

[Cierre de edición el 01 de Setiembre del 2019]

doi: <http://dx.doi.org/10.15359/ree.23-3.9>

URL: <http://www.una.ac.cr/educare>

CORREO: educare@una.cr

Programación de tecnologías para la inclusión social con Scratch: Prácticas sobre el pensamiento computacional en la formación docente

*Programming Technologies for Social Inclusion With Scratch: Computational Practices in a
Teacher's Professional Development Course*

*Programação de tecnologias para a inclusão social com Scratch: práticas de pensamento
computacional na formação de professores*



Natalia Monjelat

Instituto Rosario de Investigaciones en Ciencias de la Educación
Consejo Nacional de Investigaciones Científicas y Técnicas - Universidad Nacional de Rosario
Rosario, Argentina

monjelat@irice-conicet.gov.ar

 <https://orcid.org/0000-0002-5043-8989>

Recibido • Received • Recebido: 12 / 01 / 2018

Corregido • Revised • Revisado: 27 / 12 / 2018

Aceptado • Accepted • Aprovado: 02 / 05 / 2019

Resumen: En los últimos años se observa un interés por desarrollar prácticas, conceptos y perspectivas del pensamiento computacional y la programación en las escuelas, por lo cual resulta crucial la formación docente en estas temáticas. Sin embargo, poco se conoce sobre el proceso que siguen las maestras y los maestros cuando se acercan a estos contenidos novedosos. Considerando estas cuestiones, el presente estudio de caso exploratorio e intrínseco tiene por objetivo analizar las prácticas del pensamiento computacional desarrolladas por un grupo de 13 docentes en un taller de formación, donde crearon tecnologías para la inclusión social (TIS) programando con Scratch. El análisis de los diferentes datos recogidos durante esta experiencia piloto (encuestas, producciones realizadas por sus participantes, grabaciones en audio, etc.) evidencia una estrecha articulación entre las prácticas computacionales puestas en obra y la programación de proyectos hipermediales que abordan problemáticas propias de las personas participantes. Estos resultados muestran que el abordaje de la enseñanza de la programación se planteó como un proceso cercano a la realidad cotidiana de maestras y maestros, poniendo de relieve la importancia de generar prácticas educativas mediatizadas que sean a su vez situadas. Diseñar un trayecto formativo para crear TIS, aparece en este sentido como una opción válida para la enseñanza de la programación y el desarrollo del pensamiento computacional, en línea con estudios previos que proponen formar tanto a docentes en servicio como en formación, desde propuestas contextualizadas y ejemplos significativos en un marco comunitario y colaborativo de reflexión e intercambio.

Palabras claves: Pensamiento computacional; formación docente; Scratch; tecnologías para la inclusión social.



doi: <http://dx.doi.org/10.15359/ree.23-3.9>

URL: <http://www.una.ac.cr/educare>

CORREO: educare@una.cr

Abstract: In recent years, there is an interest in developing practices, concepts and perspectives of computational thinking and programming in schools. Therefore, it is crucial to train teachers in these subjects. However, little is known about the process that teachers follow when approaching these new contents. In this context, the present exploratory and intrinsic case study aims to analyze the computational thinking practices developed by a group of 13 teachers while participating in a professional development workshop, where they created technologies for social inclusion (TIS) by programming with Scratch. The analysis of the different data collected during this pilot experience (surveys, productions carried out by the participants, audio recordings, etc.) evidences a close articulation between computational practices and programming of hypermedia projects that address participants' contextualized problematics. These results show that professional development related to programming was proposed as a process close to the day-to-day reality of teachers, highlighting the importance of generating mediated and situated educational practices. Designing a training path to create TIS appears, in this sense, as a valid option for the teaching of programming and the development of computational thinking, in line with previous studies that proposed professional development courses for teachers from contextualized proposals, and using significant examples in a community and collaborative framework of reflection and exchange.

Keywords: Computational thinking; Teacher professional development; Scratch; Technologies for Social Inclusion.

Resumo: Nos últimos anos, há um interesse em desenvolver práticas, conceitos e perspectivas de pensamento computacional e programação nas escolas, onde a formação de professores nesses assuntos se torna crucial. No entanto, pouco se sabe sobre o processo que o corpo docente seguem quando abordam esses novos conteúdos. Considerando estas questões, o presente estudo de caso exploratório e intrínseco tem como objetivo analisar as práticas de pensamento computacional desenvolvidas por um grupo de 13 professores em uma oficina de formação, onde foram criadas tecnologias para inclusão social (TIS) através da programação com o Scratch. A análise dos vários dados coletados durante este projecto-piloto (entrevista, produções dos participantes, gravações de áudio, etc.) mostra uma ligação íntima entre as práticas computacionais utilizadas e a programação de projetos de hipermedia, que abordam problemas próprios das pessoas participantes. Os resultados mostram que a abordagem do ensino de programação foi considerada como um processo próximo ao cotidiano do corpo docente, enfatizando a importância da geração de práticas educativas mediadas e ao mesmo tempo situadas. Desenhar um caminho de formação para criar o TIS, surge como uma opção válida para o ensino de programação e desenvolvimento do pensamento computacional, em consonância com estudos previos que propõem formar professores tanto em serviço como educação, a partir de propostas contextualizadas e exemplos significativos em uma comunidade e estrutura colaborativa de reflexão e intercâmbio.

Palavras-chave: Pensamento computacional; formação de professores; Scratch; tecnologias para inclusão social.

doi: <http://dx.doi.org/10.15359/ree.23-3.9>URL: <http://www.una.ac.cr/educare>CORREO: educare@una.cr

Introducción

En los últimos años es posible observar un renovado interés por la introducción de la programación en contextos educativos, a través de múltiples iniciativas que destacan a su vez la importancia de esta temática para el desarrollo del denominado “pensamiento computacional” (Brennan y Resnick, 2012; Wing, 2011). En este sentido, la articulación con la formación docente es un punto crucial, aunque en una gran mayoría de países esta temática se encuentra aún en sus fases iniciales y no hay un consenso claro sobre qué implica su enseñanza o cómo debe ser introducida (Grover y Pea, 2013).

Las investigaciones internacionales observan un retraso entre los cambios en los planes de estudio y la preparación adecuada del profesorado, especialmente en cuanto a la integración de tecnologías de la información y la comunicación (TIC) (Balanskat y Engelhardt, 2015). Asimismo, una gran mayoría de docentes no cuentan con conocimientos previos sobre estas temáticas, suelen tener un bajo nivel de uso de tecnologías digitales en la enseñanza y no están ni técnica ni emocionalmente preparados para la introducción planificada de la educación en programación (Ohashi, 2017). Por otra parte, se observa que en el campo de la formación docente, los estudios suelen centrarse en las actitudes, percepciones o creencias del profesorado, poniendo el foco en el conocimiento conceptual y no así en las prácticas o perspectivas del pensamiento computacional, dimensiones que plantean Brennan y Resnick (2012) y que permiten el estudio del proceso, de las instancias intermedias en la construcción de un objeto tecnológico mediante la programación. Por otro lado, pese a la extensa bibliografía y materiales existentes en línea sobre enseñanza de la programación, es fundamental adecuar las propuestas de formación docente a las particularidades del contexto donde van a implementarse (Monjelat y San Martín, 2016). Además, en el contexto latinoamericano, los estudios sobre esta temática son escasos y la realidad de cada país es diferente en cuanto a políticas educativas y procesos de implementación de estas temáticas (Brackmann, Barone, Casali, Boucinha y Muñoz-Hernández, 2016).

Desde un enfoque sociocultural de la educación (Lave y Wenger, 1991; Vygotsky, 1978), las cuestiones presentadas reflejan la necesidad de realizar estudios específicos que den cuenta del proceso por el que atraviesa el personal docente cuando se enfrenta a la programación, para buscar desarrollar el pensamiento computacional, más allá del desarrollo de conceptos puntuales o el estudio sobre cambios en las creencias o actitudes. A su vez, se pone en evidencia la importancia de repensar las prácticas atendiendo a las características de nuestra región (Monjelat y San Martín, 2016).

Al plantear que las sociedades son tecnológicamente construidas al mismo tiempo que las tecnologías son socialmente configuradas, el enfoque sociotécnico podría resultar un aporte en la construcción de esta nueva mirada, al permitir considerar los procesos y productos que se generan a partir de actividades de programación como tecnologías para la inclusión social (TIS). Las TIS se definen como distintas maneras de “diseñar, desarrollar, implementar y gestionar tecnologías orientadas a resolver problemas sociales y ambientales, generando dinámicas sociales y económicas de inclusión social y desarrollo sustentable” (Thomas, 2012, p. 27).



doi: <http://dx.doi.org/10.15359/ree.23-3.9>

URL: <http://www.una.ac.cr/educare>

CORREO: educare@una.cr

Considerando estas cuestiones, este artículo reporta una experiencia de formación en programación situada durante la cual un grupo de docentes creó TIS a partir de problemáticas significativas para sus contextos. Partiendo de un estudio previo en torno al diseño del taller que funcionó como dispositivo de formación (Monjelat, 2017), el presente artículo tiene por objetivo explorar y analizar las prácticas del pensamiento computacional desarrolladas por el grupo de docentes participantes en dicha experiencia. En esa línea, las preguntas de investigación fueron las siguientes:

- ¿Cómo se presentan las prácticas del pensamiento computacional en un contexto de formación docente situado?
- ¿Cómo se articulan estas prácticas con la propuesta de creación de tecnologías para la inclusión social?

De esta manera, la investigación que aquí se reporta pretende profundizar en el estudio de la formación en el pensamiento computacional, con el objetivo de identificar cómo se acerca el profesorado a estas temáticas novedosas, pues pone el foco en el cómo y no solo en el resultado final. De esta forma, el presente estudio busca aportar evidencias que permitan diseñar trayectos de formación docente significativos, atendiendo a las particularidades del contexto y poniendo el eje en la experiencia de las docentes y los docentes a través del estudio de sus prácticas computacionales, aspectos no siempre considerados en las propuestas formativas en estas temáticas innovadoras.

El pensamiento computacional: Su desarrollo a través de la programación

Diversos autores y autoras consideran clave en la pedagogía de las ciencias de la computación el desarrollo de las habilidades de pensamiento computacional. Este concepto originalmente planteado por Papert, de la mano del construccionismo, ha sido popularizado por Wing (2011). Sin embargo, actualmente no hay consenso sobre lo que una definición para el pensamiento computacional podría abarcar y cuáles son sus elementos concretos (Grover y Pea, 2013). Pese a ello, diferentes estudios coinciden en que es una habilidad de pensamiento analítico que, aunque se basa en conceptos de la informática, es una práctica central para todas las ciencias. En virtud de ello, la esencia del pensamiento computacional implica dividir problemas complejos en subproblemas más familiares o manejables (descomposición del problema), usando una secuencia de pasos (algoritmos) para resolverlos, repasando cómo la solución se transfiere a problemas similares (abstracción) y finalmente determinando si una computadora puede ayudarnos a resolver esos problemas de manera más eficiente (automatización) (Yadav, Hong y Stephenson, 2016). En este marco, la programación es considerada como una habilidad fundamental de las ciencias de la computación y una herramienta clave para apoyar las tareas cognitivas involucradas en el pensamiento computacional, así como también una demostración de competencias computacionales (Grover y Pea, 2013).

doi: <http://dx.doi.org/10.15359/ree.23-3.9>URL: <http://www.una.ac.cr/educare>CORREO: educare@una.cr

Brennan y Resnick (2012), interesados en las formas en que la programación de medios interactivos apoyan el desarrollo del pensamiento computacional, plantean tres dimensiones para su análisis (conceptos, prácticas y perspectivas), que estudiaron en torno a la plataforma Scratch, utilizada mundialmente para la enseñanza de la programación (Maloney, Resnick, Rusk, Silverman y Eastmond, 2010). Los conceptos computacionales serían los utilizados al crear programas aplicables a otros contextos, ya sean de programación o no; las prácticas computacionales se utilizan cuando se programa y se centran en el cómo se está aprendiendo y no el qué; y finalmente las perspectivas computacionales son las que el público usuario construye sobre sí mismo y su entorno, gracias a la programación. Considerando este marco, el trabajo de revisión realizado por Lye y Koh (2014) reportó una mayoría de investigaciones centradas en los conceptos computacionales. Sin embargo, es posible relevar algunos estudios que analizan también las prácticas y perspectivas computacionales (Falloon, 2015; Kafai et al., 2013), desarrolladas en la niñez y la juventud.

En líneas generales, el desarrollo del pensamiento computacional a través de actividades de programación es una temática ampliamente estudiada en contextos educativos. Sin embargo, la mayoría de las investigaciones se centran en la evaluación del desarrollo de este mismo por parte del estudiantado, y focaliza en el resultado, en los conceptos, como se señaló previamente (Arranz y Fuente, 2017; Palma y Sarmiento, 2015; Sáez-López, Román-González y Vázquez-Cano, 2016). En contrapartida, el estudio del desarrollo del pensamiento computacional por parte del profesorado es escaso, en cualquiera de sus dimensiones. Sin embargo, se relevan estudios sobre cambios en las actitudes, percepciones o creencias en torno a este concepto luego de instancias de formación, tanto en docentes de computación en educación secundaria como primaria (Heintz, Mannila y Färnqvist, 2016; Rich, Jones, Belikov, Yoshikawa y Perkins, 2017). Se plantea, a su vez, la relevancia de no solo formar al cuerpo docente en servicio como suele ser la estrategia general en estos casos (Yadav, Gretter, Good y McLean, 2017).

La formación docente en programación: Un campo en construcción

Como se anticipaba, para formar al profesorado en contenidos vinculados a la programación, la estrategia más empleada ha sido la formación de docentes de otras áreas temáticas para enseñar estos contenidos, a través de capacitaciones en servicio o programas de desarrollo profesional intensivo (Yadav, 2017). Sin embargo, en los últimos años se observa un alejamiento de las propuestas tradicionales, para generar espacios en los que el cuerpo docente desempeñe un papel activo en el aprendizaje, practicando nuevos métodos pedagógicos. En línea con ello, se afirma que las experiencias efectivas de aprendizaje son producto de la construcción activa de objetos significativos a nivel social o personal (Monjelat, 2017). En este sentido, para participar de la cultura computacional a través de la programación, es necesario contar con herramientas accesibles, actividades y ejemplos personalmente significativos y el apoyo de otras personas (Barr y Stephenson, 2011).



doi: <http://dx.doi.org/10.15359/ree.23-3.9>

URL: <http://www.una.ac.cr/educare>

CORREO: educare@una.cr

Por su parte, la revisión de [Menekse \(2015\)](#) aporta como cualidades de los programas efectivos: la colaboración entre personal educador, el tiempo adecuado para la implementación y la práctica, los métodos de aprendizaje activos para demostrar cómo implementar nuevas prácticas de enseñanza, el énfasis en el conocimiento de contenido pedagógico, el apoyo y seguimiento para docentes y el establecimiento de comunidades de aprendizaje profesional. En el contexto local, [Martínez y Echeveste \(2014\)](#) documentaron una “experiencia de formación docente centrada en construir una visión común sobre la enseñanza de la programación en computación en la escuela secundaria” (p. 1). Las autoras remarcan, además de los puntos señalados que “la discusión sobre el contexto social, económico y político que da lugar a la innovación, el modelado que proveen los docentes pares y las reflexiones didácticas y pedagógicas, parecen ser claves para modificar las creencias previas de los docentes y construir una visión común de la innovación” ([Martínez y Echeveste 2014, p. 35](#)), pues consideran estos aspectos como claves para el diseño de trayectos formativos tecnológicos.

Retomando la discusión sobre la relevancia del contexto, [Apiola y Tedre \(2012\)](#) nos recuerdan que las convenciones de enseñanza y aprendizaje difieren fundamentalmente entre diferentes contextos, continentes y culturas, lo que en algunos casos excluye la opción de importar soluciones pedagógicas del exterior. Por ello resulta necesario contar con investigaciones que examinen prácticas de programación en diferentes contextos, donde sus aprendices presenten distintos trayectos y necesidades.

En síntesis, los estudios sobre el tema revelan una escasez de investigaciones en torno a las prácticas del pensamiento computacional desarrolladas en los trayectos formativos docentes; el eje de las investigaciones ha sido el estudio de las actitudes, percepciones o aproximaciones al contenido conceptual, con algunos trabajos que también señalan la importancia del conocimiento pedagógico. Por otro lado, la bibliografía consultada revela diferentes lineamientos para el diseño de trayectos de formación docente, y destaca la importancia de una formación significativa, contextualizada y situada, como motor de procesos de enseñanza exitosos.

Marco metodológico

Considerando la escasez de estudios dedicados a las prácticas computacionales en contextos de formación docente, se realizó un estudio de caso exploratorio siguiendo técnicas etnográficas ([Yin, 2003](#)), con el objetivo de proporcionar una investigación inicial que aporte posibles lineamientos para el diseño de experiencias de formación docente en pensamiento computacional desde una perspectiva situada, significativa y compleja. Asimismo, se trata de un estudio intrínseco ([Stake, 1995](#)) que toma como caso una experiencia piloto de formación docente innovadora. Se trata de un taller de formación docente en programación y pensamiento computacional, para crear TIS como proyectos interdisciplinarios dentro de la educación

doi: <http://dx.doi.org/10.15359/ree.23-3.9>URL: <http://www.una.ac.cr/educare>CORREO: educare@una.cr

primaria. En su diseño se tomaron como premisas la importancia de ofrecer al equipo docente propuestas de formación significativas, que se articularan con sus saberes y áreas disciplinares de trabajo y no solo se centraran en la adquisición de conceptos computacionales.

Contexto y participantes

Para la selección de la muestra, se buscó una institución que contara tanto con nivel primario de escolaridad como con un instituto de formación docente, de manera que se pudiera trabajar con docentes en formación y en servicio dentro de la educación básica y obligatoria. Asimismo, se tuvo en cuenta la participación en el proyecto nacional argentino “Primaria digital” que entrega “Aulas digitales móviles” (ADM) a las escuelas. Considerando estos aspectos, se seleccionó una escuela de gestión pública que contara a su vez con un instituto de formación docente. En entrevistas previas al taller, su personal directivo mostró un alto grado de interés en desarrollar proyectos innovadores que supusieran intercambios con otras instituciones de la ciudad. Señaló, a su vez, que el profesorado no había recibido ninguna capacitación a través de los programas nacionales o provinciales, pese a que la escuela contaba con una buena infraestructura tecnológica. Por eso, el cuerpo directivo destacó la necesidad de ofrecer algún tipo de formación que permitiera aprovechar los recursos tecnológicos disponibles. Este contexto institucional permitió formalizar una primera propuesta de formación en formato taller presencial con apoyo virtual que otorgara puntaje a las personas participantes, según la legislación provincial en la materia.

La experiencia se desarrolló en el segundo semestre del año 2016, durante 8 sesiones presenciales, de 3 horas cada una, que tuvieron lugar tanto en horario escolar como fuera de este. Participaron en total 13 docentes del nivel primario, 6 en formación y 7 en servicio que asistieron voluntariamente y se organizaron en 4 grupos. Del total de participantes, 12 no tenía experiencia previa en programación, el restante manifestó conocimientos previos de Logo. Ninguno tenía conocimientos previos de Scratch, la herramienta de programación con que trabajaron y que fue escogida por su reconocimiento a nivel mundial y su interfaz amigable. El grupo de participantes en servicio contaban en promedio con más de 9 años de formación, y en su mayoría se encontraban a cargo de las materias específicas, a excepción de una docente de inglés y otro de tecnología.

Durante los encuentros presenciales, la dinámica de trabajo apuntó a la reflexión conjunta, el diálogo y el intercambio, se focalizó el trabajo de las personas participantes en la exploración tanto de la herramienta Scratch como de la problemática sobre la que versaría cada proyecto (Monjelat, 2017). A lo largo de las sesiones se ofrecieron explicaciones sobre cómo utilizar este entorno de programación con bloques, crear objetos y escenarios, tipos de bloques, entre otras temáticas básicas necesarias para trabajar con el recurso. En función de las características de



doi: <http://dx.doi.org/10.15359/ree.23-3.9>

URL: <http://www.una.ac.cr/educare>

CORREO: educare@una.cr

cada proyecto (temática, personajes, escenas) y del tipo de producción que se buscara crear (historia, animación juego, etc.) se trabajaron contenidos más específicos. Por ejemplo, algunos proyectos al incluir sonido trabajaron en más detalle los bloques relacionados con ello y otros que requerían una edición más detallada de imágenes, dedicaron más tiempo a explorar esta opción. Se contó a su vez con un espacio de trabajo en línea, con documentación ampliatoria y espacios para el intercambio y la resolución de dudas y dificultades.

Recolección y análisis de datos

Durante el estudio se recogieron datos en diferentes formatos y momentos. Tanto al comienzo como al final del taller se administraron encuestas voluntarias. La primera para caracterizar al plantel docente de la institución y ajustar la propuesta del taller y la segunda, para recoger una valoración de lo realizado. Se recogieron también las producciones elaboradas por los docentes y las docentes participantes durante el taller, y la versión final entregada junto con un informe al término de la experiencia. Asimismo, se recolectaron las interacciones ocurridas en el entorno virtual que funcionó como instancia no presencial del taller (mensajes producidos en foros, enlaces a producciones en curso y documentos adjuntados donde se planteaban síntesis de lo realizado o avances). Para posibilitar la reconstrucción del proceso de creación de TIS en mayor detalle, se grabaron las sesiones presenciales en audio y video, buscando capturar prácticas y discursos. Estas luego fueron transcritas con la ayuda del software O Transcribe. Los nombres de las personas participantes han sido modificados y los datos de la institución han sido eliminados para proteger su anonimato.

Se siguió la perspectiva de estudio de caso (Yin, 2003) y el análisis del discurso (Gee y Green, 1998) para abordar los datos y estudiar las prácticas computacionales desarrolladas durante la experiencia formativa. Para ello, se siguieron dos fases: la narrativa, donde los momentos más significativos del taller fueron identificados usando códigos de tiempos y reconstruyendo lo acontecido en cada sesión; y la fase analítica, donde las transcripciones se clasificaron en un sistema de categorías. Para el análisis e interpretación de los datos se contó con la ayuda del software Atlas Ti.

Categorías analíticas

En la fase narrativa del análisis, los datos revelaron temáticas emergentes a partir de la identificación de episodios en las diferentes fuentes de datos recolectados donde los maestros y las maestras ponían en juego prácticas del pensamiento computacional mientras creaban colaborativamente sus TIS. Este tipo de prácticas suelen ser estudiadas en contextos educativos, como se señaló en los antecedentes, pero su estudio en formación docente no está extendido. Por ello y a su vez para profundizar en el conocimiento del proceso de construcción de TIS, se

doi: <http://dx.doi.org/10.15359/ree.23-3.9>

URL: <http://www.una.ac.cr/educare>

CORREO: educare@una.cr

planteó su estudio a lo largo de las sesiones que comprendieron el taller, focalizando el análisis en el material discursivo obtenido a partir de las transcripciones y en el estudio pormenorizado y complementario de las demás fuentes. Este tipo de análisis permitió estudiar en profundidad cómo los maestros y maestras se acercan a nociones del pensamiento computacional, centrándose en el cómo, en un contexto de creación específico. A continuación, se analizó el material recolectado siguiendo las categorías presentadas en la [Tabla 1](#), previamente identificadas en la bibliografía sobre el tema a partir de revisiones teóricas.

Tabla 1: Prácticas del pensamiento computacional

Práctica y definición	Estrategias
<i>Experimentar e iterar:</i> Desarrollar un poco, probar y luego seguir desarrollando	<ul style="list-style-type: none"> • Construir un proyecto paso a paso • Probar cosas a medida que avanza • Realizar revisiones basadas en lo que ocurre.
<i>Testear y depurar:</i> Lograr que las cosas funcionen. Encontrar y resolver problemas cuando surjan	<ul style="list-style-type: none"> • Observar qué ocurre cuando ejecutas un programa • Describir qué es diferente de lo que esperas • Leer los programas e investigar la causa del problema • Realizar cambios y testear qué ocurre
<i>Reusar y remezclar:</i> Crear con base en proyectos o ideas existentes	<ul style="list-style-type: none"> • Encontrar ideas e inspiración al probar otros proyectos y leer sus programas • Seleccionar una parte de otro proyecto y adaptarlo al propio • Modificar un proyecto existente para mejorarlo • Otorgar reconocimiento a las personas cuyos trabajos te inspiraron o permitieron construir el tuyo
<i>Abstraer y modular:</i> Explorar las conexiones entre el conjunto y las partes	<ul style="list-style-type: none"> • Decidir qué objetos son necesarios para tu proyecto y dónde deben ir • Decidir qué programas son necesarios para tu proyecto y qué deberían hacer • Organizar los programas para que tengan sentido para ti y para las demás personas

Nota: Adaptado de [Resnick y Brennan \(2011\)](#) y [Brennan y Resnick \(2012\)](#).

Siguiendo este modelo, se analizaron las transcripciones, los informes finales, el material recolectado a través de la plataforma virtual y las producciones creadas.

Resultados y discusión

Las prácticas computacionales que aquí se describen tuvieron lugar en un contexto específico de formación docente: un taller de creación de tecnologías para la inclusión social. Por ello, los primeros resultados permiten situar las prácticas y comprender el marco en el que



doi: <http://dx.doi.org/10.15359/ree.23-3.9>

URL: <http://www.una.ac.cr/educare>

CORREO: educare@una.cr

tuvieron lugar. Estas fueron identificadas en los materiales analizados, por lo que considerando la extensión de este trabajo, a continuación se presentan y discuten ejemplos representativos de la generalidad de las prácticas computacionales puestas en obra por el grupo participante durante el taller, para luego finalizar presentando las conclusiones del trabajo.

El eje de los proyectos: Problemáticas situadas

En línea con las recomendaciones relevadas, se diseñó un trayecto formativo que pudiera ser situado, contextualizado y significativo. Por ello, las primeras sesiones del taller fueron destinadas a la generación de un espacio de intercambio que permitiera revelar problemáticas contextualizadas, que luego sirvieran de base para diseñar los proyectos grupales utilizando Scratch (Monjelat, 2017). En la primera sesión del taller, la consigna propuso una reflexión a partir de una problemática disparadora, que en este caso fue el concepto de “ciudadanía”. Se solicitó al grupo de participantes definir este concepto, primero en forma individual y luego en pequeños grupos. En los intercambios de este primer encuentro surgieron reconocidas obras argentinas (por ej. la canción “Septiembre del 88” del cantante Cacho Castaña, la obra del pintor de la Cárcova “Sin pan y sin trabajo”, el Himno nacional). También se mencionaron costumbres nacionales como el mate, insignia de la cultura argentina, vinculado a la idea de pertenencia, participación y comunidad. Por otra parte, se observó en sus discursos un predominio de conceptos vinculados a la Constitución, al orden y a los derechos ciudadanos.

A partir de esta puesta en común, en la segunda sesión se trabajó en grupos definiendo los temas sobre lo que se realizarían los proyectos. Preguntas como: ¿qué implica ser un ciudadano o ciudadana en nuestro contexto? funcionaron como puntapié inicial para reflexionar sobre posibles temáticas. La Figura 1 recopila algunos discursos que refieren a estas. Como se puede observar en los diferentes fragmentos, los temas que se plantean tienen una estrecha relación con la realidad de las prácticas docentes. Son temáticas significativas para los maestros y las maestras, ancladas en ejemplos concretos y cotidianos, que permiten situar la experiencia de programación en el marco de una propuesta contextualizada.

Los grupos 1 y 2 exponen situaciones presentes en la mayoría de las escuelas públicas argentinas: el maltrato de los inmuebles y objetos escolares. Asimismo, rescatan situaciones que desde la esfera institucional surgen como propuestas para trabajar estas cuestiones, como la creación de un mural colectivo o el uso de cartucheras comunitarias. De esta forma, las problemáticas reflejan como temática transversal la idea de “respeto y cuidado”: de la propiedad colectiva (Grupo 1 y 3), del otro individuo en cuanto ser humano (Grupo 2) y en cuanto familiar (Grupo 4). En este sentido, las situaciones reales de la práctica funcionan como ejes para diseñar proyectos que les permitan trabajar estas cuestiones en situación de clase con sus estudiantes. El paso siguiente en el taller ha sido llevar estas temáticas a proyectos concretos plausibles de ser

realizados en Scratch. Como se anticipó, en ese proceso se desplegaron diferentes prácticas del pensamiento computacional que, como veremos en los siguientes apartados, se entrelazan con el desarrollo de las temáticas propuestas por cada grupo.

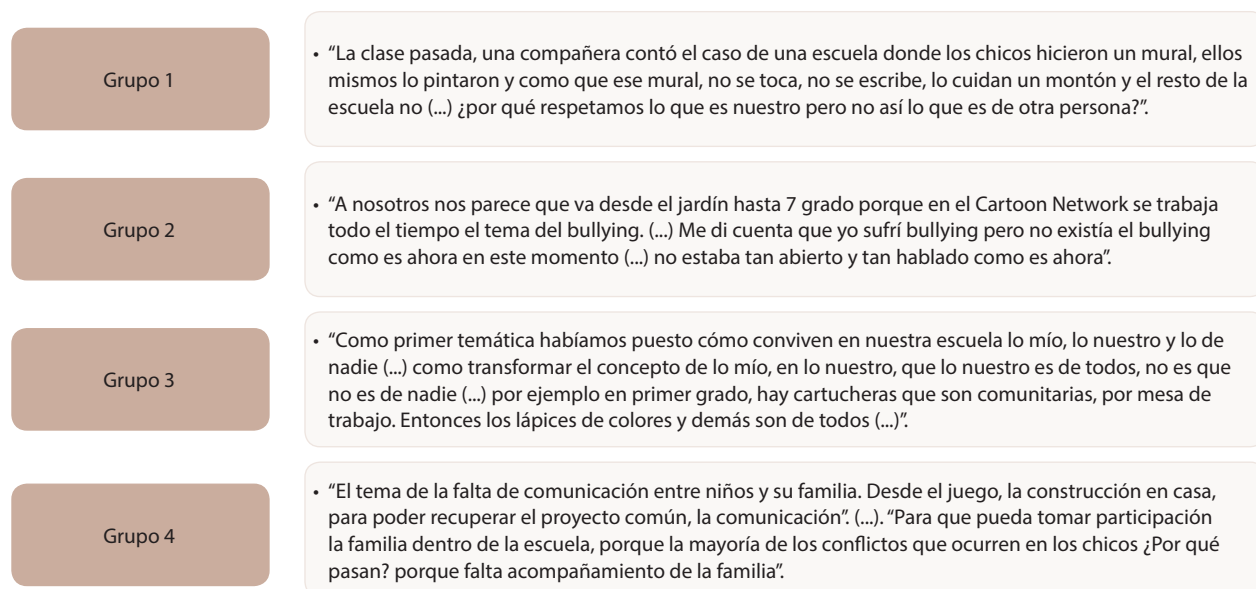


Figura 1: Extractos discursivos que reflejan temáticas situadas.

Nota: Elaboración propia.

Analizando el proceso: Prácticas computacionales en la creación de TIS

El proceso de diseño y programación de los proyectos tuvo lugar desde la primera sesión hasta la séptima y durante las mismas el grupo participante implementó diferentes estrategias que reflejan la puesta en obra de prácticas computacionales. La sesión final se dedicó a la presentación de trabajos. Al realizar el análisis fue posible identificar una articulación de las diferentes prácticas computacionales que funcionaron como categorías de estudio. Por ello, a continuación se presentan ejemplos de estas articulaciones, que permiten dar cuenta de la complejidad implicada en su desarrollo.

Avanzando de forma gradual e iterativa, mezclando y reutilizando

Los proyectos propuestos fueron rediseñados en múltiples oportunidades, fundamentalmente para refinar el abordaje de la problemática inicial, pero también para poder generar un prototipo funcional dentro de los plazos del taller. En este sentido, todos los grupos desplegaron prácticas de iteración, probando diferentes cuestiones a medida que avanzaban y realizando revisiones a partir de lo ya creado. Las iteraciones se produjeron fundamentalmente en torno a la creación de objetos y escenarios, en línea con la problemática a abordar y,

doi: <http://dx.doi.org/10.15359/ree.23-3.9>

URL: <http://www.una.ac.cr/educare>

CORREO: educare@una.cr

consecuentemente, sus programas fueron mutando junto con las temáticas propuestas. En cuanto a los objetos, los grupos tres y cuatro emplearon personajes presentes en las aulas de primero y segundo grado, lo que también contribuye a la contextualización de la propuesta, como se puede observar en el siguiente fragmento:

Fragmento 1. Personajes y escenarios: planteo inicial del proyecto. Grupo 3. Sesión 3

Carlos: Te muestro a Filipo y a Nina, que son dos animales que ya vienen desde el primer grado con el grupo

Investigadora: Y esos van a ser los dos personajes...

Armando: Sí, son un león y una cebra

C: Son dos títeres, que se escriben cartas entre ellos, del C al B

A: Son significativos porque son los personajes del salón.

I: O sea, los personajes que van a estar juntando la basura son los personajes que están en el salón

A: Claro, ellos ya los conocen, ya circulan.

Sonia: Tienen un títere cada uno

A: Entonces lo que vamos a hacer es sacarle fotos a los títeres en las posiciones en que el videojuego necesita, de manera de poder introducirlo como objeto, pero que sea la foto del personaje

Como se observa en el fragmento, los objetos a programar son conocidos por el profesorado y por eso han sido escogidos como los personajes que, en esta primera versión del proyecto, estarán juntando basura. En la versión final del proyecto solo se utilizó un objeto (el león) y recurrieron a una imagen encontrada en internet, que editaron dentro de Scratch. El uso de los personajes del salón también fue la estrategia escogida por el grupo 4, pero cobrando otro valor, como se observa en el siguiente fragmento:

Fragmento 2. Planteo inicial del proyecto: personajes. Grupo 4. Sesión 3

Paola: [Lee lo que han escrito entre todas] un diálogo entre animales que se preguntan sobre datos que no pueda responder el niño sin la ayuda de un familiar, para generar el vínculo frente al recurso... la jirafa le pregunta al elefante, porque está la jirafa, el elefante y la cebrá ...

Eliana: Con esto de la familia ensamblada

Analía: No sabíamos si era padre, madre, entonces dijimos, animal

doi: <http://dx.doi.org/10.15359/ree.23-3.9>

URL: <http://www.una.ac.cr/educare>

CORREO: educare@una.cr

En este caso, utilizar los personajes del salón sirve al grupo para poder reflejar cualquier tipo de familia, considerando el eje fundamental de la propuesta que es facilitar la comunicación entre la niña o el niño, su familia y la escuela. En la versión final, este grupo utilizó el elefante, tomándolo de la biblioteca de Scratch.

Por otra parte, se observa que los objetos y escenarios en su mayoría no han sido elaborados por los grupos participantes. Son excepciones el grupo 1, que creó uno de sus objetos en otro programa de edición de imágenes y el grupo 2 que utilizó como escenarios fotos tomadas por las participantes en la escuela. Sin embargo, la estrategia de reutilizado de creaciones elaboradas por otras personas se observa en todos los grupos, ya sea para los objetos, fondos o ambos. En este sentido, maestros y maestras recurrieron a las bibliotecas de Scratch o tomaron imágenes alojadas en internet para ilustrar sus personajes y escenarios. El grupo 1 en su informe final señalaba esta situación, mostrando en este caso una conjugación de fuentes que se articularon para la creación de objetos y escenarios, como muestra la [Figura 2](#):



Figura 2: Creación de objetos y escenarios. Múltiples estrategias.

Nota: Elaboración propia.

El reutilizado y mezclado también se evidenció en el uso de la "mochila", un atributo del programa en su versión online que posibilita el guardado tanto de partes del código como de objetos y escenarios para que luego puedan ser incluidos en las producciones personales. Con ello superaron un obstáculo: la imposibilidad de trabajar a la vez en el mismo proyecto. Estas prácticas de reusar y remezclar a través del uso de la mochila, se evidencian en el siguiente fragmento:

doi: <http://dx.doi.org/10.15359/ree.23-3.9>

URL: <http://www.una.ac.cr/educare>

CORREO: educare@una.cr

Fragmento 3. Reinención: uso de la mochila. Grupo 3. Sesión 3

A: Nosotros lo que empezamos a hacer la semana pasada, el encuentro pasado es, con todos los comandos, cosa por cosa, hacer caminar al león, que llegue a un determinado lugar, fue realmente bastante complejo. Lo que hicimos fue, buscar un videojuego ya armado dentro de la página

I: Ah eso que yo les decía de la mochila

C: Empezamos a operar...

A: Empezamos a cambiar, encontramos otra cosa, en vez de ir levantando, encontramos uno que vaya cayendo ya la basura. En realidad eran manzanas en el original y tenía otro fondo, de a poquito lo vamos reformando hacia lo que nosotros queremos

C: Alteramos bueno, la secuencia, dibujamos al leoncito

A: Se puso basura en vez de manzanas

S: El cesto ese va a ser el personaje

En líneas generales, los grupos utilizaron tanto la mochila como la reinención como estrategias para compartir los proyectos o algunas de sus partes entre integrantes del grupo, como se observa en la [Figura 3](#):



Figura 3: Re-inversiones de proyectos.

Nota: Archivo personal.

El profesorado reinventó proyectos realizados por integrantes de los grupos, buscando con ello continuar trabajando fuera de las clases presenciales o ampliar el trabajo realizado por algún compañero o compañera en una clase anterior. Asimismo y como remarcaba el fragmento 3, se tomaron partes de proyectos alojados en la web, que de esta forma, no aparecen referenciados en los proyectos finales.

Abstraer y modular: Definiendo los proyectos

A lo largo de los encuentros, las temáticas propuestas inicialmente fueron concretándose en diferentes proyectos. Como se observa en la [Figura 4](#), que rescata intercambios producidos durante la sesión tercera, los grupos fueron definiendo qué tipo de proyectos querían realizar: animaciones, juegos interactivos, video clip, etc. Al hacerlo, decidieron qué objetos eran necesarios en sus proyectos, identificando escenas y organizando los programas de manera que tuvieran sentido para ellos y ellas.

Grupo 1	<ul style="list-style-type: none"> • “Pensamos que haya 3 chicos en un salón, que ellos hayan colgado sus dibujos y entonces viene un nene malo y le rompe el dibujo a uno, o lo hace un bollo, se lo tira al piso y el dueño del dibujo se pone a llorar y el tercero se lo levanta, se lo abre y como que dice que eso no se hace, como un mensaje”.
Grupo 2	<ul style="list-style-type: none"> • “Lo que nosotros queremos hacer es un video clip donde haya chicos que canten un rap sobre la temática del Bullying, un rap que le vamos a pedir a los chicos que hagan para nuestro trabajo (...) se reúnen 3 niños en la puerta de la escuela, uno le comenta al otro que había visto a una compañera llorando en la escuela y a raíz de eso empiezan a rapear sobre esta problemática”.
Grupo 3	<ul style="list-style-type: none"> • “Lo primero que pensamos fue algo que tuviera que ver con la animación y ahí había que partir de un relato (...) la temática de formación ética que queremos trabajar era cuidado de los espacios que son comunes y qué pasa con estos elementos, como las cartucheras comunes, que como son comunes, no son de nadie y son de todos, excepto míos (...)”.
Grupo 4	<ul style="list-style-type: none"> • “Queremos que los chicos se presenten frente al recurso con la necesidad de un adulto, entonces es una escena donde hay 3 animales (...) está una jirafa y le pregunta al elefante cuál es su comida favorita. El elefante tiene que responder y la cebrá, que va a dar una prenda en el contexto del juego (...) le va a indicar que siga jugando al juego o no”.

Figura 4: Propuestas de proyectos en la sesión tercera del taller.

Nota: Elaboración propia.

Estas estrategias vinculadas a las prácticas de abstracción y modularización se reflejaron en diferentes momentos, por ejemplo, en cuanto a la idea de agrupar bloques según sus funciones e incluso volver a reutilizar ese código. Un ejemplo de ello es el fragmento 4:

doi: <http://dx.doi.org/10.15359/ree.23-3.9>

URL: <http://www.una.ac.cr/educare>

CORREO: educare@una.cr

Fragmento 4. Reinención: Uso de la mochila. Grupo 4. Sesión 6

A: Van a ser 4 historias, claro, sí. Porque vamos a trabajar todo lo que es el árbol genealógico con lo que conlleva eso. Ahora mi pregunta es, nosotros queremos hacer 4 placas que estén en el mismo proyecto. ¿Cómo hacemos?

I: A ver, nosotros precisamos, vamos a tener 4 fondos, ¿4 eran no?

A: Sí

I: En todos los fondos va a haber el mismo Chispita? [Chispita es el elefante, objeto-personaje]

A: No, lo vamos a cambiar de color y eso

I: Entonces vamos a tener 4 imágenes de fondo y un objeto. Lo que tenemos que hacer es crear una serie de eventos que nos controle que ésto aparezca, que después se oculte...

A: Ahh como está hecho con el elefante.

En este extracto se observa cómo un participante del grupo 4 recupera acciones que ya ha realizado para poder resolver un problema actual, aunque en un principio no implementa esta práctica de forma autónoma, los intercambios discursivos le permiten recuperar los aprendizajes previos y ponerlos en valor. En una primera instancia al programar el personaje se crearon una serie de eventos que ordenaron la secuencia, ahora para programar los escenarios es preciso retomar ese grupo de bloques y adaptarlo a los cambios actuales. Esta forma de abstracción presente en los datos también refiere al uso de bloques de programas en diferentes objetos que tienen un comportamiento similar, práctica empleada por los grupos en diferentes momentos de creación. El grupo 1, por ejemplo, empleó esta estrategia para generar las gotas de lluvia que inundan el patio de la escuela, como se observa en la [Figura 5](#).

El grupo de participantes tomó el mismo bloque de programas y cambió valores para lograr diferentes efectos en las gotas, alterando el movimiento y el momento de aparición de cada uno de los objetos. Asimismo, el grupo 2 empleó la misma estrategia para el movimiento de sus estudiantes por los pasillos de la escuela al rapear y el grupo 3, para programar el accionar de los objetos que debían recogerse y guardarse en las cartucheras comunitarias.

Las prácticas de abstraer y modular están presentes mayormente en las sesiones más avanzadas del taller y pudieron identificarse en las producciones finales, cuando los maestros y las maestras participantes lograron definir con mayor precisión sus proyectos. En relación con ello, algunos grupos modificaron sus temáticas casi hasta el final del taller, mientras a su vez testeaban y depuraban sus programas.

doi: <http://dx.doi.org/10.15359/ree.23-3.9>
 URL: <http://www.una.ac.cr/educare>
 CORREO: educare@una.cr

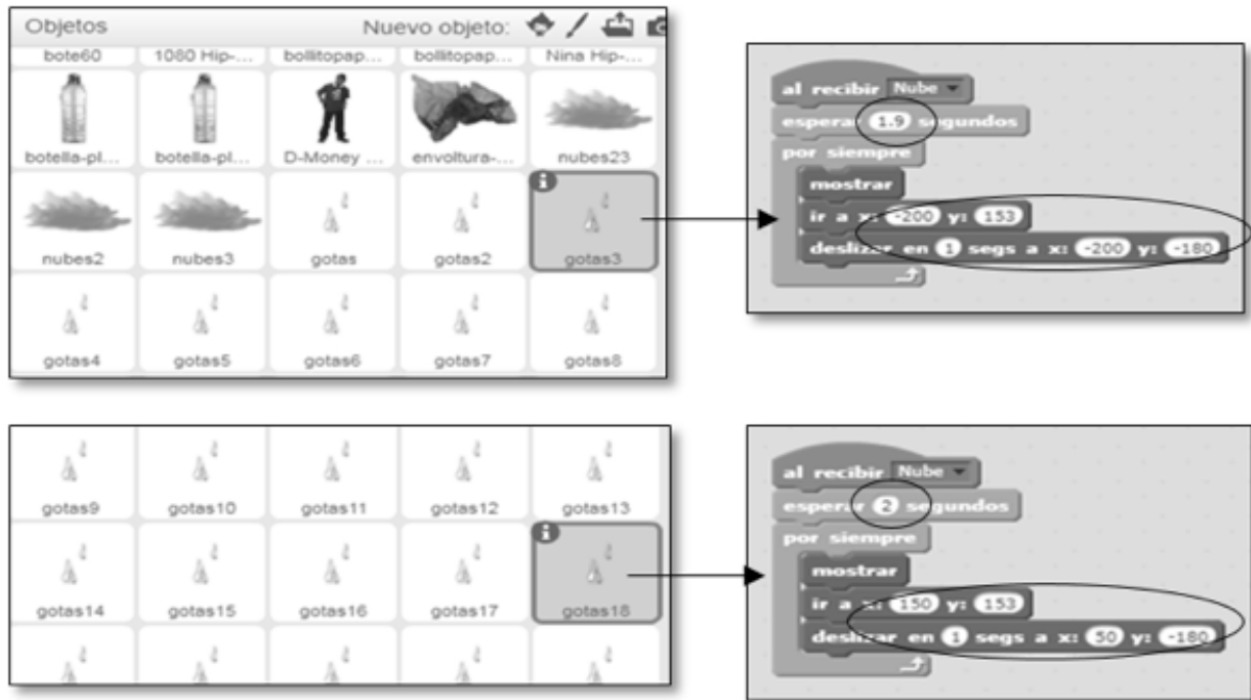


Figura 5: Bloque de programas utilizado en diferentes objetos.
Nota: Elaboración propia.

Testear y depurar programas y problemáticas

Al programar con Scratch y considerando que nadie, del grupo participante contaba con experiencia previa con esta herramienta, surgieron diferentes interrogantes y problemas a resolver. El análisis del material recolectado permitió identificar que frente a un problema, la estrategia más utilizada ha sido el andamiaje tanto de otros u otras miembros del grupo como de la docente del taller, que tomó la forma de preguntas puntuales tanto en las sesiones presenciales como a través de la plataforma virtual, con el envío de mensajes personales a la tutora.

Una dificultad que predomina en los datos es la existencia de código previo no eliminado, que altera el correcto funcionamiento del programa, como se observa en el fragmento 5:

Fragmento 5. Testar y depurar. Grupo 2. Sesión 7

Soledad: Mostrale

l: ¿Se les tilda la música?

Laura: Sí... [Escuchamos la canción de rap del proyecto]

doi: <http://dx.doi.org/10.15359/ree.23-3.9>

URL: <http://www.una.ac.cr/educare>

CORREO: educare@una.cr

I: ¿Está compartido? ¿Cómo la pusieron para que se reproduzca a ver? porque por ahí....

L: Como sonido

Violeta: Cargué el sonido, busqué la instrucción y lo seleccionamos

I: Pero ahí tenés dos

V: Este es el que habíamos cargado primero pero como se escuchaba mal lo cargamos de vuelta, es el mismo

I: Por eso, si es el mismo, cuando lo llamas para que se reproduzca, llámás a los dos. Borren uno, y mostrame a ver dónde está el bloquecito para que se reproduzca, ¿está puesto solo? está puesto en algún...

L: Tenés que hacer click en el primer personaje, no, ese no es.

I: Si no, andá abajo a los objetos y vamos pasando uno por uno. Ese no tiene nada, ese tiene que cambiar al siguiente disfraz, este también... ¿Estará en el fondo?

L: Yo lo había puesto en un personaje me acuerdo...pero ahí está

S: Sí, ahí está.

En este caso, no resulta evidente que contar con bloques u objetos duplicados pueda afectar el programa, pero al explorar lo realizado es posible investigar la causa del problema, realizar cambios y testear qué ocurre. Otra estrategia utilizada ha sido observar qué ocurre cuando se ejecuta un programa, para planificar cómo seguir y qué se ha realizado hasta el momento.

Fragmento 6. Testar y depurar. Grupo 3. Sesión 6

Investigadora: ¿Pudieron hacer que se mueva y eso?

Sonia: Nos queda que junte ahora, los lápices

Carlos: Que al tocar los lápices...bueno, esto viene así [ejecuta el programa]

I: [Lee en voz alta] "...y no guardo los útiles. Ayuden por favor a ordenar este salón"

C: Tiene el sonidito que no se escucha y demás. Entonces, eso desaparece en 10 segundos y tenemos acá la movilidad

I: O sea, que ese tendría que aparecer después, para no estar repetido

doi: <http://dx.doi.org/10.15359/ree.23-3.9>

URL: <http://www.una.ac.cr/educare>

CORREO: educare@una.cr

C: Lo desaparecemos, dale. Pudimos hacer que se mueva, se mueve y gira, que yo quería que gire. Si, tiene un poquito más de onda ... entonces bueno, la idea es poner otro objeto, no tienen por qué ser lápices, pusimos bufandas, otras cosas, y que a la hora de arrastrar y que toque acá tendría el lapicito...

S: Meterse ahí adentro

C: No sé si con los sensores, con los colores, no sé cómo sugerís vos.

I: Tocar objeto puede ser

C: Bien, ¿y eso está en los sensores?

I: Sí, tocando objeto. Es preferible en vez de que programes el león para que cada vez que toque un lápiz, que programes el lápiz...

C: Y después copio y pego...genial.

En el fragmento 6, integrantes del grupo muestran lo que han realizado y dialogan sobre cómo continuar programando. De esta forma, testean el código ya creado y, con base en ello, se toman decisiones o se realizan los cambios necesarios, depurando la programación y ajustando el proyecto a la temática propuesta. En este caso, nuevamente se observa un entrelazamiento de prácticas, que articulan el testeo y depuración con la reutilización de código.

Analizando el resultado: Tecnologías para la inclusión social creadas con Scratch

Al poner en juego las prácticas descritas en relación con el pensamiento computacional, los grupos participantes crearon diferentes producciones hipermediales. A partir del análisis de los informes finales y las narrativas de la sesión final de presentación de proyecto, se identificaron las tramas y problemáticas en su última versión, así como posibles propuestas pedagógicas y articulaciones curriculares, que se reseñan en la [Figura 6](#).

Estas creaciones pueden considerarse tecnologías para la inclusión social en la medida en que han sido diseñadas, por el profesorado que será su usuario, respondiendo a problemáticas propias. Los proyectos creados no fueron impuestos ni surgieron de la reproducción de tutoriales, guías o manuales; por el contrario, son el resultado de un complejo proceso donde el eje ha sido una temática local, promueven prácticas de aprendizaje respetuosas y reconocedoras de las particularidades de cada contexto. Los proyectos programados, en cuanto TIS en el contexto educativo, pueden funcionar como recursos didácticos en el marco de propuestas pedagógicas situadas como las que describe la figura. De esta forma, la conexión con contenidos curriculares propios del nivel educativo de los grupos participantes se realiza de manera sencilla, sin forzarse o imponerse.



doi: <http://dx.doi.org/10.15359/ree.23-3.9>

URL: <http://www.una.ac.cr/educare>

CORREO: educare@una.cr



Grupo 1

- *Problemática:* respeto por los espacios públicos
- *Trama:* el acto individual de tirar un papel en el recreo provoca una inundación del espacio cuando llueve.
- *Propuesta pedagógica:* sirve como disparador para trabajar valores ciudadanos en el marco del cuidado del medio ambiente enfatizando pautas para el reciclado. Identificar otras situaciones similares, elaborar campañas de concientización dentro de la comunidad escolar.



Grupo 2

- *Problemática:* violencia escolar
- *Trama:* alumnos rapean su mensaje anti bullying por los pasillos de la escuela.
- *Propuesta pedagógica:* involucrar a los alumnos en la creación de mensajes anti violencia en diversos formatos artísticos. Trabajar redacción de canciones, versos, tipos de géneros musicales, su historia, contextos de producción. Transmitir estos mensajes a la comunidad educativa en su conjunto.



Grupo 3

- *Problemática:* relación escuela-familia
- *Trama:* los alumnos recuperan escenas y recuerdos familiares elaborando un collage en ayuda de otro miembro de su familia. Se proponen consignas cuyas respuestas requieren una colaboración familiar.
- *Propuesta pedagógica:* Se propone esta actividad como situación disparadora que permitan recuperar en el aula las diferentes historias familiares, previo a una convocatoria para celebrar el día de la familia. Articulación con el estudio del barrio y la escuela en primeros años de primaria.



Grupo 4

- *Problemática:* lo tuyo, lo mío, lo nuestro
- *Trama:* la mascota del aula solicita ayuda para ordenar diferentes elementos de uso común en el aula. Al juntar los elementos se ganan puntos.
- *Propuesta pedagógica:* un sencillo juego propone reflexionar sobre el cuidado de los útiles de uso común en el aula, particularmente de la cartuchera comunitaria. Trabajo sobre valores comunitarios, colaboración, ética y ciudadanía, responsabilidad y derechos ciudadanos, etc.

Figura 6: Características de las tecnologías para la inclusión social creadas en el marco del taller.

Nota: Elaboración propia.

Conclusiones

El presente artículo se planteaba como objetivo estudiar cómo se presentan las prácticas del pensamiento computacional en un contexto de formación docente situado, así como también investigar la articulación de estas prácticas con una propuesta de creación de TIS.

doi: <http://dx.doi.org/10.15359/ree.23-3.9>URL: <http://www.una.ac.cr/educare>CORREO: educare@una.cr

Con respecto a las prácticas del pensamiento computacional, los resultados evidencian su presencia a lo largo del taller, estas se presentan de diversas maneras, pero siempre entrelazadas. Esto es un dato relevante, ya que refleja un entramado complejo que permite la puesta en obra de diferentes acciones necesarias para la realización de un proyecto. Como mostraron los ejemplos, la iteración se conjuga con el *remixado* y la reutilización; pero también con el testeo y la depuración; de la misma forma la abstracción se presenta junto a la reutilización y así en los demás ejemplos. Estos resultados ponen de manifiesto la importancia, ya destacada por investigaciones con niños, niñas y jóvenes, de trabajar todas las prácticas en conjunto cuando se diseñan trayectos formativos, aunque en algunos momentos puntuales se centre el foco en una de ellas en particular. Justamente, es la articulación de las prácticas lo que permite avanzar en la programación y en el desarrollo de la temática. Sin embargo, los resultados obtenidos evidencian que resulta necesario generar una identificación de estas prácticas por parte de las participantes y los participantes para lograr una mayor autonomía, aspectos que debería incorporarse en futuras formaciones.

En relación con la articulación de las prácticas con la propuesta de creación, los datos mostraron cómo no es posible desligar estos aspectos. El recorrido desde la idea inicial al prototipo final expone iteraciones, avances graduales, *remixados*, reutilizaciones, testeos que se imbrican en la problemática. Por otro lado, la problemática otorga sentido a las prácticas y permite que estas mismas cobren valor al estar enmarcadas en un proyecto significativo para sus participantes. Los cambios, avances y nuevas propuestas que se introducen a través de la programación no son porque sí, permiten definir el abordaje de la problemática que se quiere reflejar. En este sentido, se observa que aunque el personal docente participante logró crear producciones en el lapso de duración del taller, se corrobora lo planteado por otros estudios en cuanto a la necesidad de plantear este tipo de trayectos formativos con una mayor duración en el tiempo, en vez de plantear cursos intensivos, o incluso sin momento de cierre, como espacio continuo de intercambio y producción. Sin embargo, cuando el contexto adopta características de la educación formal, como en este caso, los tiempos y plazos para realizar el proyecto condicionan su desarrollo, pues la factibilidad es un aspecto que puede limitar las posibilidades tanto de avanzar en la programación como en la elaboración de la trama que refleja el producto hipermedial a crear. Revisar los formatos de las propuestas de formación aparece, entonces, como una imperiosa necesidad.

Por otra parte, los resultados evidenciaron que es posible el desarrollo de prácticas computacionales en el marco de una formación docente significativa, compleja y situada, que no solo se centre en la apropiación de conceptos computacionales o en el aprendizaje de una tecnología específica tal como se destaca en la generalidad de las propuestas formativas relevadas. Asimismo, esta estrategia didáctica innovadora resulta apropiada para trabajar con docentes de educación primaria, que abordan diferentes contenidos curriculares no necesariamente relacionados con cuestiones vinculadas a la computación, destacando el valor de los aspectos pedagógicos y los saberes con los que cuenta el profesorado, cuestiones



doi: <http://dx.doi.org/10.15359/ree.23-3.9>

URL: <http://www.una.ac.cr/educare>

CORREO: educare@una.cr

valoradas por otros estudios al momento de diseñar trayectos formativos. De esta manera, el presente estudio aporta resultados en la línea de investigaciones previas que proponen formar a docentes (en servicio y en formación), desde propuestas contextualizadas y ejemplos significativos en un marco comunitario y colaborativo de reflexión e intercambio.

El abordaje de la enseñanza de la programación se plantea aquí como un proceso más cercano a la realidad cotidiana de cada docente, poniendo de relieve la importancia de generar prácticas educativas mediatizadas que sean, a su vez, situadas. Diseñar un trayecto formativo para crear TIS aparece, en este sentido, como una opción válida para la enseñanza de la programación y el desarrollo del pensamiento computacional, que como se ha mencionado, resulta una habilidad imprescindible en la sociedad actual.

En otro orden de cuestiones, el análisis de las prácticas en un contexto situado se vería enriquecido por el estudio complementario de conceptos y perspectivas, que podrían ser objeto de futuros trabajos. De la misma forma, más estudios son necesarios para profundizar en la puesta en obra de las prácticas de pensamiento computacional en el ejercicio docente cotidiano de los grupos participantes, luego de su participación en un trayecto formativo. Asimismo, la construcción de TIS por parte de otros y otras docentes y en otros contextos resulta necesaria, no solo para consolidar la propuesta didáctica en cuanto a enseñanza de la programación desde un paradigma alternativo, sino también para aportar evidencias que permitan visibilizar las diferentes realidades y situaciones cotidianas que atraviesan día tras día a los maestros y a las maestras en su compleja labor.

Referencias

- Apiola, M. y Tedre, M. (2012). New perspectives on the pedagogy of programming in a developing country context. *Computer Science Education*, 22(3), 285-313. <https://doi.org/10.1080/08993408.2012.726871>
- Arranz, H. y Fuente, D. (2017). Evaluación del pensamiento computacional en educación. *Revista Interuniversitaria de Investigación en Tecnología Educativa (RIITE)*, 3, 25-39. Recuperado de <https://revistas.um.es/riite/article/view/267411/223391>
- Balanskat, A. y Engelhardt, K. (2015). *Computing our future. Computer programming and coding. Priorities, school curricula and initiatives across Europe*. Brussels, Belgium: European Schoolnet. Recuperado de http://www.eun.org/documents/411753/817341/Computing+our+future_final_2015.pdf/d3780a64-1081-4488-8549-6033200e3c03
- Barr, V. y Stephenson, C. (2011). Bringing computational thinking to K-12: What is involved and what is the role of the computer science education community? *ACM Inroads*, 2(1), 48-54. <https://doi.org/10.1145/1929887.1929905>

doi: <http://dx.doi.org/10.15359/ree.23-3.9>

URL: <http://www.una.ac.cr/educare>

CORREO: educare@una.cr

- Brackmann, C., Barone, D., Casali, A., Boucinha, R. M. y Muñoz-Hernández, S. (2016). Computational thinking: Panorama of the Americas. En *International Symposium on Computers in Education, SIIE 2016: Learning Analytics Technologies* (pp. 1-6). Salamanca, España: IEEE. <https://doi.org/10.1109/SIIE.2016.7751839>
- Brennan, K. y Resnick, M. (2012). New frameworks for studying and assessing the development of computational thinking. Ponencia presentada en *Annual American Educational Research Association meeting* (pp. 1-25). Vancouver, Canadá: AERA. Recuperado de https://web.media.mit.edu/~kbrennan/files/Brennan_Resnick_AERA2012_CT.pdf
- Falloon, G. (2015). Building computational thinking through programming in K-6 education: A New Zealand experience. En *Proceedings of EDULEARN15 Conference* (pp. 0882-0892). Barcelona: IATED. Recuperado de <https://researchcommons.waikato.ac.nz/bitstream/handle/10289/9455/Falloon%20full%20paper%20EDULEARN%2715.pdf?sequence=4&isAllowed=y>
- Gee, J. P. y Green, J. L. (1998). Discourse analysis, learning and social practice: A methodological study. *Review of Research in Education*, 23, 119-169. doi: <https://doi.org/10.2307/1167289>
- Grover, S. y Pea, R. (2013). Computational Thinking in K-12: A Review of the State of the Field. *Educational Researcher*, 42(1), 38-43. doi: <https://doi.org/10.3102/0013189X12463051>
- Heintz, F., Mannila, L. y Färnqvist, T. (2016). A review of models for introducing computational thinking, computer science and computing in K-12 education. En *Proceedings - Frontiers in Education Conference, FIE*. Pennsylvania, USA: IEEE. doi: <https://doi.org/10.1109/FIE.2016.7757410>
- Kafai, Y. B., Searle, K., Kaplan, E., Fields, D. A., Lee, E. y Lui, D. (2013). Cupcake cushions, scooby doo shirts, and soft boomboxes. En *Proceeding of the 44th ACM technical symposium on Computer science education - SIGCSE '13* (pp. 311-316). Denver, Colorado, USA: ACM. <https://doi.org/10.1145/2445196.2445291>
- Lave, J. y Wenger, E. (1991). *Situated learning: Legitimate peripheral participation*. Cambridge, UK: Cambridge University Press. doi: <https://doi.org/10.1017/CBO9780511815355>
- Lye, S. Y. y Koh, J. H. (2014). Review on teaching and learning of computational thinking through programming: What is next for K-12? *Computers in Human Behavior*, 41, 51-61. <https://doi.org/10.1016/j.chb.2014.09.012>
- Maloney, J., Resnick, M., Rusk, N., Silverman, B. y Eastmond, E. (2010). The scratch programming language and environment. *ACM Transactions on Computing Education*, 10(4), 1-15. <https://doi.org/10.1145/1868358.1868363>



doi: <http://dx.doi.org/10.15359/ree.23-3.9>

URL: <http://www.una.ac.cr/educare>

CORREO: educare@una.cr

- Martínez, M. C. y Echeveste, M. E. (2014). El rol de las comunidades de aprendizaje en la construcción de una visión común para la enseñanza de computación en las escuelas. *Revista Iberoamericana de Educación*, 65, 19-36. Recuperado de <https://rieoei.org/RIE/article/view/391>
- Menekse, M. (2015). Computer science teacher professional development in the United States: A review of studies published between 2004 and 2014. *Computer Science Education*, 25(4), 325-350. <https://doi.org/10.1080/08993408.2015.1111645>
- Monjelat, N. (2017). Programming technologies for social inclusion. En A. Diaz, A. Casali, M. C. Rivas, y A. S. Sprock (Eds.), *Twelfth Latin American Conference on Learning Technologies* (pp. 1-8). La Plata, Buenos Aires: IEEE. doi: [10.1109/LACLO.2017.8120901](https://doi.org/10.1109/LACLO.2017.8120901)
- Monjelat, N. y San Martín, P. S. (2016). Programar con Scratch en contextos educativos: ¿Asimilar directrices o co-construir tecnologías para la inclusión social? *Praxis Educativa*, 20(1), 61-71. doi: <https://doi.org/10.19137/praxiseducativa-2016-200106>
- Ohashi, Y. (2017). Preparedness of Japan's Elementary School Teachers for the Introduction of Computer Programming Education. En V. Dagiene y A. Hellas (Eds.), *Informatics in Schools: Focus on Learning Programming. ISSEP 2017. Lecture Notes in Computer Science* (Vol 10696, pp. 129-140). Helsinki, Finland: Springer International Publishing. doi: <https://doi.org/10.1007/978-3-319-71483-7>
- Palma, C. A. y Sarmiento, R. E. (2015). Estado del arte sobre experiencias de enseñanza de programación a niños y jóvenes para el mejoramiento de las competencias matemáticas en primaria. *Revista Mexicana de Investigación Educativa*, 20(65), 607-641. Recuperado de <https://www.redalyc.org/pdf/140/14035408013.pdf>
- Resnick, M. y Brennan, K. (2011). *Computational Thinking Practices (ScratchEd Webinar Series)*. Recuperado de <http://scratched.gse.harvard.edu/sites/default/files/april-2011-ct-practices.pdf>
- Rich, P. J., Jones, B., Belikov, O., Yoshikawa, E. y Perkins, M. (2017). Computing and engineering in elementary school: The effect of year-long training on elementary teacher self-efficacy and beliefs about teaching computing and engineering. *International Journal of Computer Science Education in Schools*, 1(1), 1-20. doi: <https://doi.org/10.21585/ijcses.v1i1.6>
- Sáez-López, J.-M., Román-González, M. y Vázquez-Cano, E. (2016). Visual programming languages integrated across the curriculum in elementary school: A two year case study using "Scratch" in five schools. *Computers & Education*, 97, 129-141. doi: <https://doi.org/10.1016/j.compedu.2016.03.003>
- Stake, R. E. (1995). *The art of case study research*. London: Sage.

doi: <http://dx.doi.org/10.15359/ree.23-3.9>

URL: <http://www.una.ac.cr/educare>

CORREO: educare@una.cr

- Thomas, H. (2012). Tecnologías para la inclusión social en América Latina: De las tecnologías apropiadas a los sistemas tecnológicos sociales. Problemas conceptuales y soluciones estratégicas. En H. Thomas, M. Fressoli y G. Santos (Eds.), *Tecnología, desarrollo y democracia: Nueve estudios sobre dinámicas sociotécnicas de exclusión/inclusión social* (pp. 25-76). Buenos Aires: Ministerio de Ciencia, Tecnología e Innovación Productiva de la Nación.
- Vygotsky, L. S. (1978). *Mind in society: The development of higher psychological processes*. Cambridge, MA: Harvard University Press.
- Wing, J. M. (2011). Research Notebook: Computational thinking--what and Why? *The link. The magazine of the Carnegie Mellon University School of Computer Science*. Recuperado de <https://www.cs.cmu.edu/link/research-notebook-computational-thinking-what-and-why>
- Yadav, A. (2017). Computer science teacher professional development: Towards a research agenda on teacher thinking and learning. En *WiPSCE '17* (pp. 1-2). Nijmegen, Netherlands: ACM Press. doi: <https://doi.org/10.1145/3137065.3137066>
- Yadav, A., Hong, H. y Stephenson, C. (2016). Computational thinking for All: Pedagogical approaches to embedding 21st century problem solving in K-12 classrooms. *TechTrends*, 60(6), 565-568. <https://doi.org/10.1007/s11528-016-0087-7>
- Yadav, A., Gretter, S., Good, J. y McLean, T. (2017). Computational thinking in teacher education. En P. J. Rich y C. B. Hodges (Eds.), *Emerging research, practice, and policy on computational Thinking* (pp. 205-220). Cham, Switzerland: Springer International. doi: <https://doi.org/10.1007/978-3-319-52691-1>
- Yin, R. K. (2003). *Case study research: Design and methods*. London: Sage.

