallinger, P., y Bridges. E. M. (2017). A Systematic Review of Research on the Use of Problem-Based Learning in the Preparation and Development of School Leaders. *Educational Administration Quarterly*, 53(2), 255-288. DOI: 10.1177/0013161X16659347.

Han, B., Bae, Y., & Park, J. (2016). The effect of mathematics achievement variables on Scratch programming activities of elementary school students. *International Journal of Software Engineering and Its Applications*, 10(12), 21–30.

Hughes, J., Gadanidis, G., & Yiu, C. (2017). Digital making in elementary mathematics education. *Digital Experiences in Mathematics Education*, 3(2), 139–153. doi:10.1007/s40751-016-0020-x.

- Jonassen, D. H. (1995) Computers as cognitive tools: *Learning with technology not from technology*. Journal of Computing in Higher Education 6(2): 40–73.
- José Antonio Rodríguez-Martínez, José Antonio González-Calero & José Manuel Sáez-López (2019): Computational thinking and mathematics using Scratch: an experiment with sixth-grade students, Interactive Learning Environments, DOI: 10.1080/10494820.2019.1612448
- Kebritchi, M., Hirumi, A., & Bai, H. (2010). The effects of modern mathematics computer games on mathematics achievement and class motivation. Computers & Education, 55, 427–443.
- Kenneth, R. (1996). Calculators in the mathematics curriculum: The scope of personal computational technology. In A. J. Bishop, K. Clements, C. Keitel, J. Kilpatrick, & C. Laborde (Eds.), *International handbook of mathematics education*. Part 1 (pp. 435–468). Dordrecht: Kluwer Academic Publishers.
- LIMA, I. P.; FERRETE, A. A. S. S.; VASCONCELOS, A. D. Potencialidad del Scratch en la educación básica. Revista Ibero-Americana de Estudos em Educação, Araraquara, v. 16, n. 2, p. 598-609, abr./jun. 2021. e-ISSN: 1982-5587. DOI: <https://doi.org/10.21723/riaee.v16i2.13225>
- Lye, S. Y., & Koh, J. H. L. (2014). Review on teaching and learning of computational thinking through programming: What is next for K-12? Computers in Human Behavior, 41, 51–61. doi:10.1016/j.chb.2014.09.012
- Maloney, J., Resnick, M., Rusk, N., Silverman, B., Eastmond, E. (2010) The Scratch programming language and environment. ACM Transactions on Computing Education 10(4): 1–15.
- Mannila, L., Dagiene, V., Demo, B., Grgurina, N., Mirolo, C., Rolandsson, L., & Settle, A. (2014). Computational thinking in K-9 education. In Proceedings of the working group reports of the 2014 on innovation & technology in computer science education conference - ITiCSE-WGR '14 (pp. 1–29). New York
- Martínez-Zarzuelo, Angélica & Rodríguez-Mantilla, Jesús & Roanes-Lozano, Eugenio & Díaz, María. (2020). Efecto de Scratch en el aprendizaje de conceptos geométricos de futuros docentes de primaria. Revista Latinoamericana de Investigación en Matemática Educativa. 23. 357-386. 10.12802/relime.20.2334.
- Mladenovic, M., Krpan, D., & Mladenovic, S. (2016, July 4–6). *Introducing programming to elementary students novices by using game development in Python and Scratch*. EDULEARN proceedings of the 8th international conference on Education and New Learning Technologies (EDULEARN), Barcelona, Spain (pp. 1622–1629).
- Moher, D., Liberati, A. Tetzlaff, J., y Altman, D. (2009). Preferred Reporting Items for Systematic Reviews and Meta-Analyses: The PRISMA

Statement. *PLoS Medicine*, 6(7), e1000097. DOI: 10.1371/journal.pmed.1000097.

Papert, S. A. (1980). *Mindstorms: Children, computers, and powerful ideas*. New York: Basic Books.

Papert, S.A. (1990). *A critique of technocentrism in thinking about the school of the future*. Cambridge, MA: Epistemology and Learning Group, MIT Media Laboratory.

Papert, S. (1993) *Mindstorms: Children, computers, and powerful ideas (2nd ed.)*, New York, NY: Basic Books.

Resnick, M. (2012) Reviving Papert's dream. *Educational Technology* 52(4): 42–46.

Resnick, M., Maloney, J., Monroy-Hernández, A., Rusk, N., Eastmond, E., Brennan, K., Kafai, Y. (2009) Scratch: *Programming for all*. *Communications of the ACM* 52(11): 60–67.

Rodríguez-Martínez, J. A., González-Calero, J. A., & Sáez-López, J. M. (2019). *Computational thinking and mathematics using Scratch: an experiment with sixth-grade students*. *Interactive Learning Environments*, 1–12. doi:10.1080/10494820.2019.1612448.

Sánchez-Meca, J., y Botella, J. (2010). Revisión sistemática y meta-análisis: Herramientas para la práctica profesional. *Papeles del Psicólogo*, 31(1), 7-17.

Shahbari, Juhaina & Daher, Wajeih & Baya'a, Nimer & Jaber, Otman. (2020). Prospective Teachers' Development of Meta-Cognitive Functions in Solving Mathematical-Based Programming Problems with Scratch. *Symmetry*. 12. 1569. 10.3390/sym12091569.

Shin, N., Sutherland, L. M., Norris, C. A., & Soloway, E. (2012). Effects of game technology on elementary student learning in mathematics. *British Journal of Educational Technology*, 43(4), 540–560.

Talan, Tarik (2020). Investigation of the Studies on the Use of Scratch Software in Education. **Journal of Education and Future; Ankara** N°18: 95-111. DOI:10.30786/jef.556701

Wing, J. M. (2008). Computational thinking and thinking about computing. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 366(1881), 3717–3725. doi:10.1098/rsta.2008.0118

Wing, J. M. (2011, February). Research notebook: *Computational thinking- what and why? The Link Magazine*, 20–23.

Wing, J. M. (2014). Computational thinking benefits society [Blog post]. Retrieved from <http://socialissues.cs.toronto.edu/index.html%3Fp=279.htm>