

Experiencias sobre la inducción de tecnologías programables para el desarrollo del pensamiento computacional en escuelas de zonas rurales y urbano marginales

Experiences on the induction of programmable technologies for the development of computational thinking in schools in rural and marginal urban areas

*V. DUARTE-MARTÍNEZ, GLADYS E. CARRILLO, ÁNGELA CARRERA, CRUZ MARÍA FALCONES

Escuela Superior Politécnica del Litoral, Campus Gustavo Galindo, Guayaquil, Ecuador

* vealduar@espol.edu.ec

RESUMEN

El pensamiento computacional es una habilidad que permite solucionar problemas de manera ordenada y sistemática mediante la aplicación de metodologías originadas en la programación de sistemas informáticos, tales como la división de tareas y el desarrollo de algoritmos, entre otras. Para el desarrollo de esta habilidad, varios estudios recomiendan que se deben utilizar mecanismos lúdicos durante la edad escolar. Este documento está enfocado en relatar las etapas implementadas en un proyecto exploratorio, cuyo objetivo es desarrollar el pensamiento computacional, a través del uso de tecnologías programables, en niños de edad escolar pertenecientes a zonas rurales y urbano-marginales. Con la implantación de este proyecto se logró capacitar a un total de 118 niños y 8 docentes en el uso de Scratch y LEGO Mindstorms, logrando así la mejora de sus capacidades de razonamiento lógico y computacional.

Palabras clave: Educación; Pensamiento computacional; Tecnologías programables; Scratch.

ABSTRACT

Computational thinking is a skill that allows solving problems in an orderly and systematic way, through the application of methodologies originated in the programming of computer systems, such as the division of tasks and the development of algorithms, among others. For the development of this ability, several studies recommend that playful mechanisms should be used during school age. This document

Recibido 29-06-2020
Aceptado 13-08-2020

Experiencias sobre la inducción de tecnologías programables para el desarrollo del pensamiento computacional en escuelas de zonas rurales y urbano marginales. V. Duarte-Martínez, Gladys E. Carrillo, Ángela Carrera, Cruz María Falcones • VÍNCULOS-ESPE (2020) VOL.5, No.3: 67 - 81

is focused on reviewing the stages implemented in an exploratory project, whose objective was to develop computational thinking, through the use of programmable technologies. The target group were school-age children belonging to rural and marginal urban areas. With the implementation of this project, a total of 118 children and 8 teachers were trained in the use of Scratch and LEGO Mindstorms, thus achieving the improvement of their logical and computational reasoning abilities.

Keywords: Education; Computational Thinking; Programmable technologies; Scratch.

1. INTRODUCCIÓN

El pensamiento computacional en el ámbito educativo se ha ido consolidando como una forma de desarrollar la habilidad de resolución de problemas, utilizando técnicas derivadas de la programación de sistemas informáticos, que son extrapoladas para aplicarse en el desarrollo de soluciones a problemas en cualquier ámbito (Zapata-Ros, 2015).

Varias iniciativas han utilizado algunas herramientas de programación lúdicas para desarrollar el pensamiento computacional en niños y adolescentes. Una de las herramientas más populares es Scratch¹, este es un entorno de programación desarrollado por un grupo de investigadores del Instituto de Tecnología de Massachusetts (MIT) que presenta la programación de manera atractiva y accesible para todo aquel que se enfrente por primera vez a programar (Resnick et al., 2009). Esta herramienta incluye el uso de bloques de código

predefinidos que los usuarios puedan arrastrar hasta su área de trabajo para determinar las acciones que realizara un personaje (Capot & Espinoza, 2015).

La programación tangible es otra herramienta usada para el desarrollo del pensamiento computacional y se la relaciona a la robótica. Siendo esta una manera de atraer la atención de los niños que aprenden armando piezas, a la vez que es posible que programen la autonomía de un robot (Bers, 2008). Este tipo de actividades son recomendadas para introducir la programación a niños de edad temprana. Un ejemplo de ello es la plataforma LEGO Mindstorm®² que permite a los niños interactuar con piezas, mientras construyen una máquina a la cual se le programa las actividades a realizar. De esta manera, los niños se divierten, mientras aprenden y aplican prácticas de abstracción y descomposición (At-matzidou & Demetriadis, 2016).

En diversos países europeos (Inglaterra, Francia, Finlandia, Italia, Portugal) se ha incluido a la programación y pensamiento computacional y algorítmico en sus programas de educación primaria y secundaria, ya sea como un curso requerido o una actividad extra curricular. Incluso países como Portugal incluyen al pensamiento computacional como uno de sus resultados de aprendizaje para sus estudiantes de secundaria (Bocconi et al., 2016). Cabe recalcar que estas iniciativas son en gran parte posibles debido al alto nivel de acceso a computadores y a conexión a internet de los países europeos.

En Latinoamérica, los desafíos son mayores, solo el 45.6 % de hogares posee una computadora y el 67% tiene acceso a internet (Rojas & Poveda, 2017). Sin embargo, no deja de ser un tema cada vez más importante en el desarrollo educacional. Tal es el caso de países como Chile, Colombia y Argentina, que dedican sus esfuerzos a la implementación de tecnologías que fortalezcan la educación y el desarrollo del pensamiento computacional (Rico & Olabe, 2018; Sarmiento Bolívar, 2019; Vilanova, 2017).

¹ <https://scratch.mit.edu/>

² <https://www.lego.com/es-ar/themes/mindstorms>

Tomando en cuenta esta realidad, la *Escuela Superior Politécnica del Litoral (ESPOL)* de la ciudad de Guayaquil, en Ecuador, mediante su *Unidad de Vinculación con la Sociedad* busca ayudar a solucionar problemas y necesidades de sectores poblacionales de atención prioritaria a través del desarrollo de proyectos donde intervienen docentes y estudiantes que aplican los conocimientos de sus respectivas carreras en un ámbito social.

Los sectores poblacionales de interés, en su mayoría, se han desarrollado a partir de asentamientos humanos sin planificación gubernamental, pudiendo carecer en gran medida de servicios básicos. Se presentan, por tanto factores de riesgo como la pobreza, el estigma social, la violencia naturalizada, deficiencia de cuidados familiares, el abandono y el acceso limitado a programas y servicios, lo que puede afectar la supervivencia y desarrollo de los niños del sector (Unicef et al., 2013).

En el presente trabajo se expone la metodología aplicada por un proyecto de servicio comunitario, relacionado al uso de herramientas lúdicas para el desarrollo del pensamiento computacional. Dicho proyecto ha sido desarrollado a la par en dos zonas distintas. Por un lado, se ha implementado con participantes de una comunidad rural del cantón Durán, y a la vez se ha desarrollado con estudiantes de una escuela perteneciente a una zona urbano marginal de Guayaquil.

Descripción de las comunidades participantes

Ambas comunidades están ubicadas en la provincia del Guayas pero poseen características muy dispares. Por un lado, el recinto La Unión ubicado en el cantón Durán, en una zona rural de difícil acceso y limitadas vías que dificultan la movilización; sus

pobladores evidencian un precario nivel de educación que deriva en pobreza, tal como lo evidencia un estudio socioeconómico realizado en el 2017 por la Facultad de Ciencias Sociales y Humanísticas de la ESPOL (FCSH) y PRODUMAR (empresa de procesamiento y venta de productos hidrobiológicos ubicada en las cercanías 2017); su única escuela no cuenta con un laboratorio de computación que permita a los niños recibir formación en informática. Por otro lado, la Unidad Educativa *Sagrada Familia de Nazareth* se ubicada en Nueva Prosperina, un asentamiento del sector noroeste del cantón Guayaquil; posee un laboratorio de computación con quince computadoras de características básicas que están conectadas mediante una red de área local, con una conexión a Internet de 1Mbps que impide el acceso simultáneo a sitios web externos ricos en contenido multimedia, y su capacidad es de treinta estudiantes que reciben formación básica en el uso del computador y herramientas ofimáticas.

El proyecto ejecutado en las mencionadas comunidades de tipo rural y urbano marginal contribuyen al desarrollo del pensamiento computacional de los niños mediante la enseñanza de conceptos básicos de programación con herramientas lúdicas (Scratch y Lego Mind-Storm), permitiendo incrementar sus habilidades de resolución de problemas en cualquier ámbito, alineándose de manera directa con el objetivo 4 de los Objetivos de Desarrollo Sostenible (2017-2030) (United Nations, 2016), que busca garantizar una educación inclusiva, equitativa y de calidad y promover oportunidades de aprendizaje durante toda la vida para todos.

2. METODOLOGÍA

Se ha seguido una línea base para el desarrollo de cada una de las etapas del proyecto, como se muestra en la Figura 1. Sin embargo, debido a las limitaciones de las zonas donde se ha implantado, el proyecto se ha adaptado a la realidad de los participantes.

Experiencias sobre la inducción de tecnologías programables para el desarrollo del pensamiento computacional en escuelas de zonas rurales y urbano marginales. V. Duarte-Martínez, Gladys E. Carrillo, Ángela Carrera, Cruz María Falcones • VÍNCULOS-ESPE (2020) VOL.5, No.3: 67 - 81

2.1 Formulación de los proyectos

El proyecto plantea sesiones de clases para fomentar el pensamiento computacional y están dirigidas a estudiantes de primaria, de cuarto a sexto año de educación básica, cuyas edades están comprendidas entre ocho y diez años, por lo cual, las herramientas seleccionadas para la enseñanza fueron: Scratch y Lego MindStorm.

Por un lado, en la zona urbano marginal, se incluyeron capacitaciones adicionales a ocho docentes. Esto debido a que la institución tiene un laboratorio de computación que aunque tiene máquinas con características básicas, garantiza de alguna manera que el

proyecto podrá ser replicado en otros cursos, con diferentes grupos de niños en el futuro.

Por otro lado, en la zona rural, la falta de infraestructura impidió que las capacitaciones puedan realizarse directamente en la escuela. Por lo que no fue posible incluir a los docentes en las capacitaciones que se realizaron fuera de la comunidad. Esto evidentemente limita el impacto y continuidad que puede tener el proyecto, después de la intervención universitaria. La Tabla 1 muestra la distribución de participantes.

Figura 1: Etapas de la metodología

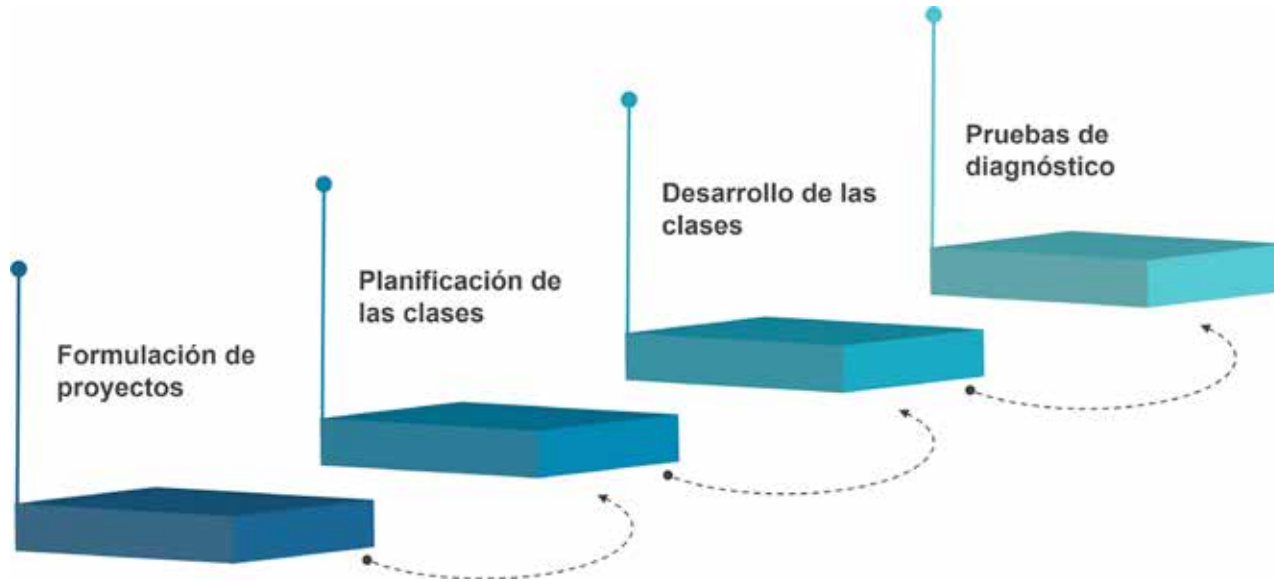


Tabla 1: Distribución de participantes

Grupo	Rural	Urbano marginal
4to. Año de Básica	12	26
5to Año de Básica	8	36
6to. Año de Básica	0	36
Docentes	0	8

Adicionalmente, la intervención de la ESPOL estuvo integrada por docentes y estudiantes de las carreras de Ingeniería en Computación, Ingeniería en Telemática y Licenciatura en Producción Audiovisual. Los estudiantes se encargaron de preparar el material didáctico, imprimible y la planificación de las clases, mientras los profesores se encargaron de tutoriar sus actividades.

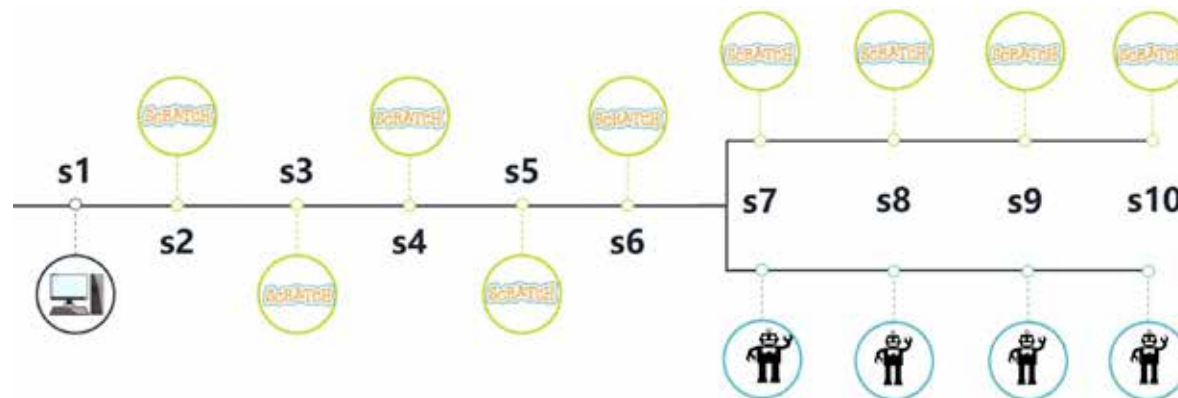
2.2 Planificación y desarrollo de las clases

En base en la descripción de la realidad de los participantes, se decidió que los niños pertenecientes a la zona rural asistieran a

ESPOL, para hacer uso del laboratorio de computación equipado. Estas clases se planificaron los sábados como actividades extracurriculares. En cambio, los niños de la zona urbano marginal, podían recibir las clases en la escuela y durante su horario habitual.

En la zona urbano marginal los niños pudieron empezar directamente con las clases de Scratch. Mientras que, en la zona rural, fue imprescindible incluir una clase introductoria sobre el uso del computador, para tratar temas claves como conceptos de hardware, software y uso del sistema operativo Windows.

Se prepararon nueve sesiones del lenguaje de programación Scratch y mientras, un grupo sólo recibió clases de Scratch durante todo el periodo de duración del proyecto, el otro grupo recibió cuatro clases de robótica al final usando la plataforma Lego Mindstorms. Para seleccionar a los niños que iban a recibir las clases de robótica, se seleccionaron aquellos con mejor rendimiento en las actividades realizadas en clases. La Figura 2, muestra la distribución de las clases.

Figura 2: Distribución de las clases

Experiencias sobre la inducción de tecnologías programables para el desarrollo del pensamiento computacional en escuelas de zonas rurales y urbano marginales. V. Duarte-Martínez, Gladys E. Carrillo, Ángela Carrera, Cruz María Falcones • VÍNCULOS-ESPE (2020) VOL.5, No.3: 67 - 81

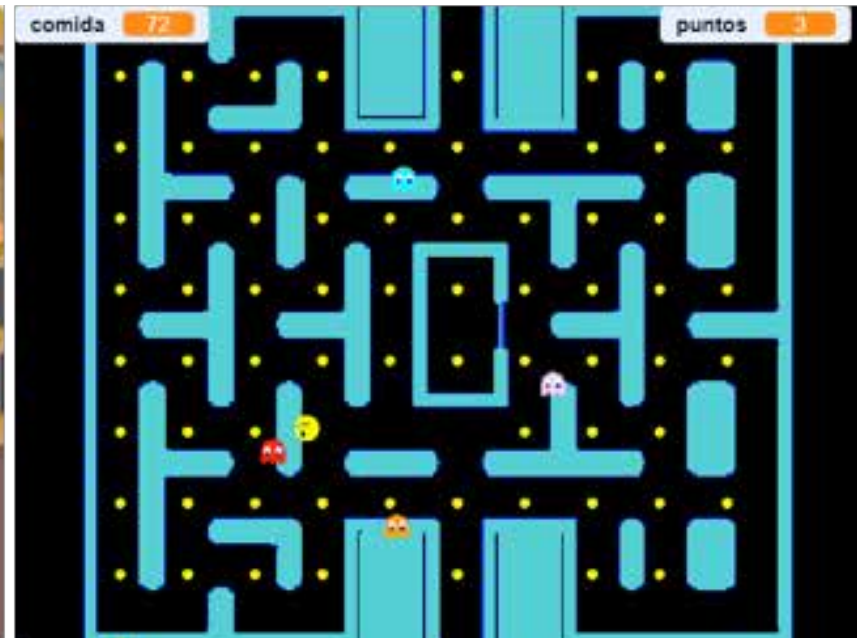
Cada sesión se compuso de tres momentos: la introducción, donde se realizaban actividades romper el hielo y se recordaba la clase anterior; el desarrollo, donde los tutores (estudiantes de ESPOL) explicaban el tema central de la sesión y se realizaban dos talleres, un taller guiado con instrucciones y un taller autónomo. Con cada taller y recurso utilizado en clases, se buscaba obtener un beneficio afectivo, es decir, que las actividades fueran divertidas y llamativas

para los niños y que a su vez colaboren al desarrollo cognitivo del pensamiento computacional (Wilson & Moffat, 2010). Algunos de los recursos han sido ampliamente usados y corresponden a tarjetas de código brindadas por Scratch en múltiples idiomas y otros fueron originales o adaptaciones de juegos populares locales para aumentar la familiaridad con los niños, como se muestra en la Figura 3a y la Figura 3b.

Figura 3: Ejemplos de adaptaciones de juegos desarrollados en Scratch.



(a) Juego del gato y el ratón.



(b) Juego Pacman.

Figura 4: Muestras de las clases realizadas en los proyectos.



(a) Interacción con los niños antes de clases.



(b) Clases de Scratch



(c) Niños reunidos el primer día de clases.



(d) Niños revisando el robot armado con LEGO Mindstorms.

Finalmente, al cierre de la sesión, los tutores remarcaban los puntos más importantes de la clase y revisaban los talleres autónomos de cada estudiante. Cabe recalcar que no se llevaba un control cuantitativo en los talleres, el objetivo era motivar a cada estudiante a través de una escala cualitativa positiva, por lo que un trabajo podría evaluarse como excelente o muy bueno. Cada niño presentaba su cartilla de talleres realizados y el tutor marcaba con una pegatina que representaba

la evaluación de dicho taller. Este recurso motivaba a los niños a concentrarse para presentar su trabajo pronto y obtener la pegatina correspondiente. La Tabla 2, muestra la descripción de temas abarcados en cada sesión de Scratch y su respectivo proceso de pensamiento computacional asociado, mientras que la Tabla 3, muestra los temas abordados en las sesiones de robótica. Por otro lado, las Figuras 4a, 4b, 4c y 4d muestran la interacción con los niños en las clases.

Experiencias sobre la inducción de tecnologías programables para el desarrollo del pensamiento computacional en escuelas de zonas rurales y urbano marginales. V. Duarte-Martínez, Gladys E. Carrillo, Ángela Carrera, Cruz María Falcones • VÍNCULOS-ESPE (2020) VOL.5, No.3: 67 - 81

Tabla 2: Temas abordados sobre Scratch

Sesión	Tema-Descripción	Proceso Pensamiento Computacional
1	Introducción al entorno de trabajo <ul style="list-style-type: none"> Presentar el concepto de programación y ambiente de trabajo 	
2	Uso de Variables <ul style="list-style-type: none"> Utilizar variables para modelar atributos de objetos de la vida real Utilizar variables globales para contar elementos 	Abstracción
3	Algoritmos <ul style="list-style-type: none"> Introducir el concepto de algoritmos Crear historias en Scratch utilizando algoritmos y bloques de movimiento 	Generalización Algorítmico - Secuenciales
4	Algoritmos con decisiones <ul style="list-style-type: none"> Introducir el concepto de condicionales utilizando situaciones de la vida real Utilizar sensores y condiciones para crear pequeños juegos 	Generalización Algorítmico – Flujos de Control
5	Estructuras de Control Iterativas <ul style="list-style-type: none"> Introducir el concepto de estructuras iterativas utilizando situaciones de la vida real. Demostrar el uso de estructuras iterativas para la repetición de acciones como mover personajes 	Generalización Algorítmico – Flujos de Control
6	Eventos y Animaciones <ul style="list-style-type: none"> Introducir el concepto de evento, para permitir la interacción de un usuario en juegos Introducir disfraces 	Generalización
7	Paralelismo y Aleatoriedad <ul style="list-style-type: none"> Introducir el concepto de paralelismo utilizando situaciones de la vida real Crear historias o pequeños juegos con múltiples personajes Explicar la aleatoriedad e incluirla como elementos de un juego 	Descomposición
8-9	Proyecto Final <ul style="list-style-type: none"> Pequeño juego que integre todos los temas dados. 	Generalización Algoritmos Descomposición Depuración

2.3 Pruebas de diagnóstico

Durante el transcurso de este curso, se realizaron dos pruebas de diagnóstico, una al inicio para evaluar las habilidades previas de los niños, y una final para poder comparar la efectividad del curso.

Medir el pensamiento computacional puede ser desafiante y dependiente a variables como la edad, el aspecto a medir y el escenario. A menudo, los trabajos existentes se enfocan en la capacidad de los estudiantes en crear artefactos usando habilidades del pensamiento

computacional en una variedad de escenarios y plataformas. Por ejemplo, Ba- sawapatna, Bennett y Repenning desarrollaron una evaluación de pensamiento computacional automática que buscaba medir la lógica de los estudiantes para replicar juegos o simulaciones. Werner, Denner, Campe, y Kawamoto (2012) evaluaron estudiantes de diez a catorce años en la resolución de tres desafíos usando el ambiente Alice, una herramienta de programación en bloques proporcionada por Oracle.

Tabla 3: Temas abordados sobre LEGO Mindstorms

Sesión	Tema-Descripción
1	Introducción a la robótica
	• Piezas del LEGO
	• Relacionar scratch (programación con bloques)
	• Construcción del robot
	• Introducción al software EV3
2	Uso de Variables
	• Pequeños ejercicios
	• Sensor de seguridad
3	Uso de Variables
	• Sensor de movimiento
	Construcción de un dispositivo de seguridad
4	• Manejo correcto de las piezas
	• Razonamiento espacial
	• Prácticas de iteración y soluciones de diseño
4	Construcción de robot Bazooka
	• Seguimiento de instrucciones para construir prototipo de proyectil

Experiencias sobre la inducción de tecnologías programables para el desarrollo del pensamiento computacional en escuelas de zonas rurales y urbano marginales. V. Duarte-Martínez, Gladys E. Carrillo, Ángela Carrera, Cruz María Falcones • VÍNCULOS-ESPE (2020) VOL.5, No.3: 67 - 81

Sin embargo, en este proyecto uno de los desafíos es la falta de experiencia de los participantes en cualquier plataforma de programación, más aun, un grupo de niños no tenían experiencia en el uso de un computador. Por lo cual, se utilizaron pruebas desconectadas, es decir, pruebas impresas en hojas de papel, para medir tres aspectos básicos del pensamiento computacional: pensamiento algorítmico secuencial, flujos de control, y depuración (Selby, 2014).

Este enfoque de actividades desconectadas no es nuevo y ha sido usado previamente (Chen et al., 2017; Román-González, Pérez-González, & Jiménez-Fernández, 2015), evidenciando que el

pensamiento computacional es una habilidad cognitiva que puede ser fomentado a través de actividades que van más allá del uso del computador.

El instrumento evaluativo se diseñó con preguntas de opción múltiple y desarrollo, que reflejan escenarios didácticos. Las opciones presentaban bloques de instrucciones en pseudocódigo, en los cuales los estudiantes tenían que seleccionar la respuesta correcta para cumplir un objetivo u ordenar las instrucciones en el flujo adecuado. Un ejemplo de pregunta que se utilizó para medir la capacidad algorítmica secuencial y depuración se muestra en la Figura 5.

Figura 5: Ejemplo de pregunta para medir la capacidad algorítmica secuencial y depuración.

El pollito Alex, busca a su mamá.

El pollito Alex se perdió y necesita regresar con su mamá la Gallina Gloria siguiendo el camino café. Alex tiene pensado en una ruta, pero no está muy seguro de la misma, la ruta que tomara se encuentra en el **cuadro amarillo**. Si encuentras un error en la ruta, escríbelo en el cuadro verde:

Ruta del pollito Alex	Ruta Correcta
Repetir 7 veces Avanzar un cuadro	Repetir ___ veces _____
Girar a la izquierda	_____
Repetir 2 veces Avanzar un cuadro	Repetir ___ veces _____
Girar a la derecha	_____
Repetir 4 veces Girar a la izquierda	Repetir ___ veces _____

De esta manera, fue posible determinar la evolución durante el proceso de aprendizaje de las diferentes temáticas que fueron abordadas en los talleres. Las calificaciones de las evaluaciones solo fueron de uso interno y no se revelaron a los estudiantes.

2.4 Capacitación a docentes

Esta etapa solo se realizó en la zona urbano marginal, ya que el proyecto en esta zona se llevó a cabo en la escuela, donde sí contaban con un laboratorio de computación, por lo cual, fue posible involucrar a los profesores en el desarrollo de las actividades in situ.

Buscando que los docentes de la escuela estén en la capacidad de continuar con las capacitaciones de Scratch, se realizaron cuatro talleres fuera del horario de trabajo de los docentes. Cada sesión con una duración de dos horas.

En la primera sesión se realizó una introducción sobre el pensamiento computacional para socializar los beneficios y la importancia del desarrollo de esta habilidad desde edades tempranas. Los temas básicos de Scratch que se cubrieron se muestran en la Tabla 4.

2.5 Desarrollo de sistema computacional

Esta etapa se realizó también solo para la zona urbano marginal como una manera de potenciar el uso del laboratorio de computación de la escuela, y para continuar las actividades de capacitación de Scratch luego de la finalización del proyecto.

Para que las guías didácticas y los video tutoriales puedan estar

Tabla 4: *Temas de la capacitación a docentes*

Sesión	Tema-Descripción
1	La interfaz de Scratch, atributos, posición y dirección de un personaje
2	Instrucciones de movimiento, apariencia y control (repetición y condicionales)
3	Instrucciones de control, eventos, sonidos y disfraces de los personajes.
4	Uso de mensajes entre personajes y ejercicio integrador.

disponibles para un uso posterior en la escuela y para otros establecimientos educativos, se desarrolló una aplicación web, usando el framework Laravel (PHP)³, donde se puede acceder a los materiales divididos por las herramientas utilizadas: Scratch y LEGO Mindstorms.

La aplicación fue instalada y configurada en un dispositivo de almacenamiento USB con un sistema operativo Linux con arranque automático (Live USB) para permitir que la aplicación web pueda ser desplegada sin necesidad de ejecutar comandos adicionales. Este Live USB puede ser utilizado para proveer un entorno de enseñanza-aprendizaje que permita a los alumnos acceder a ejercicios de programación que permitan desarrollar la habilidad de resolución de problemas, incluso sin la asistencia directa de un profesor. Esta aplicación no necesitara de un servidor exclusivo, por lo que podrá también utilizarse en cualquier computador con solo conectar el dispositivo USB.

³ <https://laravel.com/>

3. RESULTADOS

Los resultados del proyecto fueron satisfactorios para ambas partes, los niños beneficiarios terminaron muy contentos y los padres de familia agradecidos por las enseñanzas recibidas. Por otro lado, los estudiantes de la ESPOLE enriquecieron sus conocimientos y terminaron también satisfechos con la labor realizada al poder colaborar con la comunidad.

Figura 5: Ejemplo de pregunta para medir la capacidad algorítmica secuencial y depuración.

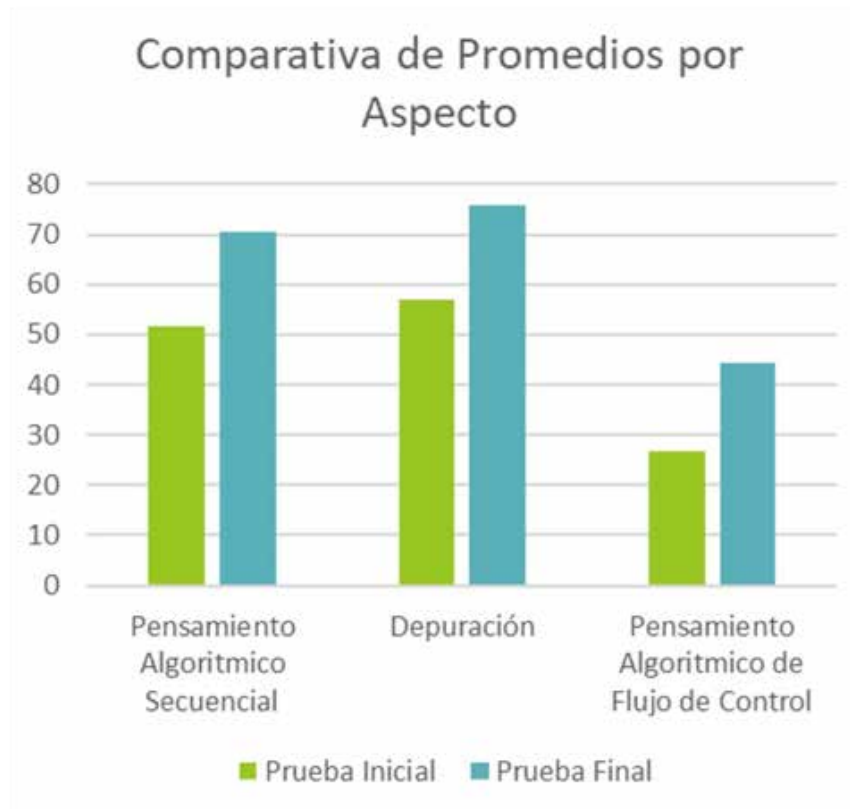


Tabla 5: Promedios de los resultados obtenidos en las evaluaciones

Aspecto	Prueba Inicial	Prueba Final
Pensamiento algorítmico secuencial	51.67	70.38
Depuración	56.9	75.95
Pensamiento algorítmico de flujo de control	26.81	44.38

Los resultados se han medido comparando las calificaciones obtenidas en la evaluación que se tomó el primer día de clases versus las calificaciones de la evaluación tomada al final del curso.

Con base en estos datos, es posible evidenciar la mejora contundente de los estudiantes comparando el promedio de las pruebas inicial y final por aspectos. Se ha obtenido una mejora significativa en cada uno de los aspectos evaluados. En la Tabla 5 y en la Figura 6 se muestran los promedios de calificaciones obtenidas en las evaluaciones calificadas sobre cien puntos.

Los ocho docentes que participaron en las capacitaciones recibieron las clases en el laboratorio de la escuela, siguiendo el programa de clases definidos en la Tabla 4. En cada sesión, se desarrollaron ejercicios relacionados a los temas cubiertos en cada clase y en la última sesión, se planteó un ejercicio con mayor complejidad, el cual pudo ser desarrollado por los profesores sin mayores inconvenientes. Con las clases recibidas, los docentes quedaron en la capacidad de poder replicar las capacitaciones de Scratch con sus futuros alumnos.

Respecto al dispositivo USB configurado, este fue probado en el laboratorio de computación de la escuela donde se realizaron las capacitaciones a los niños, pero sólo para probar que la aplicación

web se puede acceder desde otro computador. El inicio del sistema operativo es un poco lento, tarda unos cuantos minutos en estar disponible, sin embargo, la aplicación web se levanta sin problemas y no es necesaria ninguna configuración adicional. Las guías de los ejercicios prácticos y los video tutoriales pueden ser visualizados normalmente.

La red del laboratorio de la escuela asigna a las computadoras direcciones con DHCP⁴ por lo que sólo fue necesario ejecutar el comando correspondiente (hostname) para conocer la dirección IP⁵. Aunque la asignación de direcciones por DHCP es popular, se pueden encontrar laboratorios con asignación manual, por lo tanto, junto al dispositivo USB se provee un pequeño documento con las instrucciones básicas para la configuración manual de la dirección de red. Sin embargo, se espera que en cada escuela exista una persona con conocimientos técnicos de computación para realizar esta tarea, ya que a pesar de ser una acción fácil de realizar siguiendo las instrucciones, genera un poco de rechazo en personas sin experiencia en el uso del sistema operativo Linux.

4. CONCLUSIONES Y RECOMENDACIONES

La alfabetización digital es fundamental en el desarrollo de las nuevas generaciones. En los proyectos expuestos en este documento, cuyo objetivo principal era desarrollar el pensamiento computacional en estudiantes de educación primaria, nos topamos con realidades que quizás muchos no habíamos palpado antes, niños

que realizan sus estudios en una escuela pública de bajos recursos, en situación precaria pero con muchas ganas de aprender. Los proyectos han sido realmente alentadores para quienes los conformamos y dejan gratas enseñanzas y mucha satisfacción por haber aportado al desarrollo de los niños participantes, así como las lecciones aprendidas sobre qué será necesario mejorar para futuras ocasiones.

Trabajar con niños de ocho a diez años de edad ha sido un reto para estudiantes y docentes que antes no habían tenido esta experiencia y nos deja claro que, en caso de querer replicarlo, sería de mucha utilidad poder capacitar a los estudiantes en algún taller de pedagogía infantil, previo al inicio de clases del curso, donde puedan recibir tips y consejos sobre cómo llevar una clase para niños.

Finalmente, otra limitante que enfrentamos fue la falta de equipos de robótica, por lo cual se sugiere invertir en la compra de equipos LEGO Mindstorms. De esta manera, más niños podrían tener acceso a estas prácticas. En los cursos desarrollados, solo se contaba con cuatro equipos de LEGO para cada sesión, así que se formaron grupos de tres estudiantes y solo doce niños por cada zona pudieron participar; sin embargo, nos dimos cuenta de que lo ideal sería que se deberían formar parejas de trabajo en lugar de equipos de tres niños, para que participen de forma activa durante toda la clase.

Con la implantación de este proyecto explorativo, se espera que los niños puedan utilizar las habilidades desarrolladas para mejorar su rendimiento académico. En futuros trabajos, pretendemos hacer una comparación de la evolución de los niños que se capacitaron versus aquellos que no fueron parte de este proyecto por distintos factores. De esta manera, se pretende comprobar el efecto positivo de exponer a los niños, desde edades tempranas, al uso de tecnologías como apoyo para el aprendizaje.

⁴ *Dynamic Host Configuration Protocol*

⁵ *Internet Protocol*

Experiencias sobre la inducción de tecnologías programables para el desarrollo del pensamiento computacional en escuelas de zonas rurales y urbano marginales. V. Duarte-Martínez, Gladys E. Carrillo, Ángela Carrera, Cruz María Falcones • VÍNCULOS-ESPE (2020) VOL.5, No.3: 67 - 81

REFERENCIAS

- Atmatzidou, S., & Demetriadis, S. (2016, jan). Advancing students' computational thinking skills through educational robotics: A study on age and gender relevant differences. *Robotics and Autonomous Systems*, 75, 661–670. doi: 10.1016/j.robot.2015.10.008
- Bers, M. U. (2008). *Teachers College Record: Blocks to Robots: Learning with Technology in the Early Childhood Classroom*. Teachers College Press.
- Bocconi, S., Chiocciariello, A., Dettori, G., Ferrari, A., Engelhardt, K., Kampylis, P., & Punie, Y. (2016). Developing computational thinking in compulsory education. European Commission, JRC Science for Policy Report.
- Capot, R. B., & Espinoza, R. M. (2015). Desarrollo del Pensamiento Computacional con Scratch (Tech. Rep.).
- Chen, G., Shen, J., Barth-Cohen, L., Jiang, S., Huang, X., & Eltoukhy, M. (2017). Assess- ing elementary students' computational thinking in everyday reasoning and robotics programming. *Computers & Education*, 109, 162–175.
- Koh, K. H., Basawapatna, A., Bennett, V., & Repenning, A. (2010). Towards the automatic recognition of computational thinking for adaptive visual language learning. In 2010 IEEE Symposium on Visual Languages and Human-Centric Computing (pp. 59–66).
- Martín, I., & Moreira, L. (2018). Productive inclusion of women to promote community strengthening of la union-duran in guayas-ecuador. In *Innovation in education and inclusion : Proceedings of the 16th laccei international multi-conference for engineering, education and technology*.
- Resnick, M., Maloney, J., Monroy-Hernández, A., Rusk, N., Eastmond, E., Brennan, K.,... others (2009). Scratch: programming for all. *Communications of the ACM*, 52(11), 60–67.
- Rico, M. J., & Olabe, X. B. (2018). Pensamiento computacional: rompiendo brechas digitales y educativas. *Edmetic*, 7(1), 26–42.
- Rojas, E. F., & Poveda, L. (2016). Estado de la banda ancha en america latina y el caribe 2016. Recuperado de <http://repositorio.minedu.gob.pe/handle/123456789/4805>.
- Román-González, M., Pérez-González, J. C., & Jiménez-Fernández, C. (2015). Test de pensamiento computacional: diseño y psicometría general. In *Iii congreso internacional sobre aprendizaje, innovación y competitividad (CINAIC 2015)*.
- Sarmiento Bolívar, M. I. (2019). Experiencias y estrategias educativas con TIC para el desarrollo del pensamiento computacional en Iberoamérica. *Pensamiento Actual*, 19, 12–27. doi: 10.15517/PA.V19I32.37792
- Selby, C. C. (2014). How can the teaching of programming be used to enhance Computational Thinking skills? (Doctoral dissertation). doi: 10.1016/j.jsv.2010.04.020
- Unicef, et al. (2013). El desarrollo del niño en la primera infancia y la discapacidad: Un documento de debate. Recuperado de [http://www.unicef.org/earlychildhood/files/ECDD_SPANISHFINAL \(low res\).pdf](http://www.unicef.org/earlychildhood/files/ECDD_SPANISHFINAL_low_res.pdf).
- United Nations. (2016). The sustainable development goals report 2016. United Nations. doi: 10.29171/azu acku pamphlet k3240 s878 2016
- Vilanova, G. E. (2017). Tecnología educativa para el desarrollo del pensamiento computacional. In *CISCI 2017 - decima sexta conferencia iberoamericana en sistemas, cibernética e informática, décimo cuarto simposium iberoamericano en educación, cibernética e informática, SIECI 2017 - memorias* (pp. 69–73).

Werner, L., Denner, J., Campe, S., & Kawamoto, D. C. (2012). The fairy performance assessment: measuring computational thinking in middle school. In Proceedings of the 43rd acm technical symposium on computer science education (pp. 215–220).

Wilson, A., & Moffat, D. C. (2010). Evaluating scratch to introduce younger schoolchildren to programming. In Ppig (p. 7).

Zapata-Ros, M. (2015). Pensamiento computacional: Una nueva alfabetización digital. Revista de Educación a Distancia(46).



VERÓNICA DUARTE, M.SC.

- Máster en Ciencia de Datos de la Universidad de Granada (España), es también ingeniera Ciencias Computacionales de la Escuela Superior Politécnica del Litoral (ESPOL). Actualmente, se encuentra realizando su trabajo doctoral en la Universidad de Granada. En su carrera profesional ha trabajado como parte del equipo de desarrollo de software para empresas privadas y también dentro del área de sistemas de ESPOL. Se desempeña como docente de la carrera de Ingeniería en Computación de la ESPOL desde el año 2017, donde también ejerce como directora y tutora de proyectos de vinculación con la sociedad para estudiantes de la misma carrera.



ANGELA CARRERA RIVERA

- Licenciada en Sistemas de Información en la Escuela Superior Politécnica del Litoral (Ecuador) y posee un título de Máster en Tecnologías de la Información de la Universidad de Melbourne (Australia). En su carrera profesional ha trabajado en varios proyectos de desarrollo de software en el sector bancario, industrial y educativo. Actualmente, se desempeña como Docente de la carrera de Ingeniería en Computación de la ESPOL, donde ejerce también como tutora de proyectos de vinculación con la sociedad para estudiantes de la misma carrera.



CRUZ MARÍA FALCONES, M.SC.

- Máster en Ciencia y Tecnología Informática de la Universidad Carlos III de Madrid (España), es también Magíster en Sistemas de Información Gerencial e Ingeniera en Computación de la Escuela Superior Politécnica del Litoral (Ecuador). Es profesora de la carrera de Ingeniería en Computación de la ESPOL desde el 2005, en las áreas de matemáticas, programación y sistemas de información; además, es coordinadora de Prácticas Comunitarias de Vinculación con la Sociedad desde el 2015. En su carrera profesional se ha desempeñado como Jefa de Sistemas, Analista y en el área de Soporte Técnico.



GLADYS E. CARRILLO

- Ingeniera en Computación y Magíster en Ciencias de la Computación graduada en la Escuela Superior Politécnica del Litoral (Ecuador), además, posee un título de Máster en Software Libre por la Universidad Abierta de Cataluña (España). En su carrera profesional ha trabajado en varios proyectos de desarrollo de software para diversos sectores económicos, tales como comercio, telecomunicaciones, industria, salud y educación. Actualmente, se desempeña como Docente de la carrera de Ingeniería en Computación de la ESPOL, donde también ejerce como directora y tutora de proyectos de vinculación con la sociedad para estudiantes de la misma carrera.