

Formación del profesorado con scratch: análisis de la escasa incidencia en el aula

Jon Bustillo Bayón

Universidad del País Vasco (España)

jon.bustillo@ehu.eus

Resumen

La popularización de dispositivos digitales educativos, ha favorecido el desarrollo de muchas acciones formativas orientadas hacia la incorporación en el ámbito escolar de nuevas prácticas educativas, que favorezcan el desarrollo del pensamiento computacional. Después de haber impartido más de 30 talleres a diferentes niveles de profesorado y alumnado de Magisterio con la herramienta de programación Scratch, se ha observado que estas acciones están teniendo poca incidencia en las aulas de los centros escolares. En el siguiente trabajo se analizan cuáles son los motivos para proponer un nuevo modelo que tras una primera experiencia ha arrojado resultados prometedores.

Palabras clave: Formación de profesorado, Scratch, Mentoring, Pensamiento computacional.

Teacher Training With Scratch: Analysis of Low Impact in the Classroom

Abstract

The popularization of educational digital devices has favored the development of many formative actions faced towards the incorporation of new educational practices in schools. The main aim of these practices

is the development of computational thinking. Having delivered more than 30 workshops on different levels of teachers and students of Education with Scratch programming tool, it is found that these actions are having little incidence in classrooms of schools. In this paper we analyze the reasons for this lack of transfer of knowledge and we propose a new model that after an initial experience has shown promising results.

Key words: Teacher training, Scratch, Mentoring, Computational thinking.

1. INTRODUCCIÓN

El desarrollo tecnológico está suponiendo la creación de nuevos recursos educativos que permiten de forma sencilla, la adquisición de competencias clave a través de la generación de diferentes tipos de construcciones, artefactos o programas. Estos entornos se caracterizan por ser atractivos, tener una barrera de entrada pequeña y atender a muy diferentes tipos de intereses y conocimientos (Resnick y Silverman, 2005 y Traylor, 2008).

El aprendizaje de conceptos abstractos no suele ser sencillo por la falta de referencias concretas que permitan ser experimentados. Las características que ofrecen los dispositivos digitales facilita que se hagan tangibles muchos de estos conceptos. Su capacidad de representar los fenómenos de forma que sean perceptibles a través de los sentidos de las personas, la posibilidad de adaptarlos a las necesidades de las personas aprendices (edad, discapacidades,...) y el carácter social derivado de que sea observable por muchas personas simultáneamente, son los tres pilares básicos que permiten a los dispositivos digitales hacer tangibles muchos de los conceptos abstractos (Zuckerman, Arida y Resnick, 2005). Los procesos de aprendizaje asociados a la experimentación si bien ofrecen muchas ventajas, también tienen su límite en el “horizonte del aprendizaje”. Éste vendría delimitado por aquellas situaciones de aprendizaje cuyas consecuencias no sean perceptibles, bien porque la respuesta a un evento se dé fuera del alcance espacio-temporal del observante, o bien porque los efectos tienen lugar en otros elementos diferentes al afectado. En todos estos casos, el uso de conceptos abstractos y la comprensión de la lógica de los sistemas, son esenciales para comprender fenómenos no experimentables (Senge, 2010). No obstante, la comprensión de la lógica de sistemas que incluyen múltiples elementos y sus relaciones, es mu-

cho más sencilla que se produzca a través de procesos activos de aprendizaje, en donde el alumnado tiene la oportunidad de experimentar el comportamiento del sistema interaccionando con él.

Autores como Forrester (1994) consideran que en el marco del s. XXI se pueden distinguir dos tipos de aprendizajes para conceptos abstractos o sistemas. Por un lado, estaría lo que denomina “system thinking” centrado en aprender del sistema sus características, las experiencias de otras personas y las relaciones de las partes de una forma teórica. Por otro lado, estaría lo que se denomina como “system dynamics”, en donde el alumnado aprende haciendo y experimentado a través de lo que se conoce como “ensayo-error”. La principal diferencia entre estos dos tipos de aprendizaje reside en que, mientras el primero apenas incide en los modelos de pensamiento, el segundo tiene una incidencia directa en la creación de nuevos modelos mentales. En este sentido, el desarrollo de dispositivos que acerquen a través de experiencias tangibles a las relaciones de un sistema, serán de gran ayuda en el desarrollo de capacidades cognitivas asociadas al uso de abstracciones (Resnick et al., 1998; Zuckerman y Resnick, 2004).

La labor docente (en un rol de facilitador) o bien el acceso a comunidades en donde participan otras personas creadoras, son elementos clave que recalcan el carácter social del proceso de aprendizaje (Kurland y Pea, 1985; Maher, 2012 y Resnick, 1996). Compartir las creaciones se convierte en una magnífica oportunidad para facilitar el acceso a nuevas ideas, sugerencias y conocimientos que faciliten la mejora continua de la creación inicial. Las producciones compartidas enfatizan el valor que sobre los procesos de aprendizaje tiene compartir el conocimiento (Brennan, Monroy-Hernández y Resnick, 2010 y Monroy-Hernández y Resnick, 2008).

Tanto el proceso cognitivo de generación de conocimientos, como la generación de artefactos que los representen, requieren de un entorno que favorezca su desarrollo. El proceso de aprendizaje exigiría que tanto en el entorno cognitivo como en el creativo-productivo hubiera elementos que facilitasen su desarrollo y respetasen la singularidad de los aprendices. Para ello, el tiempo de trabajo se erige como una variable crítica, haciendo que no sea un elemento asociado al tiempo medio estipulado para ese tipo de tarea y sí, al tiempo necesario para que el alumnado logre interiorizar dichos conocimientos o habilidades. Los procesos de aprendizaje están sujetos a las características y motivaciones de las personas y eso exige proponer entornos de aprendizaje en donde haya lugar para cada una de ellos. La práctica habitual en los entornos formativos se distancia de esta premisa, adoptando más

un modelo industrial en el que se diseñan planes de enseñanza gobernados por la presión del tiempo, en donde toma más relevancia la ordenación temporal de la acción docente, que los diferentes tiempos que necesita cada estudiante para integrar de forma significativa lo que aprende (Kafai y Resnick, 1996 y Papert, 1996).

La tecnología debe ser usada como algo que el niño controle, manipule y que le permita extenderse para controlar el entorno, adquiriendo un sentido de poder sobre la tecnología que le haga sentirse como agente intelectual. Dicho de otra manera, coincido con Dewey, Montessori, y Piaget en que los niños aprenden haciendo y pensando sobre lo que hacen, es por ello, que la innovación educativa debe venir a través de ofrecer mejores entornos para producir cosas y para pensar sobre cómo se están haciendo (Papert, 2005: 353).

1.1. Lenguaje de programación Scratch

El grupo de investigación del MIT Media Lab “Lifelong Kindergarten” junto con el grupo de la University of California, Los Ángeles (UCLA) liderado por Yasmin Kafai implementaron “Scratch”, un lenguaje de programación pensado para ser usado por personas ajenas al mundo de la informática y que permitiera realizar diferentes tipos de creaciones atendiendo a las características, conocimientos, gustos e intereses del mayor número de personas posible. Scratch es un lenguaje de programación gráfico y gratuito que se ejecuta en diferentes sistemas operativos (Windows, Mac y Linux) y que permite crear, programar y compartir producciones multimedia en un entorno digital y online. A través de este recurso se facilita que los estudiantes pasen de ser personas que consumen diferentes tipos de “media” a ser personas que producen sus propios “media”, adquiriendo una participación activa en el desarrollo de producciones que se ajustan a los intereses, formatos y nivel de competencia.

Las primeras versiones fueron testadas con éxito en diferentes Computer ClubHouse (Gallagher, 2010) y se comprobó que las personas que participaban en estos centros preferían crear programas con Scratch antes que usar otro tipo de software disponible, por ejemplo editores de fotografías, mezcladores de audio, etc. Un seguimiento durante 18 meses de las personas que trabajaron con Scratch, así como el análisis de más de 530 proyectos, confirmó algunas de las claves del éxito previstas en el diseño del programa (Maloney, Peppler, Kafai, Resnick y Rusk, 2008). A modo de resumen se pueden destacar las siguientes:

- Las personas participantes reconocían rápidamente el valor y el potencial de la herramienta para abordar proyectos que resultaban apasionantes para ellos.
- Permitió crear productos finales que se podían mostrar y compartir con otros compañeros.
- Permitió crear muy diferentes programas que atendían intereses de personas de muy diferentes aficiones, edades, conocimientos y culturas.
- La curva de aprendizaje fue muy pequeña, permitiendo crear sencillas aplicaciones muy rápidamente.
- Favoreció el incremento de la complejidad en función de sus necesidades y progresos.

Scratch permitía a estos jóvenes diseñar, crear y compartir con otros, proyectos diseñados y realizados en función de sus intereses. Siguiendo la argumentación de Resnick (2007) este modelo de aprendizaje es muy apropiado para la adquisición de competencias.

Scratch incorpora la programación a actividades asociadas con la manipulación de diferentes tipos de medios que actualmente son muy populares entre la gente joven. Éste entorno invita a las personas jóvenes a aprender a través de la exploración y el compartir entre iguales, lejos de los planes de instrucción utilizados habitualmente para otros tipos de lenguajes de programación (Resnick, 2007:2).

En la actualidad, la mayoría de los ordenadores de las aulas de educación primaria y secundaria de la Comunidad Autónoma Vasca incorporan versiones de Scratch tanto para Windows como para Linux, siendo un recurso que está a disposición de la mayoría de los docentes en las aulas de los centros escolares.

Desde la administración educativa se han ofrecido múltiples talleres formativos para que los docentes incorporen este recurso a las escuelas, pero los esfuerzos realizados hasta la fecha han tenido poca incidencia en las aulas. Esta circunstancia, priva a muchos alumnos de la adquisición de competencias claves para los ciudadanos del s. XXI, en el entorno escolar reglado (Johnson et al., 2014). En las siguientes líneas se analizan las razones por las que los docentes no logran incorporar Scratch a su práctica docente. Por último se relatará una propuesta que ya se ha experimentado en 5 centros y cuyos resultados preliminares son esperanzadores.

2. METODOLOGÍA

Para comprender las causas que no están permitiendo que todas estas acciones formativas tengan una incidencia en la escuela, en el siguiente trabajo se recogerá información proveniente de diferentes informantes a través del uso de distintos tipos de instrumentos. Se desarrollan entrevistas en profundidad con docentes de diferentes centros educativos que han participado en talleres de Scratch, se realizan grupos de debate con alumnos que dentro de su formación reglada reciben formación con Scratch, se medirá a través de un cuestionario el grado en que el alumnado valora la incorporación de Scratch en las aulas y se incluirán tasas de éxito académico de los futuros docentes que deben adquirir competencia en el uso de Scratch.

Para la interpretación de toda esta información se realiza un análisis semántico a través de un sistema categorial en donde se definen las dimensiones, categorías y subcategorías correspondientes, delimitando las circunstancias que actúan como favorecedoras y obstaculizadoras en la incorporación de Scratch a la escuela. La validez del estudio vendría dada tanto por la participación de diferentes informantes, como por el uso de diferentes instrumentos de acceso a la información.

Seguidamente y a la vista de los resultados obtenidos, se realiza una propuesta para la actualización del profesorado que ayude a la incorporación en las aulas de nuevos recursos, medios y metodologías. Una propuesta que se ha puesto en marcha durante el presente curso en cinco centros de educación primaria de la provincia de Álava y cuyos resultados preliminares son esperanzadores.

2.1. Plan formativo inicial

El objetivo inicial del plan formativo consistía en que el profesorado experimentara que era capaz de crear videojuegos de su interés y del nivel de complejidad que cada uno quisiera. Para ello, se comenzó por ofrecer herramientas suficientes para que se familiarizarasen con el entorno de trabajo y comprendieran la dinámica de programación a través del uso de los comandos. Se propusieron varios ejemplos que servían de base para la creación de videojuegos, concretamente, a través del desarrollo de un circuito simple, se mostró la gestión de los movimientos, la interacción con el teclado, las condiciones y el uso de variables. Posteriormente se mostró cómo realizar cuentos, en donde los personajes realizaban acciones sincronizadas a través del

envío y recepción de mensajes entre sprites y fondos. Seguidamente se mostró cómo crear proyectos con instrumentos musicales que ejecuten sonidos a través de la interacción en el teclado y se mostró cómo realizar un reloj analógico en donde las manecillas giraban cada segundo y/o minuto. Por último, se mostró cómo hacer preguntas y generar distintas actuaciones en función de las respuestas obtenidas.

Este plan formativo se desarrolló durante 10 sesiones de 75 minutos, a razón de dos por semana. Después, durante 5 sesiones, cada participante debía abordar en clase el desarrollo de algún proyecto propio (preferentemente sobre alguna temática de su interés) que fuese multimedia y permitiese algún tipo de interacción por parte de quien estuviera jugando con su creación. La evaluación estaba consensuada y observaba el proceso de creación, otorgando más valor al esfuerzo de mejora que a la finalización del proyecto. Se trataba de observar la capacidad de cada uno de afrontar el desarrollo de producciones propias que estuvieran reguladas por la idea que cada uno había imaginado, todo ello dentro de la espiral del pensamiento proceso creativo descrita por Resnick (2007).

3. RESULTADOS

Los resultados se han agrupado en función de los informantes. Por una parte estarían los asociados a los datos recogidos con el alumnado de Magisterio y por otra parte los relativos a docentes en activo y asesores educativos del centro de innovación pedagógica de Vitoria-Gasteiz.

3.1. Incidencia sobre el alumnado de Magisterio

Desde el curso 2011-12 se incorporó el manejo de la herramienta Scratch al plan formativo de los nuevos docentes, concretamente en la asignatura “Nuevas tecnologías aplicadas a la educación” impartida en la Escuela de Magisterio de Vitoria-Gasteiz. Durante los primeros años las calificaciones que obtenían los estudiantes estaban lejos de lo deseable superando el 40% de no aprobados. Posteriormente se optó por realizar un cambio adoptando una propuesta idéntica a la realizada en los cursos de formación para profesorado en activo y que pone mayor énfasis en el desarrollo de propuestas didácticas aplicables a diferentes áreas curriculares. En concreto se abordó la creación de videojuegos para trabajar lengua (diferentes idiomas), geometría y música. Los resultados obtenidos son muy alentadores, ya que se observa un descenso significativo del número de no aprobados (ver tabla 1).

Tabla 1. Distribución de las calificaciones obtenidas por el alumnado de Magisterio de Vitoria-Gasteiz (NR: no presentado; A: aprobado; B: notable; C: Sobresaliente).

Año académico	N	Calificación formación Scratch			
		NR	A	B	C
2011-2012	93	42,00%	29,00%	19,00%	9,00%
2012-2013	76	30,00%	39,00%	27,00%	4,00%
2013-2014	109	36,00%	23,00%	37,00%	4,00%
2014-2015	132	24,00%	51,00%	23,00%	2,00%

No obstante, una vez concluido el periodo de formación con la herramienta Scratch, la mayoría del alumnado muestra una vaga idea de lo que es Scratch y de cómo se puede integrar en las dinámicas de aula de los centros escolares. A la hora de definir qué es Scratch, de forma mayoritaria se hace referencia a que es “*algo para programar o para entender algo de programación*” [M1, N1, N2, N3-G1] centrando la visión en el desarrollo de programas y no en el potencial que tiene para el desarrollo de las competencias básicas de los estudiantes.

Las referencias educativas previas a la formación en Scratch, habitualmente suelen estar condicionadas por las experiencias propias. En el caso de las personas que actualmente están estudiando Magisterio, éstas son básicamente tres. La vivencia como escolar en donde no tuvieron ocasión de conocer recursos educativos que les permitiesen aprender a programar, el tipo de formación que de forma mayoritaria reciben en la Escuela de Magisterio y por último las prácticas formativas que realizan en centros escolares. En ninguno de estos casos es habitual que los futuros docentes encuentren referencias a Scratch ni a la incidencia que puede tener en el desarrollo de las competencias básicas. Tal y como indican varios alumnos “*Yo no había escuchado Scratch en mi vida y si no fuera porque trabajamos con él en clase ... no se escucha en ningún sitio, ni a la gente, ni a profesores, ni en la televisión...*” [M2, N1-G1]. En la misma línea y en relación a su formación como futuros docentes, relatan que trabajar con Scratch supone una experiencia única, ya que las demás asignaturas siguen formatos formativos en donde los docentes permanentemente van dirigiendo el proceso de aprendizaje. Tal y como relata una alumna “*Creo que no estamos acostumbradas, trabajar con Scratch es algo totalmente diferente. Como decía antes, todos los*

días tenemos un montón de horas de clase en las que la dinámica es la misma. Esta clase es la única que es diferente” [N1-G1], la metodología utilizada para la formación con Scratch resulta totalmente diferente a la del resto de las disciplinas que tienen y esto se traduce en una falta de hábito y en la necesidad de hacer un acto de fe que otorgue credibilidad a las virtudes de Scratch como recurso didáctico. En el mismo sentido, cuando los estudiantes van de prácticas a diferentes centros escolares, tampoco encuentran ejemplos que les ayude a comprender cómo puede ser de utilidad el uso de recursos como Scratch para favorecer la adquisición de competencias por parte del alumnado. Los alumnos indican que sería de gran ayuda poder ver y conocer de cerca buenas prácticas de aula utilizando Scratch.

En el momento que veamos que el alumnado trabajando en el aula con Scratch desarrolle la lógica, entonces supongo que empezaremos a creer lo que nos indicas. Si me dices que está comprobado que cuando el alumnado trabaja con Scratch se desarrollan no sé qué capacidades, entonces empezaré a creer en la validez de Scratch como recurso didáctico [N2-G1].

A la hora de aprender a usar Scratch, todos los alumnos creen que pueden ser capaces de adquirir competencia suficiente, pero indican que es necesario hacer un importante esfuerzo que sin la motivación o necesidad necesaria es difícil de realizar. Igualmente consideran que el proceso de aprendizaje propuesto no es el adecuado, ya que no se les da un plan de trabajo claro e inequívoco para su ejecución. Prefieren algo que sea muy dirigido y no abierto, esto es, una propuesta formativa que esté más centrada en la ejecución y no en el proceso.

El problema es que no estamos acostumbrados a esta dinámica, desde pequeños siempre hemos tenido un docente que permanentemente nos ha estado marcando qué teníamos que hacer. No hemos trabajado la autonomía y claro, cuando tenemos que ponernos delante del ordenador para imaginar, crear, programar... un proyecto, sin que nadie te diga qué ni cómo hacer, se hace muy difícil [N3-G1].

No disponer de una guía paso a paso, implica tener que estar en un bucle con “imaginar-definir-programar-probar-compartir” en donde lo más relevante es el proceso y que según indican les resulta muy cansado. “*Se hace muy cansado tener que diseñar desde el comienzo, qué queremos, cómo hacerlo, cómo incorporar mejoras,...*” [N1-N2, G1].

Esto difiere de un modelo mucho más dirigido, centrado en el producto y caracterizado por leer-comprender-organizar-programar-comprobar, en donde se encuentran mucho más habituados. En este sentido, los errores son vividos por los alumnos no como una oportunidad de aprender, sino como algo que ralentiza la finalización de un producto. *“Nuestra idea es realizar el proyecto con Scratch en el mínimo tiempo posible, ... en ese caso, los errores los vivimos como una putada, eso implica dedicar mucho tiempo para avanzar poco”* [N1-G1].

Algo muy destacable es que los estudiantes de Magisterio no han sido capaces de identificar qué competencias han trabajado cuanto ellos han estado trabajando con Scratch. Esto no permite una fácil comprensión, ni transferencia de la potencialidad de Scratch como recurso didáctico favorecedor del desarrollo de competencias.

Es algo que yo no veo, cuando he estado trabajando con Scratch he estado frente al ordenador viendo si lo que programaba funcionaba pero ni se me ocurrió pensar qué estaba trabajando cognitivamente. Realmente no sabemos qué estamos trabajando cuando usamos Scratch [N2, N3, M1-G1].

A pesar de no ver Scratch como un recurso didáctico de gran valor y que ninguno cree que va a utilizarlo cuando ejerza como docente, todos los alumnos coincidían en el atractivo que para los niños de hoy en día tienen los dispositivos digitales. Igualmente destacaban la necesidad de que hubiera en la escuela un cambio metodológico que aunque reconocían no saber cómo acometer, sí entendían que exigiría un uso de las TICs diferente al que han conocido en la escuela. *“Lo que sí es cierto es que aunque se están incorporando las nuevas tecnologías a las aulas, estas se están utilizando como antes. Por ejemplo, la mayoría de las veces vemos utilizar las pizarras digitales como si fueran las de tiza”* [N3-G1].

Se analizó el grado de entusiasmo que suscitó el uso de Scratch entre alumnado que había trabajado con él durante los cursos anteriores (N=69). Para ello se utilizó en modelo propuesto por O'Brien y Torms (2010) que cuenta con seis dimensiones: La usabilidad percibida, el valor estético, grado de tolerancia, el grado de novedad, grado de concentración exigido y el grado de participación. En los siguientes gráficos se muestran los resultados obtenidos (ver Gráfico 1).

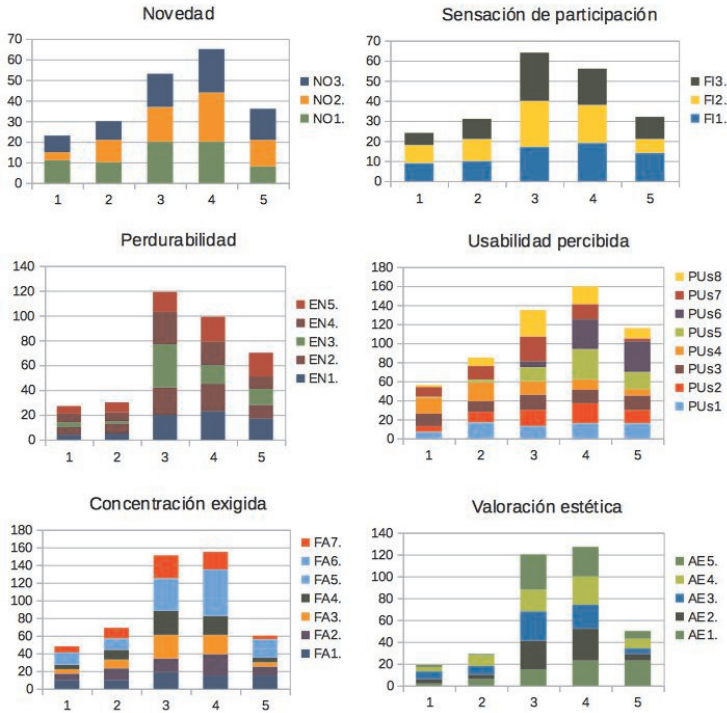


Gráfico 1. Novedad (NO): NO1. He seguido programando con Scratch por curiosidad. NO2. Programar con Scratch ha despertado mi curiosidad. NO3. Estoy interesado en programar con Scratch. Sensación de participación (FI): FI1. Me ha gustado mucho programar con Scratch. FI2. Me siento motivado programando con Scratch. FI3. Scratch es divertido. Perdurabilidad (EN): EN1. Mereció la pena programar con Scratch. EN2. Considero un éxito el tiempo dedicado a Scratch. EN3. Programar con Scratch no resultó como esperaba. EN4. Programar con Scratch ha sido gratificante. EN5. Recomendaría probar Scratch a otra gente. Usabilidad percibida (PUs): PUs1. Me siento frustrado/a cuando uso Scratch. PUs2. Me resulta confuso usar Scratch. PUs3. Me pongo nervioso mientras programo con Scratch. PUs4. Me siento desmotivado a programar con Scratch. PUs5. Programar en Scratch es realmente cansado mentalmente. PUs6. Programar requiere mucha atención. PUs7. Cuando programo con Scratch, siento que tengo el control. PUs8. No puedo hacer algunas de las cosas que quiero hacer cuando programo con Scratch. Concentración exigida (FA): FA1. Cuando uso Scratch, estoy tan metido en la programación que pierdo la noción del tiempo. FA2. Estoy tan atento a mi programación que no me doy cuenta del tiempo que paso con Scratch. FA3. No presto atención al resto de cosas cuando estoy programando con Scratch. FA4. Cuando programo con Scratch, no me fijo en el resto de cosas que me rodean. FA5. El tiempo vuela cuando programo con Scratch. FA6. Cuando programo en Scratch estoy absorto/a. FA7. Me dejo llevar mientras programo con Scratch. Valoración estética (AE): AE1. Scratch es bonito. AE2. Scratch es agradable estéticamente. AE3. Me gustan los gráficos e imágenes usados en Scratch. AE4. Programar en Scratch es agradable visualmente. AE5. La organización de los elementos en la pantalla de Scratch es agradable.

3.2. Incidencia sobre los docentes en activo

Se realizó un grupo de discusión con asesores educativos pertenecientes al centro de Innovación Educativa dependiente del Gobierno Vasco¹. En él participaron 18 asesores (13 mujeres / 5 hombres) de edades comprendidas entre los 40 y 63 años, que de forma voluntaria decidieron participar en este estudio. Cada asesor ayuda en el desarrollo de innovaciones y proyectos educativos en varios centros, teniendo así una magnífica perspectiva para determinar el grado en que se incorporan nuevas prácticas en las aulas de los centros tutelados.

El 78% considera que la presencia de Scratch en el aula es muy pequeña a pesar de los numerosos cursos realizados a tal efecto. El 22% restante indica no tener ninguna referencia de la presencia de Scratch en las aulas de Educación Primaria. Sobre estos datos, se planteó una discusión en la que se invitó a los asesores a indicar las razones que a su juicio pudieran explicar la escasa incidencia que estaban teniendo los planes formativos con Scratch en el aula. Las respuestas quedaron agrupadas en tres bloques, a) falta de formación, b) metodología en el aula y c) apoyo institucional.

En primer lugar se indicaba la necesidad de ofrecer mayor formación para que los docentes pudieran dominar la herramienta con garantías suficientes para abordar las demandas que se derivasen por parte de los alumnos. En segundo lugar, se apuntaba la necesidad de disponer de una guía didáctica en la que se hicieran propuestas de intervención educativa que justificasen el desarrollo por parte del alumnado de las competencias previstas en el *currículo* de cada curso. Por último, se indica que es necesario un entorno favorecedor por parte de la administración educativa, que aliente a poner en práctica estas y otras nuevas herramientas y metodologías educativas.

Atendiendo a la capacidad de los docentes para incorporar este tipo de recursos didácticos, solo un 21% de los asesores cree que el profesorado sería capaz de llevarlo efectivamente al aula. Otro 21% declaraba que dependería de las circunstancias particulares de cada docente, es decir, de su competencia tecnológica, del apoyo recibido por parte de la administración y de la situación laboral. Por contra, el 58% de los asesores consideraba muy poco probable que los docentes actuales fueran a incorporar herramientas como Scratch a su práctica docente.

4. DISCUSIÓN DE LOS RESULTADOS

El objetivo último de incorporar la programación a las escuelas no reside en lograr más personas que se dediquen a las ciencias de la computación. Más bien se pretende generalizar a través de la educación obligatoria el desarrollo del pensamiento computacional para las generaciones venideras. No obstante, esta no es una cuestión sencilla que tiene implicaciones en las actividades cognitivas asociadas al diseño, comprensión, modificación y depuración de los programas (Rogalski y Samurçay, 1990). A pesar de ser reconocidos los beneficios relativos al desarrollo del pensamiento creativo, la resolución de problemas, el razonamiento lógico y la experimentación sistematizada, todas estas competencias digitales tienden a ser olvidadas por los planes de estudios desarrollados en las escuelas (Kafai, Peppler y Chapman, 2009).

Teniendo como horizonte la adquisición de competencias para el s. XXI, tanto los futuros docentes como los que ya están en ejercicio, reconocen la necesidad de incorporar nuevas prácticas al aula asociadas a las nuevas tecnologías. No obstante, a través de los datos recogidos se pueden identificar algunas situaciones que no facilitan un tránsito natural desde los planes de formación a docentes (grado en educación o formación continua) a las aulas.

Los futuros docentes destacan que no haber aprendido nada de programación cuando cursaron estudios en educación primaria o secundaria, supone un hándicap que en ocasiones no es sencillo de superar. En este sentido, la mayoría de los alumnos del grado de Magisterio inicialmente no consideran que vayan a ser capaces de realizar videojuegos con Scratch. Afortunadamente esta autopercepción va cambiando en la medida en que se van involucrando en el desarrollo de creaciones propias que se ajusten a sus intereses y su formación. Casos similares se han podido observar en los trabajos de (Kafai et al., 2009 y Bustillo, Vizcarra y Aristizabal, 2015). Igualmente, que el alumnado perciba la mayoría de las asignaturas de Magisterio como acciones formativas tradicionales, en donde apenas hay espacio para desarrollar creaciones de su interés, tampoco ayuda a incorporar recursos como Scratch en la práctica de aula. Esta falta de referentes, también se observa cuando el alumnado hace sus prácticas en centros de educación primaria. Tal y como reportan los alumnos, excepto uno o dos compañeros (menos del 1%) ningún otro ha visto utilizar Scratch como recurso de aula.

La falta de experiencias en las aulas contrasta con los planes de formación continua del profesorado que anualmente ha ido ofertando cursos de formación sobre Scratch. Esto indica que no existe un trasvase del conocimiento adquirido en dichas acciones formativas hacia el aula. Sobre esta circunstancia tanto docentes que han participado en diferentes talleres de Scratch, como asesores de innovación educativa coinciden en destacar tres pilares que tienen una incidencia negativa: manejo de la herramienta Scratch, disposición de guías didácticas y soporte por parte de la administración educativa (dirección del centro, departamentos, AMPAs e inspección).

La necesidad de tener un buen manejo de la herramienta, denota un temor del profesorado a no saber suficiente, incluso de perder el control de lo que suceda en el aula. En este punto sería deseable una redefinición del rol del docente, pasando de ser la persona que siempre sabe responder a lo que pregunte el alumnado a una persona que ayuda a buscar las respuestas al alumnado. En la misma línea se encontraría la demanda de “guías didácticas”, que actuarían como libros de textos o fichas que dibujan un carril por el que todo el alumnado debería transitar, no dejando espacio ni tiempo para que pudieran diseñar, crear, equivocarse, rediseñar, ... en un bucle similar a la espiral del pensamiento creativo propuesto por (Resnick, 2007a).

Tal y como Jacobsen, Clifford y Friesen (2002) indican, la incorporación de la programación en las aulas de educación básica no es sencilla y requiere por parte del profesorado grandes dosis de imaginación, inteligencia, creatividad y mucho coraje, especialmente en aquellos entornos educativos dominados por un modelo curricular. De esta forma el mayor reto no se encuentra en tener mucha destreza en el manejo de una herramienta como Scratch, si no en mostrar experiencias de aula que integren la observación, las preguntas, la prueba y el error como elementos necesarios para la adquisición de la competencia digital (Ahern, 2009).

Tanto los estudiantes de Magisterio como los docentes en activo coinciden a la hora de indicar que Scratch es una herramienta atractiva para el alumnado. Esta valoración también coincide con la recogida en los estudios realizados por Fesakis y Serafeim (2009) y Wolz, Leitner, Malan y Maloney (2009). No obstante y coincidiendo con el trabajo realizado por Meerbaum-Salant, Armoni y Ben-Ari (2010) ambos colectivos destacan la necesidad de conocer buenas prácticas de aula en las que a través del desarrollo de videojuegos con Scratch el alumnado adquiere competencias asociadas al pensamiento computacional.

4.1. Alumnado de Magisterio como mentores de Scratch

El análisis de los factores obstaculizadores permitió dibujar un esquema con tres dimensiones que se debían superar. Por una parte está la necesidad de disponer de recursos para dominar la herramienta Scratch, por otra estaría la experimentación en el entorno habitual de trabajo del docente y por último, tener respaldo por parte de las diferentes administraciones educativas.

Bajo esta perspectiva se diseñó una experiencia piloto en la Ikastola Etxaurren (Álava). En este caso, en colaboración con la profesora de inglés, se realizaron 10 sesiones de Scratch con alumnado de 5º y 6º de primaria, orientadas a la creación de videojuegos relacionados con la asignatura. En palabras de la docente,

Así sí que se aprende. No tienen nada que ver ir a un curso de formación o que venga otro docente al aula a ayudar en la incorporación de herramientas como Scratch.... además, estar en tu entorno, con tus recursos y con tu alumnado, te ayuda a ver qué funciona, qué no y así poder tomar decisiones en función de cómo se integran estas nuevas herramientas en la dinámica de clase. Todos los docentes deberían poder tener la oportunidad de ver cómo funcionan estas nuevas herramientas en su aula y con su alumnado [D1-EP1].

Igualmente, el alumnado aceptó la presencia de un nuevo docente que les ayudaba a aprender a realizar videojuegos.

El recorrido de esta experiencia no solo se redujo al tiempo en el que hubo la ayuda de un docente experto en Scratch, sino que el desarrollo de historias interactivas en lengua inglesa fue una actividad que se integró en la dinámica de clase durante todo el curso, ofreciendo así nuevas posibilidades de aprendizaje al alumnado.

Esta experiencia mostró que la incorporación por un tiempo limitado de una persona con conocimientos suficientes de Scratch en el aula, podía disipar en gran medida los factores obstaculizadores descritos por los docentes en activo, facilitando así la incorporación de nuevos recursos y metodologías en las aulas de educación primaria.

Por otra parte, el alumnado de Magisterio de Vitoria-Gasteiz, que recibe formación en el uso educativo de Scratch, realiza diferentes estancias de prácticas en muchos de los centros de educación primaria de Álava. Esto abría la posibilidad de extender la presencia de estudiantes de Magisterio

con formación en Scratch en centros educativos que solicitasen ayuda para incorporar esta herramienta educativa en las prácticas docentes.

Durante el presente curso se realizó una experiencia piloto a través de un acuerdo entre el centro de innovación pedagógica de Vitoria-Gasteiz, la Escuela de Magisterio de Vitoria-Gasteiz y cinco centros de educación primaria. Los resultados preliminares son muy prometedores y prueba de ello es que de cara al próximo curso, además de los centros que ya han participado en este curso, se han incorporado otros dos más para el que viene. Otro indicador ha sido el importante incremento de proyectos provenientes de los centros participantes en esta experiencia piloto en el ScratchEguna celebrado en el museo Artium de Vitoria-Gasteiz. Tal y como relataba una de las profesoras participantes,

Más de la mitad de los proyectos se han creado gracias a la ayuda de los alumnos en prácticas. Sin ellos muchos de los trabajos aquí presentados no hubieran sido posibles. Además, han permitido que muchos alumnos hayan aprendido a desarrollar proyectos con Scratch. Han sido de gran ayuda [D2-EP1].

5. CONCLUSIÓN

La escuela, con el objetivo último de ayudar a que los educandos adquieran las competencias definidas por la administración educativa como clave para desenvolverse en el siglo XXI, se encuentra ante la necesidad de incorporar recursos y metodologías que aborden el desarrollo del pensamiento computacional. La programación ha sido un área de conocimiento ajena a la gran mayoría de los docentes actuales y ello conlleva la necesidad de emprender acciones formativas que aseguren al alumnado de primaria la adquisición de los primeros fundamentos del pensamiento computacional en entornos educativos reglados (Berry, 2015).

Los esfuerzos realizados por la administración educativa para formar en el uso educativo de la herramienta Scratch a los docentes en activo, no ha traído consigo un transvase de ese conocimiento a las aulas de educación primaria. No obstante, y coincidiendo con estudios previos (Ertmer y Ottenbreit-Leftwich, 2010), se ha observado que cuando los estudiantes de Magisterio o los propios docentes, tienen la oportunidad de ver en las aulas de primaria intervenciones didácticas que incorporan Scratch, su perspectiva sobre el valor de esta herramienta educativa cambia de forma significativa. Esto se traduce en un mayor interés y entu-

siasmo a la hora de incorporarlo a la práctica docente, atendiendo tanto a los intereses como a la competencia en el uso de Scratch de alumnado y profesorado (Ertmer, 2005).

El alumnado de Magisterio podría actuar a través de las prácticas formativas, como el mentor o prescriptor que lleve de forma temporal la incorporación de Scratch a las aulas, permitiendo a los docentes en activo, aprender, experimentar y observar en su entorno la incidencia que estas actuaciones tienen sobre su alumnado. La breve experiencia realizada con cinco centros educativos avala este formato, que exige un marco de colaboración más amplio entre la universidad y la escuela y que podría extenderse a otras innovaciones educativas que también pudieran ser de interés.

Notas

1. Accesible en: <http://ingurugela-a.berritzeguneak.net/es/index.php>

Referencias Bibliográficas

- AHERN, Terence. 2009. Bridging the gap: cognitive scaffolding to improve computer programming for middle school teachers. **Frontiers in Education Conference**. FIE'09. 39th IEEE (pp.1-5). San Antonio (USA).
- BERRY, Miles. 2015. **QuickStart Primary Handbook**. Swindon (UK). BCS.
- BRENNAN, Karen; MONROY-HERNÁNDEZ, Andrés y RESNICK, Mitchell. 2010. Making projects, making friends: Online community as catalyst for interactive media creation. **New Directions for Youth Development**. Vol.128:75-83. <http://doi.org/10.1002/yd.377>
- BUSTILLO, Jon; VIZCARRA, M^a Teresa y ARISTIZABAL, Pilar. 2014. Análisis del proceso formativo de un grupo de reclusos en un taller de Scratch. **RELATEC**. Vol.13. N^o1: 37-49.
- ERTMER, Peggy. 2005. Teacher pedagogical beliefs: The final frontier in our quest for technology integration? **Educational technology research and development**. Vol. 53 N^o4: 25-39.
- ERTMER, Peggy y Ottenbreit-Leftwich, Anne. 2010. Teacher Technology Change: How Knowledge, Confidence, Beliefs, and Culture Intersect. **Journal of Research on Technology in Education**. Vol. 42. N^o3: 255-284. <http://doi.org/10.1080/15391523.2010.10782551>
- FESAKIS, Giorgios y SERAFEIM, Kiriaki. 2009. Influence of the familiarization with «scratch» on future teachers' opinions and attitudes about pro-

- programming and ICT in education. **ACM SIGCSE Bulletin**. Vol. 41. Nº3:258-262. <http://doi.org/10.1145/1595496.1562957>
- FORRESTER, Jay. 1994. Learning through System Dynamics as Preparation for the 21st Century. **Systems Thinking and Dynamic Modeling Conference for K-12 Education**, California, EEUU. Recuperado a partir de <http://mitsloan.mit.edu/alumni/convocation2008/pdf/D-4434-4.Concord%2021st.pdf>
- GALLAGHER, Larry; MICHALCHIK, Vera y EMERY, Deborah Kim. 2010. **Assessing Youth Impact of the Computer Clubhouse Network**. SRI International. Menlo Park (USA).
- JACOBSEN, Michele; CLIFFORD, Pat y FRIESEN, Sharon. 2002. Preparing Teachers for Technology Integration: Creating a Culture of Inquiry in the Context of Use. **Contemporary Issues in Technology and Teacher Education**. Vol. 2. Nº3: 363-388.
- JOHNSON, Larry; ADAMS BECKER, Samantha; ESTRADA, Victoria y FREEMAN, Alex. 2014. **The NMC horizon report Europe 2014** New Media Consortium. Luxemburgo. Recuperado a partir de <http://dx.publications.europa.eu/10.2791/83258>
- KAFAI, Yasmin; PEPLER, Kylie y CHAPMAN, Robbin. 2009. **The Computer Clubhouse?: constructionism and creativity in youth communities**. Teachers College Press. New York (USA).
- KAFAI, Yasmin y RESNICK, Mitchel. 1996. **Constructionism in practice: Designing, thinking, and learning in a digital world**. Erlbaum Associates. New Jersey (USA).
- KURLAND, Midian y PEA, Roy. 1985. Children's Mental Models of Recursive Logo Programs. **Journal of Educational Computing Research**. Vol.1. Nº2: 235-243. <http://doi.org/10.2190/JV9Y-5PD0-MX22-9J4Y>
- MAHER, Mary Lou. 2012. Computational and collective creativity: Who's being creative? **International Conference on Computational Creativity**, (pp. 67-71) Dublin (Ireland).
- MALONEY, John; PEPLER, Kylie; KAFAI, Yasmin; RESNICK, Mitchel y RUSK, Nataly. 2008. Programming by choice: urban youth learning programming with scratch. **ACM Press**. Vol. 40. Nº1: 367-371. <http://doi.org/10.1145/1352135.1352260>
- MEERBAUM-SALANT, Orni; ARMONI, Michal y BEN-ARI, Mordechai. 2010. Learning computer science concepts with scratch. **ACM Press**. Vol. 23. Nº3: 239-264 <http://doi.org/10.1145/1839594.1839607>
- MONROY-HERNÁNDEZ, Andrés y RESNICK, Mitchel. 2008. Empowering kids to create and share programmable media. **Interactions**. Vol. 15. Nº2: 50-53 <http://doi.org/10.1145/1340961.1340974>

- PAPERT, Seymour. 1996. A word of Learning. En KAFAI, Yasmin y Resnick, Mitchel (Eds.), **Constructionism in practice: designing, thinking, and learning in a digital world** (pp. 9-24). Lawrence Erlbaum Associates. Mahwah (USA)
- PAPERT, Seymour. 2005. Teaching Children Thinking. **Contemporary Issues in Technology and Teacher Education**. Vol. 5. N°3: 353-365.
- RESNICK, Mitchel. 1996. Distributed constructionism . **Proceedings of the 1996 International conference on Learning sciences**. (pp. 280-284). Illinois (USA).
- RESNICK, Mitchel. 2007a. All I really need to know about creative thinking I learned in kindergarten. **Proceedings of the 6th ACM SIGCHI conference on Creativity & cognition** (pp. 1-6). Washington. (USA). <http://doi.org/10.1145/1254960.1254961>
- RESNICK, Mitchel. 2007b. Sowing the seeds for a more creative society. **Learning and Leading with Technology**. Vol. 35. N°4: 18-22.
- RESNICK, Mitchel y SILVERMAN, Brian. 2005. Some reflections on designing construction kits for kids . **Interaction Design and Children**, (pp. 117-122). New York (USA). <http://doi.org/10.1145/1109540.1109556>
- ROGALSKI, Janine y SAMURÇAY, Renan. 1990. Acquisition of programming knowledge and skills. **Psychology of programming**. Vol 18. N°1: 157-174.
- SENGE, Peter M. 2010. **The Fifth Discipline: The Art and Practice of the Learning Organization**. Doubleday. New York (USA).
- TRAYLOR, Scott. 2008. Scratch that: MIT's Mitchel Resnick Says Kids Should Do It for Themselves. **Technology & Learning**. Vol. 29. N°1: 27-29.
- WOLZ, Ursula; LEITNER, Henry; MALAN, David y MALONEY, John. 2009. Starting with scratch in CS 1. **ACM SIGCSE Bulletin**. Vol. 41. N°1: 2-3 <http://doi.org/10.1145/1508865.1508869>.
- ZUCKERMAN, Oren; ARIDA, Saeed y RESNICK, Mitchel. 2005. Extending tangible interfaces for education. **Proceedings of the SIGCHI conference on Human factors in computing systems**. (pp. 859-868). New York (USA). <http://doi.org/10.1145/1054972.1055093>.
- ZUCKERMAN, Oren y RESNICK, Mitchel. 2004. Hands-on modeling and simulation of systems. **Proceedings of the 2004 conference on Interaction design and children: building a community**. (pp. 157-158). New York (USA). <http://doi.org/10.1145/1017833.1017868>.