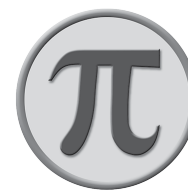

Artículo original

HACIA EL DISEÑO DE UN SISTEMA INTERACTIVO BASADO EN INTERACCIÓN TANGIBLE COMO APOYO AL DESARROLLO DE COMPETENCIAS DEL PENSAMIENTO COMPUTACIONAL PARA NIÑOS EN AULAS REGULARES ENTRE 9 A 10 AÑOS



TOWARDS THE DESIGN OF AN INTERACTIVE SYSTEM BASED ON TANGIBLE INTERACTION AS A SUPPORT TO THE DEVELOPMENT OF COMPUTATIONAL THINKING SKILLS FOR CHILDREN IN REGULAR CLASSROOMS BETWEEN 9 AND 10 YEARS

Sandoval, Diana Fernanda; Lara, María Juliana; Cano, Sandra

Diana Fernanda Sandoval dfsandoval09@gmail.com
Universidad San Buenaventura, Colombia
María Juliana Lara mariajuli12@hotmail.com
Universidad San Buenaventura, Colombia
Sandra Cano sandra.cano@gmail.com
Universidad San Buenaventura, Colombia

Publicaciones e Investigación
Universidad Nacional Abierta y a Distancia, Colombia
ISSN: 1900-6608
ISSN-e: 2539-4088
Periodicidad: Semestral
vol. 12, núm. 2, 2018
publicaciones.investigacion@unad.edu.co

Recepción: 02 Mayo 2018
Aprobación: 19 Mayo 2018

URL: <http://portal.amelica.org/ameli/jatsRepo/129/1292399001/index.html>

Resumen: El pensamiento computacional involucra la resolución de problemas en cualquier disciplina. Hoy en día, con el crecimiento de las tecnologías, ha surgido un gran interés por incluir el pensamiento computacional dentro del aula, usando otras maneras de interacción no tradicionales. Por lo que, la interacción tangible desde el área de la educación ha sido foco de estudio, ya que propone una interacción con elementos físicos y digitales. Siguiendo la filosofía de diseño centrado en el usuario, se propone el diseño de un sistema interactivo que pueda servir de apoyo para desarrollar habilidades del pensamiento computacional en niños en aulas regulares entre 9 y 10 años.

Palabras clave: niños en aulas regulares, pensamiento computacional, sistemas interactivos.

Abstract: Computational Thinking involves solving problems in any discipline. Nowadays, the growth of technologies, there has been a great interest to include computational thinking into the classroom, using other non-traditional ways of interaction. Therefore, the tangible interaction with physical and digital elements. Following the user-centered design philosophy, it is proposed a interactive system design that can support to develop skills related with the computational thinking in children in regular conditions between 9 to 10 years.

Keywords: Children in regular aula, Computational Thinking, Interactive Systems.

1. INTRODUCCIÓN

El pensamiento computacional (en inglés, Computational Thinking) es un término que se dio a conocer por Jeannette Wing en el 2006, el cual se define “como la resolución de problemas en todas las disciplinas, incluidas las humanidades, matemáticas y ciencias” (Wing 2006). Hoy en día, ha aumentado un gran interés por incluirse dentro del ámbito escolar, para la formulación y resolución de problemas con ayuda de herramientas tecnológicas, de tal manera que ayude a promover en los estudiantes pensamientos, como: algoritmo, lógico, abstracto y creativo. En el 2009 Jeannette Wing, relaciona el pensamiento computacional como la capacidad de análisis que tiene el niño, explicando el tema en una conferencia en Pensacola en el 2009 (Wing 2009):

“El Pensamiento Computacional (PC) será una habilidad fundamental utilizada por todos en el mundo. A la lectura, escritura y aritmética, vamos a añadir el PC a la capacidad de análisis de cada niño. El Pensamiento Computacional es un enfoque para la solución de problemas, construcción de sistemas, y la comprensión del comportamiento humano que se basa en el poder y los límites de la computación. Si bien, PC ya ha comenzado a influir en muchas disciplinas, desde las ciencias de las humanidades, lo mejor está aún por venir. De cara al futuro, podemos anticipar incluso efectos más profundos del pensamiento computacional en la ciencia, la tecnología y la sociedad: Entretanto, nuevos descubrimientos se realizará, habrá innovación y las culturas evolucionarán”.

Por lo que, Wing en su conferencia ha enfatizado que el pensamiento computacional, es pensar de manera recursiva, es reformular un problema aparentemente difícil sin saber cómo resolver usando diferentes métodos. Hoy en día, Scratch (Scratch), es usado como un entorno gráfico de programación en los colegios. Este entorno funciona mediante el aprendizaje de unión de bloques, y está orientado al aprendizaje de la programación en niños y adolescentes.

Por otro lado, desde el área de tecnología, un sistema interactivo puede definirse como la relación entre el usuario y el sistema (Cano et al. 2017). El nivel de interactividad es adquirido por los elementos existentes permitiendo al usuario establecer una participación, y un proceso de comunicación con el sistema. Esta interacción puede motivar al niño a aprender. Sin embargo, para el diseñar un sistema interactivo se requieren analizar un conjunto de aspectos del niño, como: edad, género, escolaridad, intereses, comportamiento, entre otros. Estos aspectos deben tenerse en cuenta para diseñar el sistema de acuerdo a sus necesidades. De esta manera, un sistema interactivo puede involucrar elementos físicos y digitales, y puede servir como herramienta de apoyo para incentivar el pensamiento computacional. De tal manera, que puedan comprender cada uno de los objetos que se le presenten para estructurarlos de manera correcta siguiendo un proceso de aprendizaje.

Hoy en día, el pensamiento computacional se está comenzando a involucrar en los colegios en la asignatura de informática en las aulas regulares. Algunos colegios comienzan a partir desde el nivel escolar primaria y otros desde secundaria. Desde el 2009 hasta el 2013 un proyecto titulado “Programación de Computadores en la Educación Escolar” fue realizado por la Universidad ICESI de Cali (López, Eduteka), para apoyar el desarrollo del pensamiento computacional usando Scratch, donde los estudiantes aprendieron a seleccionar,

crear y manejar múltiples formas de contenidos multimedia (textos, imágenes, grabaciones de audio y animaciones) usando Scratch para los estudiantes de básica primaria. También, desde el 2017 hay un proyecto en marcha de innovación en la educación, realizado entre RENATA (Red Nacional de Investigación y Educación de Colombia) y la Universidad del País Vasco, España (Basogain et al. 2017), donde colaboran en la creación de un curso llamado introducción del pensamiento computacional en los colegios de Colombia en estudiantes que están finalizando 5 de primaria o iniciando bachillerato en el grado 6, edades comprendidas entre 10 y 12 años. El objetivo del curso es permitir a los estudiantes y profesores que se familiaricen con los conceptos básicos del PC y a futuro poder incluirlos en el plan de estudios de las escuelas colombianas para lograr que las nuevas generaciones no sean solamente consumidores de tecnología, sino productores de ella.

Por otro lado, las habilidades de orden superior de la Taxonomía de Bloom (Lorin et al. 2001) involucran habilidad de meta-cognición, pensamiento crítico-reflexivo, pensamiento inferencial y pensamiento creativo. La mayoría de estas habilidades se incluyen dentro del pensamiento computacional, con la idea de no formar programadores sino trabajar estas habilidades. Por lo que es importante insistir en la orientación del desarrollo de estas habilidades, ya que hay colegios que solo se centran en enseñar un lenguaje de programación, olvidando en enseñar estas habilidades y potencializarlas.

Este artículo está estructurado de la siguiente manera. En la sección 2 se describen algunos trabajos realizados en el diseño de sistemas interactivos como herramienta de apoyo al desarrollo del pensamiento computacional. En la sección 3, se realiza una descripción del planteamiento del problema relacionando conceptos de pensamiento computacional e interfaces tangibles. En la sección 4, se presenta una propuesta. Finalmente, conclusiones y trabajo futuro.

2. TRABAJOS RELACIONADOS

En esta sección se expondrá un conjunto de trabajos realizados, los cuales están relacionados con el trabajo de investigación a realizar.

Un primer trabajo propuesto en 1980 fue desarrollo por Papert con su lenguaje llamado LOGO (Papert 1980), el cual permite a los niños construir conocimientos. Papert desarrollo su enfoque basado en su experiencia con Piaget. Su idea consistió en presentar a los niños retos intelectuales que pudieran resolver mediante el desarrollo de programas en LOGO.

En 1988, se lanza una versión oficial de ALICE (Carnegie Mellon University) un software educativo libre y abierto, el está programado en JAVA. Alice fue diseñado para enseñar a programar.

En el 2012, el grupo Lifelong Kindergarten del Instituto Tecnológico de Massachusetts (MIT) crea una herramienta llamada Scratch, el cual es un lenguaje de programación que facilita la creación de arte interactivo, utilizado por niños, jóvenes, profesores y padres de familia en más de 150 países diferentes y está disponible en más de 40 idiomas. Scratch ayuda a los usuarios a aprender a pensar de forma creativa, razonar sistemáticamente, y trabajar de forma colaborativa. Gran variedad de colegios a nivel nacional e internacional

implementan esta herramienta para enseñar a programar a los estudiantes, lo cual permite desarrollar habilidades propias del pensamiento computacional.

En el 2013 un grupo de investigadores han propuesto herramientas para la enseñanza del pensamiento algorítmico. Fue un trabajo realizado por Danli Wang et al. (2013), donde desarrollan una herramienta de programación tangible para niños con edades entre los 4 a 10 años, cuyo objetivo es que los niños puedan programar, usando como método de enseñanza llamado e-blocks, el cual construye mediante bloques determinadas acciones que hacen que sus personajes se dirijan hacia direcciones específicas. Los bloques están contruidos con la interacción de objetos reales, donde usan un sensor de temperatura, un sensor de luz y un sensor de botón, con el objetivo que el niño vaya construyendo la secuencia de bloques y de misma manera el juego presenta una animación a través del ordenador de acuerdo a la secuencia construida.

Un trabajo propuesto en el 2016 por Maritza A. García et al. (2016) en la Universidad del Cauca, donde expone la enseñanza del pensamiento computacional usando estrategias basadas en la robótica. En este artículo se presentan algunas estrategias, como, clubes de robótica, y de cómo aplicar una metodología de enseñanza basada en problemas. Estas estrategias ayudan en el desarrollo de competencias de varios tipos y al desarrollo del pensamiento computacional en los estudiantes. Además, se mencionan los resultados del proyecto desarrollado en el Instituto Técnico Industrial Pascual Bravo de la ciudad Medellín-Colombia.

Teniendo en cuando las teorías de Piaget (Piaget 1950), que el niño necesita interactuar con objetos reales, ya que la actividad de un niño supone manipulación de objetos. Por lo que, se comienzan a crear elementos físicos o programación tangible, de tal manera que el niño pueda programar y ver esa interacción en un entorno real.

Tern (Horn et al. 2008), se crea un programa de computación físico usando construcción por bloques. Los usuarios deben darle un orden en que se conectan los bloques para generar programas. La captura del orden como están los bloques se usa cámara para capturar la imagen de la secuencia de los bloques. La imagen se transfiere al computadora para identificar el programa desarrollado y controlar funciones virtuales.

Por otro lado, en el 2013 Cubetto robot, con el propósito de apoyar en el desarrollo de competencias en el pensamiento computacional. La interacción es netamente física, y el niño construye la secuencia de movimientos por medio de un robot. Para este desarrollo se han basado en el aprendizaje de Montesorri para niños de 3 a 6 años. Cubetto fue desarrollo con el propósito de incluir como lenguaje de programación tangible en la enseñanza para el aprendizaje de conceptos de programación. Las áreas que apoya, son: comunicación, socio-emocional, matemáticas y razonamiento lógico.

Los trabajos mencionados anteriormente indican la importancia en integrar un estilo de enseñanza en niños, de tal manera que ayude a estimular el pensamiento computacional por medio de un sistema que pueda integrar una interacción física y digital.

3. PENSAMIENTO COMPUTACIONAL E INTERACCIÓN TANGIBLE

Hoy en día se han desarrollado diferentes sistemas orientados a enseñar programación, usando interacción con objetos reales. Estos sistemas permiten a los usuarios que puedan escribir programar usando elementos de interacción no tradicionales. La interacción tangible desde el área de la educación ha sido el foco de estudio, ya que propone una interacción no tradicional con elementos físicos y digitales. A través de una manipulación física de la representación de los objetos.

Las interfaces tangibles combinan control y la representación en un mismo objeto o espacio físico. El propósito es dar una forma física a la información digital, de tal manera que los usuarios puedan manipular los objetos y tener el control de la interacción. Para interactuar en el contexto del PC, hay que relacionar el estilo de aprendizaje que se usa como estrategia de enseñanza, que es la construcción por bloques, ya que el una serie de pasos ordenados permite solucionar un determinado problema.

Tomando la teoría de aprendizaje de Piaget (1950), el cual hace referencia que en las primeras etapas del desarrollo del niño, el juego simbólico establece una relación entre la representación y el lenguaje. Mientras que, Vygostky (1978), se centra en la interacción que se tenga con el contexto, el cual involucra como herramientas objetos físicos que puedan ayudar en el desarrollo psicológico del niño.

Papert (1980), siguiendo las ideas de Piaget, creo una teoría de aprendizaje llamada construccionismo, donde destaca la importancia de la acción, es decir del proceder activo en el proceso de aprendizaje. En palabras de Papert en la revista Time(1999): “El mejor aprendizaje no derivará de encontrar mejores formas de instrucción, sino de ofrecer al educando mejores oportunidades para construir”.

Por otro lado, el pensamiento computacional puede definirse como una metodología que implementa conceptos básicos de programación para resolver problemas, diseñar soluciones y realizar tareas. A su vez puede ayudar al estudiante a ser más crítico a la hora de tomar decisiones frente a un problema. Por lo que, no debe relacionarse con que solo debe enseñarse solo a personas interesadas con seguir en el ámbito de la ingeniería informática.

En la educación superior la mayoría de los estudiantes cuando matriculan materias relacionadas con programación. Se observa que hay un alto índice de deserción por parte de los estudiantes. A su vez se encuentran un número de mujeres matriculadas en menor cantidad comparada con los hombres.

Teniendo en cuenta el alto índice de decadencia académica, y cómo afecta en la formación de un ingeniero en multimedia, surge la incógnita frente a los procesos de enseñanza y cómo se promueven competencias de pensamiento lógico y solución de problemas en la básica primaria a través de la clase de informática, de modo que, se hace importante identificar la didáctica y metodología que implementa el docente para potenciar en los niños de 9 a 10 años el pensamiento computacional. Hoy en día, dos instituciones académicas de Cali, en primaria incorpora la materia de informática e involucra el uso de herramientas interactivas como Scratch. Ésta ayuda a desarrollar la lógica, el análisis de problemas y la abstracción, logrando en los niños experiencias significativas al interactuar con este tipo de plataforma que les permite construir

de manera creativa historias, juegos y animaciones. A su vez, es importante revisar las estructuras curriculares y evidenciar cómo se viene trabajando al interior de las instituciones educativas la asignatura de informática, las estrategias y herramientas que pone en circulación y uso el docente para promover la creatividad e innovación mediante diferentes programas o plataformas.

Hoy en día, el crecimiento de las Tecnologías de la Información y Comunicación (TIC) ha servido para incluirlas dentro de las aulas como material de apoyo para el docente, por lo que se busca con las tecnologías potenciar las habilidades de abstracción y resolución de problemas, de tal manera que pueda motivar al niño durante el aprendizaje. Sin embargo, otro de los retos en Colombia en introducir formalmente el estudio de los lenguajes de programación en los colegios es el número limitado de profesores que puedan enseñar esta materia.

Lo que, puede definirse algunos interrogantes: ¿Cómo se puede apoyar al desarrollo del pensamiento computacional usando interacción tangible a niños en edades de 9 a 10 años?, ¿Qué estrategias educativas se están aplicando para el desarrollo del pensamiento computacional?, ¿Existen metodologías de enseñanza para el pensamiento computacional para niños?.

4. PROPUESTA

La propuesta del proyecto es desarrollar un sistema interactivo usando elementos físicos (Hardware) y digitales (Software), de tal manera que motive a los niños entre los 9 y 10 años de edad a desarrollar habilidades del pensamiento computacional en el aula regular. Para el diseño del sistema interactivo se sigue la metodología de diseño centrado en el usuario (Norman, & Draper 1986), con el objetivo de encontrar las necesidades del niño en las aulas regulares, y ajustar el sistema de acuerdo a sus necesidades. La metodología de diseño centrado en el usuario se compone de las siguientes etapas: especificación del contexto de uso, especificación de los requisitos, producción de las soluciones de diseño y evaluación.

A continuación se describen las fases que se han realizado:

4.1 Especificación del contexto de uso

En esta fase se aplican unos métodos para evaluar la experiencia del niño en el contexto del pensamiento computacional dentro y fuera del aula.

Fuera del aula

Se selecciona un grupo de 10 niños entre 9 a 10 años (Fig. 1), que pertenecen a diferentes colegios (privado y público). La actividad que se realiza con ellos es una observación directa al interactuar con un juego físico llamado Code And Go (Danli et al. 2013), y Go Robot Mouse Activity Set (Go Robot). Ambos son juegos orientados a la enseñanza de la programación a partir de niños de 4 años. En este juego a los niños se les presenta un problema y deben encontrar una solución. En esta actividad, ellos deben encontrar un camino dando una conjunto de instrucciones de manera ordenada para que el ratón pueda llegar al queso.

En esta actividad se puede observar que los niños presentan un poco de dificultad al inicio del juego, debido a que no comprenden el objetivo del mismo, pero al momento que logran aprender la estrategia es muy fácil para ellos. En esta

actividad se observa que la imaginación de los niños no tiene límites, por ejemplo cuando cambiaban las reglas del juego convirtiéndolo más interesante.



Fig. 1.

Niños jugando con Code and Go Robot Mouse Activity Set.

Dentro del aula

Se seleccionan dos instituciones educativas. Una es de carácter privado, calendario B y mixto. Otra es un colegio público, que tiene como objetivo aportar educación de vanguardia a niños de escasos recursos. Ambas instituciones se encuentran ubicadas en Cali-Valle del Cauca, donde los cursos a trabajar son 3, 4 y 5 de primaria, niños y niñas entre edades de 9 a 10 años. Es importante decir que el nombre de ambas instituciones se reserva, debido a ordenes recibidas por ellas mismas.

Se realiza el ejercicio de observación en las aulas, y una entrevista a los docentes de informática de ambas instituciones. Se formulan algunas preguntas, como: ¿qué tipo de herramientas tecnológicas usa en el curso de informática?, ¿Cómo se puede evidenciar el aprendizaje durante el curso?, Conoce acerca del término pensamiento computacional? , entre otras. Es importante mencionar que los docentes no conocían el término. Sin embargo, cuando se les mostró un esquema (Fig. 2), de los diferentes tipos de pensamientos que pueden involucrar el pensamiento computacional, han indicado que cada uno de estos los han trabajado de manera independiente sin integrarlos en uno solo.



Fig 2.

Estructura del pensamiento computacional

A continuación se realiza una descripción más detallada de la observación realizada en el aula de cada colegio.

Colegio I (privado): La clase de informática es brindada por medio de un video beam, donde el docente proyecta un código en Scratch para realizar un videojuego. Los niños copian el código que se les muestra para reproducirlo en Scratch. El proceso de enseñanza que tienen, se compone de 3 fases, son: identificación (3 semanas), implementación (3 semanas) y diseño (2 semanas). El docente indica que los estudiantes tienen la capacidad de crear desde cero un

programa a partir de la etapa de diseño. Además hace los siguientes aportes los cuales se consideraron relevantes para tener en cuenta al momento de planificar el sistema.

Aportes del docente:

- Los niños que tienen problemas para socializar con las personas son los que tienen mayor concentración y a la vez mejores habilidades al momento de programar.
- Los niños imperativos tienen más concentración con los juegos físicos.

De la observación en el aula surgieron algunas preguntas, como:

- Si el estudiante hace el ejercicio de replicar lo que se proyecta en el video beam. Las dos primeras etapas (identificación e implementación) se brindan en un tiempo de 6 semanas, por lo que genera la siguiente incógnita ¿realmente los niños están desarrollando algún pensamiento con esta práctica?
- ¿Es suficiente el tiempo que se le da a la etapa de diseño, alrededor de 2 semanas para que el niño ponga en práctica según lo aprendido con la herramienta, la planeación de estrategias, creatividad o la resolución de problemas?

Colegio II (público) : Observación realizada al taller de Scratch, el cual se brinda de forma extracurricular en horas de la tarde para los estudiantes que voluntariamente deseen asistir. En el aula se encontraban niños de 3, 4 y 5 de primaria de 8 y 9 años de edad. Se puede observar que los estudiantes trabajan proyectos de forma individual, lo que permite que el resto de sus compañeros puedan practicar el aprendizaje del algebra. El modo de enseñanza para la herramienta Scratch es: introducción, enseñanza de algoritmos aplicados a la vida diaria, luego se pasa al manejo de la herramienta. Con la visita se puede mencionar que los estudiantes que asisten al taller tienen un gran gusto por la programación, les encantan los retos y también son muy sociables.

También se realizaron encuestas a los estudiantes de ambos colegios, con el objetivo de conocer la cercanía que tienen con la herramienta Scratch, y conocer su opinión personal acerca de la herramienta. También se quiere evaluar de qué forma aplican la resolución de problemas mediante un juego sencillo, cuyo objetivo es escribir los pasos que el ratón debe realizar para llegar al queso.

Resultados obtenidos de las encuestas realizadas a ambas instituciones. La primera institución se encuestaron a un total de 16 niños del grado 5 (Fig 3) y la segunda institución a un total de 4 niños participantes del taller de Scratch (Fig 4).

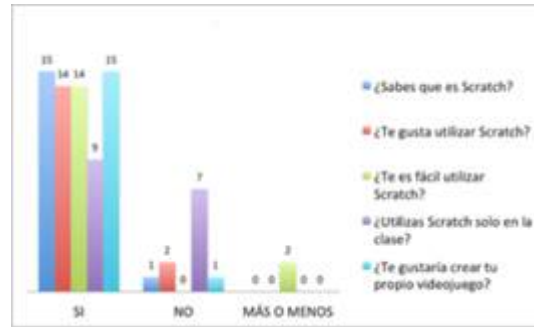


Fig 3.

Resultados de la encuesta realizada en el Colegio 1 (Privado) Valle del Cauca.

Como se observa en los resultados obtenidos (Fig 3-4), ambos colegios conocen scratch. Sin embargo, los niños no se sienten motivados para crear un videojuego. Además, la mayoría de ellos usan scratch solo en la clase de informática.

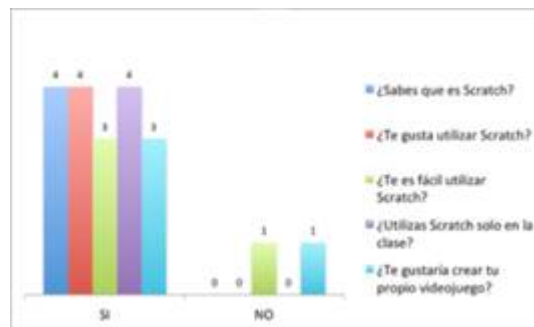


Fig 4.

Resultados de la encuesta realizada en el Colegio 2 (Público), Valle del Cauca.

A partir de los resultados obtenidos de la encuesta, se analizaron algunas definiciones del PC, dada por algunos investigadores:

- PC implica la resolución de problemas, el diseño de sistemas, y la comprensión de la conducta humana, haciendo uso de los conceptos fundamentales de la informática (Jeannette 2006).
- PC es el proceso de pensamiento involucrado en la formulación de problemas, de tal manera que sus soluciones puedan ser representadas como pasos computacionales discretos y algoritmos (Aho 2012).
- PC es un conjunto de habilidades y técnicas de solución de problemas, que los ingenieros de software usan para escribir los programas informáticos que subyacen a las aplicaciones que usamos a diario, como Google.
- El PC, definido por Zapata (Zapata 2015) se puede vincular a los siguientes componentes, como: pensamiento divergente, creatividad, pensamiento abstracto, iteración, recursividad, patrones, entre otros.
-

Tomando las definiciones dadas, se puede decir que el pensamiento computacional involucra diferentes pensamientos que ayudan a un niño o niña que pueda adquirir competencias relacionadas con la solución y creación.

Se realiza una evaluación con los niños. Por lo que, es un cuestionario para evaluar el pensamiento computacional. Para realizar este test de pensamiento computacional, se baso en algunos test propuestos por investigadores (Román-González 2015) (Gouws, et al. 2013). A partir de ello se ha propuesto un cuestionario para indagar acerca del pensamiento computacional .



Fig 5.

Pregunta 1 del test de pensamiento computacional realizado a los niños

El cuestionario consta de dos preguntas (Fig 5- 6). La primera pregunta está relacionada con la asociación del código scratch con la realidad. Por lo que, se les solicita analizar el código y que describan lo que hace el código.

La segunda pregunta, se les presenta un laberinto, donde se debe seleccionar los movimientos correctos para llegar el ratón hasta el queso. En esta pregunta se realizaron dos modelos (Fig 6a y 6b), donde el modelo correspondiente a la Fig 6a es aplicado al al primer colegio y el modelo de la Fig 6b al segundo colegio. El objetivo de esta pregunta, es que encuentren la secuencia de pasos para que el ratón pueda llegar al queso.

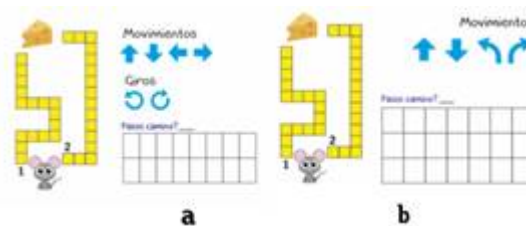


Fig 6.

Pregunta 2 del cuestionario del pensamiento computacional realizado a los niños.

Se describen algunas observaciones encontradas, como:

- Cinco niños se equivocaron en algunos pasos o no tenían sentido sus indicaciones.
- Siete niños respondieron exactamente la forma 2 del camino de la Fig 6b.
- Algunos niños responden de forma diferente, sus respuestas tienen sentido y solucionan el objetivo del juego. Por lo que, no se considera que estén de forma incorrecta. A continuación se mencionan algunos casos encontrados en la evaluación.

Caso 1: El camino 1 fue seleccionado por un niño, y se puede observar que interpreta los giros para las orientaciones de izquierda y derecha según el camino.

Caso 2-4: Un grupo de niños toman los giros para las orientaciones de izquierda y derecha. A la vez consideran que es suficiente indicar una sola vez cada señal.

Algunos de los aspectos encontrados son:

- tres niños toman el segundo camino, de los cuales un niño lo realiza como se esperaba, otro lo resuelve de forma diferente. Por último, otro no lo logró.
- Una niña lo resuelve de forma diferente. Por lo que, no se puede concluir que lo realiza de manera incorrecta, ya que ella tomó las flechas de giro como si fueran orientaciones de derecha e izquierda.

4.2 Especificación de los requerimientos

A partir del análisis realizado y de los aportes brindados por los docentes de informática de ambas instituciones. Se tiene en cuenta para el diseño del primer prototipo los siguientes criterios:

1. Algunos niños presentan mayor concentración al interactuar con elementos físicos.
2. Algunos presentan mayor concentración y mejor desarrollo de habilidades al interactuar con el computador.
3. Automatizar soluciones a través del PC, de tal manera que el niño debe dar una serie de pasos discretos y ordenados para lograr alcanzar la actividad.
4. Identificar algunas mecánicas de juego que puedan incorporarse.
5. Los niños resuelven los problemas según como los entiendan e interpretan.
6. El sistema debe tener múltiples soluciones para que los niños puedan planificar cuál es su mejor opción.
7. Un grupo de niños hicieron ver en la realización del juego del ratón que no es necesario repetir una misma señal varias veces.
8. Necesitan tener una retroalimentación permanente de cada una de las acciones que realiza.
9. Usar varios canales de retroalimentación, visual y auditivo.

Por otro lado, al incorporar el contenido digital como un juego, se deberán tener en cuenta algunas mecánicas de juego, como: puntuación, niveles, retos y retroalimentación.

4.3 Producir Soluciones de Diseño

De acuerdo a los requerimientos que se plantea, se propone un primer prototipo para el diseño del sistema interactivo, el cual puede estar sujeto a cambios a medida que se avanza en las investigaciones o pruebas con estudiantes. El sistema será en modo de juego que interactuará con elementos físicos (hardware) y digitales (software). El elemento digital es un videojuego (Fig. 7), el cual está relacionado con un pirata cuyos objetivos, niveles y estructura están en proceso de construcción. El objetivo es que el pirata llegue a su espada. Por lo que, el jugador (niño) debe elegir el camino que mejor considere teniendo en cuenta que en el camino se puede encontrar con retos, los cuales evitarán que llegue hasta su objetivo principal.



Fig 7.

Propuesta del videojuego para el pensamiento computacional.

El hardware (Fig. 8) está conformado por un tablero donde se colocan las fichas con las señales que el niño elija para realizar su camino. También se tiene en cuenta unos contadores para que el jugador indique la cantidad de veces que desea repetir el mismo paso, hasta el momento se han definido las siguientes interacciones:



Fig 8.

Propuesta del tablero físico que interactuará con el videojuego.

El juego tendrá dos niveles de dificultad, en el primer nivel se usaran las fichas de movimientos de orientación y condicionales. En el segundo nivel se usaran los ciclos y funciones.

Cada ficha estará representada por la acción a realizar, y cada acción estará identificada con un color específico. La forma en que las fichas interactuarán con el juego es por medio de un sensor de colores, el cual se enviará los datos de manera serial usando como plataforma para la construcción del juego, Unity.

El sensor de color funciona usando una fotoresistencia y un diodo RGB. El sensor intenta de una forma que el niño pueda identificar visualmente cada función de la ficha acompañado de la representación simbólica asociado a un color. El lector de sensor tendrá la forma de una brújula, ya que la interacción debe estar relacionado con el contexto del juego que está relacionado con piratas.

5. CONCLUSIONES Y TRABAJO FUTURO

Hoy en día, se están incorporando dentro de las aulas estrategias lúdicas, de tal manera que logre motivar al estudiante durante su aprendizaje. También se ha mencionado en trabajos relacionados que el pensamiento computacional no solo involucra a nivel computacional, sino que integrar un conjunto de

pensamientos que pueden ayudar al niño a potenciar ciertas competencias en las primeras etapas de desarrollo. Además, la mayoría de las investigaciones que se han mencionado involucra el uso de objetos tangibles a través del uso de dispositivos no tradicionales, lo que lleva a que el niño logre motivarse más. Sin embargo, diseñar un sistema que involucre elementos físicos y digitales, requiere realizar un análisis de las diferentes aspectos del niño en las aulas regulares, ya que el tipo de interacción física o digital puede verse afectada de acuerdo al perfil del estudiante, como género, edad, nivel académico, entre otros.

Como trabajo futuro se quiere implementar el sistema y a la vez evaluar la propuesta con un grupo de niños de colegios privados y públicos.

Referencias

- Jeannette M. Wing 2006. Computational Thinking. Communications of the ACM, vol 49 No 3, pp 1- 2.
- Jeannette M. Wing 2009. Computational Thinking About Computing. Evening Lecture Series in Pensacola In IHMC [en línea], consultado: 29 enero del 2018 <https://www.ihmc.us/lectures/jeannette-m-wing/>.
- Scratch, [en línea], consultado: 03 de agosto 2018 <https://scratch.mit.edu/>
- S. Cano, C. Manresa-Yee, C. A. Collazos, V. Peñeñory, Javier Varona 2017. Chapter: Interactive Systems Design oriented to children with special needs. Book HCI for Children with Disabilities edited by Josefina Guerrero- Garcia, Juan Manuel Gonzalez-Calleros, Jaime Munoz- Arteaga, Cesar Collazos.
- Juan Carlos López, [en línea], consultado: 03 de agosto 2018, <http://eduteka.icesi.edu.co/articulos/ProgramacionEdEcolar>.
- Xabier Basogain, M.A Olabe, Juan Carlos Olabe, Mauricio Javier Rico, Leonardo Rodríguez, Miguel A. Amortegui 2017. Pensamiento computacional en las escuelas de Colombia: colaboración internacional de innovación en la educación, [en línea], consultado: 02 de agosto 2018, <http://recursos.portaleducoas.org/sites/default/files/5188.pdf>
- David Krathwohl, James D. Raths, Kathleen A. Cruikshank, Merlin Wittrock, Paul Pintrich, Peter W Airasian, and Richard E. Mayer, 2001, A Taxonomy for Learning, Teaching, an Assessing: A Revision of Bloom's Taxonomy of Educational Objectives;. Longman pp 352.
- Danli Wang, Yang Zhang, and Shengyong Chen, 2013 E-Block: A Tangible Programming Tool with Graphical Blocks. Mathematical Problems in Engineering, vol Article ID 598547.
- García, M. A., Deco, C., And Collazos, C. A, 2016, Estrategias basadas en robótica para apoyar el pensamiento computacional. In XXII Congreso Argentino de Ciencias de la Computación.
- Papert, S., 1980, Mindstorms: Children, computers, and powerful ideas. New York: Basic Books
- Alice: Tell Stories, build games. [en línea], consultado: 02 de agosto 2018, <http://www.alice.org/>.
- Piaget J. and Inhelder B, 1950, The Psychology of the child.
- M.S. Horn, E.T. Solovey, R.J. Jacob, 2008, Tangible programming and informal science learning: making TUIs work for museums, in: Proceedings of the 7th international conference on Interaction design and children, ACM, pp. 194-201.

- Cubetto, [en línea], consultado: 02 de agosto 2018 <https://www.primotoys.com/>
- Vigotsky L.S. *Mind in society*, 1978, Harvard University Press, Cambridge M.A.
- Papert, S, 1999, *Time magazine Special Issue: The century's Greatest Minds*, pp 105, [en línea], consultado: 02 de agosto 2018, <http://papert.org/articles/Papertonpiaget.html>, última visita 20 de mayo del 2018.
- Norman, D. A., & Draper, S. W. (Eds.), 1986, *User centred system design*. Hillsdale, NJ:Erlbaum.
- Code & Go Robot, [en línea], consultado: 03 de agosto del 2018, https://www.youtube.com/watchv=_tGb9bLe0YA
- Román-González, Marcos & Pérez-González, Juan-Carlos & Jiménez-Fernández, Carmen, 2015, *Test de Pensamiento Computacional: diseño y psicometría General*, *Computational Thinking Test: design & general psychometry*. 10.13140/RG.2.1.3056.5521,
- Gouws, L., Bradshaw, K., & Wentworth, P, 2013, *First year student performance in a test for computational thinking*. *Proceedings of the South African Institute for Computer Scientists and Information Technologists Conference*, 271-277.
- Aho A, 2012, *Computation and Computational Thinking*. *The computer Journal*, vol 55, No 7.
- Google, [en línea], consultado: 03 de agosto del 2018 <https://computationalthinkingcourse.withgoogle.com/uit>.
- Zapata M, 2015, *Pensamiento Computacional: una nueva alfabetización digital* *Computational Thinking: a new digital literacy*. *Red Revista de educación a distancia*, 46 (4).
- Rosa Barrera Capot, Rosa Montaña Espinoza, 2015 *Desarrollo del pensamiento computacional con Scratch*. *Nuevas Ideas en Informática Educativa*, pp 616-620.