



Universidad de San Carlos de Guatemala
Facultad de Ingeniería
Escuela de Ingeniería en Ciencias y Sistemas

**IMPLEMENTACIÓN DEL PROGRAMA DE CAPACITACIÓN Y DESARROLLO EN CIENCIAS
DE LA COMPUTACIÓN UTILIZANDO EL MODELO DE ENSEÑANZA “DESPUÉS DE LA
ESCUELA”, EN LA ESCUELA OFICIAL NO. 53 DAVID VELA**

Santos William Jonathán Quiacain Navichoc
Asesorado por el Ing. José Ricardo Morales Prado

Guatemala, agosto de 2015

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA



FACULTAD DE INGENIERÍA

**IMPLEMENTACIÓN DEL PROGRAMA DE CAPACITACIÓN Y DESARROLLO EN CIENCIAS
DE LA COMPUTACIÓN UTILIZANDO EL MODELO DE ENSEÑANZA “DESPUÉS DE LA
ESCUELA”, EN LA ESCUELA OFICIAL NO. 53 DAVID VELA**

TRABAJO DE GRADUACIÓN

PRESENTADO A LA JUNTA DIRECTIVA DE LA
FACULTAD DE INGENIERÍA
POR

SANTOS WILLIAM JONATHÁN QUIACAIN NAVICHOC
ASESORADO POR EL ING. JOSÉ RICARDO MORALES PRADO

AL CONFERÍRSELE EL TÍTULO DE

INGENIERO EN CIENCIAS Y SISTEMAS

GUATEMALA, AGOSTO DE 2015

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA
FACULTAD DE INGENIERÍA



NÓMINA DE JUNTA DIRECTIVA

DECANO	Ing. Pedro Antonio Aguilar Polanco
VOCAL I	Ing. Angel Roberto Sic García
VOCAL II	Ing. Pablo Christian de León Rodríguez
VOCAL III	Inga. Elvia Miriam Ruballos Samayoa
VOCAL IV	Br. Narda Lucía Pacay Barrientos
VOCAL V	Br. Walter Rafael Véliz Muñoz
SECRETARIA	Inga. Lesbia Magalí Herrera López

TRIBUNAL QUE PRACTICÓ EL EXAMEN GENERAL PRIVADO

DECANO	Ing. Alfredo Enrique Beber Aceituno
EXAMINADORA	Inga. Floriza Felipa Ávila Pesquera de Medinilla
EXAMINADOR	Ing. Marlon Antonio Pérez Türk
EXAMINADORA	Inga. Susan Verónica Gudiel Herrera
SECRETARIO	Ing. Hugo Humberto Rivera Pérez

HONORABLE TRIBUNAL EXAMINADOR

En cumplimiento con los preceptos que establece la ley de la Universidad de San Carlos de Guatemala, presento a su consideración mi trabajo de graduación titulado:

IMPLEMENTACIÓN DEL PROGRAMA DE CAPACITACIÓN Y DESARROLLO EN CIENCIAS DE LA COMPUTACIÓN UTILIZANDO EL MODELO DE ENSEÑANZA “DESPUÉS DE LA ESCUELA”, EN LA ESCUELA OFICIAL NO. 53 DAVID VELA

Tema que me fuera asignado por la Dirección de la Escuela de Ingeniería en Ciencias y Sistemas, con fecha 9 de abril de 2014.


Santos William Jonathán Quiacain Navichoc

Guatemala, 22 de septiembre de 2014

Ing. Silvio José Rodríguez Serrano
Director de la Unidad de EPS
Facultad de Ingeniería
Universidad de San Carlos de Guatemala

Estimado Ingeniero Silvio José Rodríguez Serrano:

Por este medio atentamente le informo que procedo a **DAR POR APROBADA** la Práctica del Ejercicio Profesional Supervisado (E.P.S.) del estudiante universitario **SANTOS WILLIAM JONATHAN QUIACAIN NAVICHOC** de la Carrera de Ingeniería en Ciencias y Sistemas, con carné No. **200011128**, cuyo título es **"IMPLEMENTACIÓN DEL PROGRAMA DE CAPACITACIÓN Y DESARROLLO EN CIENCIAS DE LA COMPUTACIÓN UTILIZANDO EL MODELO DE ENSEÑANZA "DESPUÉS DE LA ESCUELA", EN LA ESCUELA OFICIAL NO. 53 DAVID VELA"**, solicitándole darle el trámite respectivo.

Sin otro particular, me es grato suscribirme.

Atentamente,


Ing. José Ricardo Morales Prado
Asesor del proyecto de EPS

José Ricardo Morales Prado
INGENIERO EN SISTEMAS
COLEGIADO No. 4746



Guatemala, 25 de mayo de 2015.
REF.EPS.DOC.395.05.2015.

Ing. Silvio José Rodríguez Serrano
Director Unidad de EPS
Facultad de Ingeniería
Presente

Estimado Ingeniero Rodríguez Serrano .

Por este medio atentamente le informo que como Supervisora de la Práctica del Ejercicio Profesional Supervisado, (E.P.S) del estudiante universitario de la Carrera de Ingeniería en Ciencias y Sistemas, **Santos William Jonathán Quiacain Navichoc** carné No. 200011128 procedí a revisar el informe final, cuyo título es **IMPLEMENTACIÓN DEL PROGRAMA DE CAPACITACIÓN Y DESARROLLO EN CIENCIAS DE LA COMPUTACIÓN UTILIZANDO EL MODELO DE ENSEÑANZA "DESPUÉS DE LA ESCUELA", EN LA ESCUELA OFICIAL No. 53 DAVID VELA.**

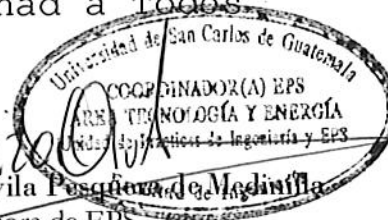
En tal virtud, **LO DOY POR APROBADO**, solicitándole darle el trámite respectivo.

Sin otro particular, me es grato suscribirme.

Atentamente,

"Id y Enseñad a Todos"

Inga. Floriza Felipa Avila *Pescadora de Medinilla*
Supervisora de EPS
Área de Ingeniería en Ciencias y Sistemas



FFAPdM/RA



Guatemala, 25 de mayo de 2015.
REF.EPS.D.263.05.2015.

Ing. Marlon Antonio Pérez Turk
Director Escuela de Ingeniería Ciencias y Sistemas
Facultad de Ingeniería
Presente

Estimado Ingeniero Perez Turk.

Por este medio atentamente le envío el informe final correspondiente a la práctica del Ejercicio Profesional Supervisado, (E.P.S) titulado **IMPLEMENTACIÓN DEL PROGRAMA DE CAPACITACIÓN Y DESARROLLO EN CIENCIAS DE LA COMPUTACIÓN UTILIZANDO EL MODELO DE ENSEÑANZA "DESPUÉS DE LA ESCUELA", EN LA ESCUELA OFICIAL No. 53 DAVID VELA**, que fue desarrollado por el estudiante universitario **Santos William Jonathán Quiacain Navichoc** carné No. **200011128**, quien fue debidamente asesorado por el Ing. José Ricardo Morales Prado y supervisado por la Inga. Floriza Felipa Ávila Pesquera de Medinilla.

Por lo que habiendo cumplido con los objetivos y requisitos de ley del referido trabajo y existiendo la aprobación del mismo por parte del Asesor y la Supervisora de EPS, en mi calidad de Director apruebo su contenido solicitándole darle el trámite respectivo.

Sin otro particular, me es grato suscribirme.

Atentamente,
"Id y Enseñad a Todos"

Ing. Silvio José Rodríguez Serrano
Director Unidad de EPS



SJRS/ra



Universidad San Carlos de Guatemala
Facultad de Ingeniería
Escuela de Ingeniería en Ciencias y Sistemas

Guatemala, 22 de Julio de 2015

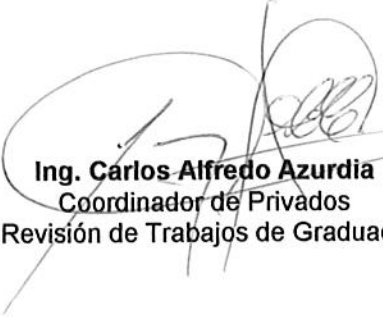
Ingeniero
Marlon Antonio Pérez Türk
Director de la Escuela de Ingeniería
En Ciencias y Sistemas

Respetable Ingeniero Pérez:

Por este medio hago de su conocimiento que he revisado el trabajo de graduación-EPS del estudiante **SANTOS WILLIAM JONATHÁN QUIACAIN NAVICHOC** carné 2000-11128, titulado: "IMPLEMENTACIÓN DEL PROGRAMA DE CAPACITACIÓN Y DESARROLLO EN CIENCIAS DE LA COMPUTACIÓN UTILIZANDO EL MODELO DE ENSEÑANZA 'DESPUÉS DE LA ESCUELA', EN LA ESCUELA OFICIAL No. 53 DAVID VELA", y a mi criterio el mismo cumple con los objetivos propuestos para su desarrollo, según el protocolo.

Al agradecer su atención a la presente, aprovecho la oportunidad para suscribirme,

Atentamente,


Ing. Carlos Alfredo Azurdia
Coordinador de Privados
y Revisión de Trabajos de Graduación



E
S
C
U
E
L
A

D
E

C
I
E
N
C
I
A
S

Y

S
I
S
T
E
M
A
S

UNIVERSIDAD DE SAN CARLOS
DE GUATEMALA



FACULTAD DE INGENIERÍA
ESCUELA DE CIENCIAS Y SISTEMAS
TEL: 24767644

*El Director de la Escuela de Ingeniería en Ciencias y Sistemas de la Facultad de Ingeniería de la Universidad de San Carlos de Guatemala, luego de conocer el dictamen del asesor con el visto bueno del revisor y del Licenciado en Letras, del trabajo de graduación **“IMPLEMENTACIÓN DEL PROGRAMA DE CAPACITACIÓN Y DESARROLLO EN CIENCIAS DE LA COMPUTACIÓN UTILIZANDO EL MODELO DE ENSEÑANZA “DESPUÉS DE LA ESCUELA”, EN LA ESCUELA OFICIAL No. 53 DAVID VELA**”, realizado por el estudiante **SANTOS WILLIAM JONATHÁN QUIACAIN NAVICHOC**, aprueba el presente trabajo y solicita la autorización del mismo.*

“ID Y ENSEÑAD A TODOS”


Ing. Maylon Antonio Pérez Türk
Director, Escuela de Ingeniería en Ciencias y Sistemas



Guatemala, 25 de Agosto de 2015



El Decano de la Facultad de Ingeniería de la Universidad de San Carlos de Guatemala, luego de conocer la aprobación por parte del Director de la Escuela de Ingeniería en Ciencias y Sistemas, al trabajo de graduación titulado: **IMPLEMENTACIÓN DEL PROGRAMA DE CAPACITACIÓN Y DESARROLLO EN CIENCIAS DE LA COMPUTACIÓN UTILIZANDO EL MODELO DE ENSEÑANZA “DESPUÉS DE LA ESCUELA”, EN LA ECUELA OFICIAL No.53 DAVID VELA**, presentado por el estudiante universitario: **Santos William Jonathán Quiacain Navichoc**, y después de haber culminado las revisiones previas bajo la responsabilidad de las instancias correspondientes, se autoriza la impresión del mismo.

IMPRÍMASE.


Ing. Pedro Antonio Aguilar Polanco
Decano



Guatemala, agosto de 2015

/cc

ACTO QUE DEDICO A:

Dios

Ser Supremo, el Creador de los cielos y la tierra, el Todopoderoso que me ha dado el aliento de vida, el Misericordioso lleno de gracia, que me ha permitido existir en la tierra con su amor y bondad. El que me da la sabiduría cuando la he pedido, el que ha guiado mis pasos y ha estado conmigo en todo momento.

Jesús

Dios hecho carne, que me dio de su salvación, hombre valiente, revolucionario, que con su mirada refleja respeto, autoridad, sabiduría, inteligencia, alegría, paz, y con su ejemplo me ha permitido acercarme al Padre Celestial, el líder por excelencia, quien con su enseñanza estoy aprendiendo a vivir para la eternidad.

AGRADECIMIENTOS A:

Mis padres

Santos Quiacain Pérez y Rosalía Navichoc. Gracias por darme su amor incondicional, y brindarme todo su apoyo a lo largo de mi vida estudiantil.

Mis hermanos

Oswaldo, Ileana y Débora Quiacain Navichoc, por sus consejos y ánimos para culminar la carrera.

Mi novia

Melissa Ramírez, por su apoyo y motivación para consecución de esta meta.

Ing. Ricardo Morales Prado

Por ser una importante influencia en mi carrera, por su apoyo, su consejo y haber confiado en mí para emprender este proyecto.

SAT

Por permitir desarrollarme profesionalmente y obtener otros logros importantes en mi vida.

ÍNDICE GENERAL

ÍNDICE DE ILUSTRACIONES.....	III
GLOSARIO	V
RESUMEN.....	IX
OBJETIVOS.....	XI
INTRODUCCIÓN	XIII
1. MARCO TEÓRICO.....	1
1.1. Las ciencias de la computación.....	1
1.2. Impacto de las ciencias de la computación en la sociedad	2
1.3. ¿Por qué es importante la enseñanza de ciencias de la computación?	7
1.4. Guatemala, la educación y las ciencias de la computación.....	8
1.5. Ciencias de la computación en las escuelas públicas de Guatemala.....	11
1.6. Implementación de las ciencias de la computación como EPS	13
1.7. Duración del EPS	15
1.8. Atribuciones en el desempeño del EPS	15
2. CURSO DE SCRATCH	17
2.1. ¿Qué es Scratch?.....	18
2.2. Estadísticas de Scratch	19
2.3. Descripción del curso de Scratch	23
2.4. Entorno de trabajo de la herramienta Scratch	24
2.5. Contenido del cuaderno “Programación con Scratch”	25

2.6.	Sellos y apartados del cuaderno “Programación con Scratch”	26
3.	EVOLUCIÓN DEL CURSO	33
3.1.	Metodología	33
3.1.1.	Primeros pasos en Scratch	33
3.1.2.	Animando un paisaje	35
3.1.3.	Diálogos animados.....	37
3.1.4.	Carrera de autos	39
3.1.5.	Ampliando el juego Pong.....	39
3.1.6.	Tablas de multiplicar	39
3.2.	Dificultades encontradas	40
3.3.	Casos especiales	41
3.4.	Resultados	41
4.	PROPUESTA PARA MEJORAR LA IMPLEMENTACIÓN DEL FUTUROS PROYECTOS SIMILARES	45
4.1.	Metodología	45
4.2.	Escuela candidata antes de realizar el anteproyecto	45
4.3.	Coordinar el calendario	45
4.4.	Proyector.....	46
4.5.	Material de apoyo al estudiante	46
4.6.	Incluir el contenido de plano cartesiano y grados	47
	CONCLUSIONES.....	49
	RECOMENDACIONES	51
	BIBLIOGRAFÍA.....	53
	APÉNDICES.....	57

ÍNDICE DE ILUSTRACIONES

FIGURAS

1.	Los 10 países con más computadoras	3
2.	Tasa educativa, por nivel educativo, por año	9
3.	Hogares con TV, computador e internet, 2012.....	10
4.	Tendencias mensuales de actividad	19
5.	Distribución por edades de los nuevos scratchers	20
6.	Scratchers alrededor del mundo	21
7.	Proyecto compartidos mensualmente	22
8.	Entorno de trabajo de Scratch.....	24
9.	Programas en bloques de Scratch	25

GLOSARIO

Algoritmo de Programación	Conjunto de instrucciones o reglas bien definidas y ordenadas, que le indican a la computadora los pasos específicos para llevar a cabo una tarea. Los algoritmos son rigurosamente definidos para que la computadora pueda interpretarlos.
CodeAcademy	Plataforma web que permite a las personas aprender sobre diversos lenguajes de programación como html, php, JavaScript.
EPS	Ejercicio Profesional Supervisado.
Excel	Paquete de software destinado al procesamiento de hojas de cálculo, utilizado normalmente en tareas financieras y contables.
Hardware	Conjunto de componentes físicos tecnológicos, que trabajan o interactúan de algún modo con la computadora.
Hipertexto	Herramienta de software con estructura no secuencial, que permite crear, agregar, enlazar y compartir información de diversas fuentes por medio de enlaces asociativos.

HTML	Hiper Text Mark-up Language. Lenguaje de marca de hipertextos, es el lenguaje de programación para la presentación de texto y contenido de páginas web y que puede ser interpretado por los navegadores de internet.
IBM	International Business Machines Corporation; es una empresa multinacional estadounidense de tecnología y consultoría se dedica a fabricar y comercializar hardware y software para computadoras, y ofrece servicios de infraestructura, alojamiento de internet, y consultoría en una amplia gama de áreas relacionadas con la informática, desde computadoras centrales hasta nanotecnología.
Internet	Conjunto descentralizado de redes de comunicación interconectadas que utilizan la familia de protocolos TCP/IP.
Implementación	Poner en funcionamiento, aplicar los métodos y medidas necesarios para llevar algo a cabo.
Pensamiento computacional	Es un proceso de solución de problemas que permitan usar computadores y otras herramientas, organizando los datos de manera lógica y representar datos mediante abstracciones, automatizar soluciones mediante pensamiento algorítmico, Identificar, analizar e implementar

posibles soluciones con el objeto de encontrar la combinación de pasos.

Photoshop

Paquete de software destinado a la edición de gráficos, principalmente para el retoque de fotografías y gráficos.

Scratch

Entorno de aprendizaje de lenguaje de programación, que permite a las personas sin sentido de la programación, obtener resultados sin tener que escribir de manera sintácticamente correcta.

Scratching

Es una técnica utilizada para producir sonidos característicos a través del movimiento de un disco de vinilo hacia delante y hacia atrás, sobre un tocadiscos, creando piezas de sonido repetitivas.

Software

Programa o aplicación, diseñado para realizar tareas específicas.

Turntablism

Es el arte de arreglar o crear música mediante efectos de sonido y manipulación de las rutinas de rotación y lectura de los discos de vinilo sobre un plato giradiscos, creando así una pieza musical con distintos extractos de sonidos repetitivos.

Tynker	Entorno de aprendizaje de lenguaje de programación que introduce a las personas al tema de la programación similar a Scratch.
Website	Sitio web; es una colección de páginas de internet relacionadas y comunes a un dominio de internet o subdominio en la World Wide Web.
Word	Paquete de software destinado al procesamiento de textos.
World Wide Web	Red informática mundial, comúnmente conocida como la web, es un sistema de distribución de documentos de hipertexto o hipermedios interconectados y accesibles vía Internet.

RESUMEN

El desarrollo de las personas es un factor importante en la sociedad, los principios y valores aunados con la superación deben ser el objetivo que el país debe tener. Hoy en día leer, escribir, saber inglés y ciencias de la computación son necesarios para obtener mejores oportunidades en un mundo cambiante.

Las ciencias de la computación abarcan una gama de temas y se pueden aplicar a cualquier campo como la medicina, ingeniería, economía, odontología, veterinaria, educación, comunicación, entre otros.

Durante un periodo de seis meses se llevó a cabo este EPS en la Escuela Oficial No. 53 David Vela, con la finalidad de que los estudiantes practicaran el pensamiento computacional con Scratch, creando su propio programa para desarrollar en ellos la habilidad de resolver problemas con instrucciones básicas en forma de bloques que brinda Scratch. Esto produce en ellos la búsqueda y el interés en las ciencias de la computación; también les ayuda a ver el problema de distintos puntos de vista, para resolverlo de cualquier perspectiva.

El presente informe es un documento completo con toda la información relacionada con la ejecución del Ejercicio Profesional Supervisado (EPS). Se incluye el programa del curso tal y como se presentó a los estudiantes, los enunciados de los ejercicios y proyectos, los diversos inconvenientes y problemas a resolver, así como los resultados del proyecto.

Asimismo se detalla una serie de recomendaciones que pueden mejorar la implementación de este programa en otra escuela, ya que ese fue uno de los objetivos planteados en el anteproyecto del trabajo de Ejercicio Profesional Supervisado (EPS).

OBJETIVOS

General

Implementar un programa de capacitación en ciencias de la computación en una escuela pública, que sea autosostenible, dejando una guía para que pueda replicarse a más escuelas, brindando al estudiante la oportunidad de desarrollar en ellos el pensamiento computacional y familiarizarse con las ciencias de la computación.

Específicos

1. Verificar que el estudiante comprenda el significado de ciencias de la computación y el pensamiento computacional.
2. Proponer que el estudiante aprenda sentencias de programación por medio de bloques, funciones y algoritmos de programación.
3. Desarrollar el pensamiento computacional de una forma sencilla y práctica por medio de ejemplos, ejercicios y tareas.
4. Incentivar a los estudiantes en la búsqueda del conocimiento de las ciencias de la computación, creando un ambiente apropiado logrando aumentar su motivación.
5. Medir constantemente el progreso de los estudiantes por medio de evaluaciones periódicas sobre el aprendizaje obtenido.

6. Capacitar a una persona dentro de la escuela, que se haga responsable en darle continuidad al proyecto para que el programa sea autosostenible.

INTRODUCCIÓN

El presente informe corresponde al Ejercicio Profesional Supervisado (EPS), desarrollado en la Escuela Oficial Núm. 53 “David Vela” ubicada en la 31 ave. “C” 5-56, zona 7 colonia Centro América. Dicho EPS es el resultado de la propuesta del Ing. Ricardo Morales Prado, el cual plantea un programa de capacitación a estudiantes del nivel básico y diversificado en escuelas públicas, desarrollando en ellos la destreza de programación informática, con el fin de valerse por sí mismos al hacer uso de dicha destreza, así como lo hace un carpintero, un herrero, un pintor, entre otros.

El trabajo de EPS consistió en implementar el programa de capacitación y desarrollo en ciencias de la computación a estudiantes de quinto y sexto primaria por medio de la herramienta llamada Scratch; la cual permite a los niños de todas las edades desarrollar el pensamiento computacional de una forma muy sencilla, creando un sinfín de proyectos como ellos puedan imaginar, dándoles así el conocimiento básico en programación orientado a eventos, programación estructurada y paralelismo.

Los estudiantes de la escuela David Vela estuvieron muy entusiasmados al recibir este tipo de capacitación; esto los motivó a buscar la tecnología y a interesarse en el pensamiento computacional. De este modo se sienta un precedente para que otros estudiantes no graduados y con pénsum cerrado de la escuela de Ciencias y Sistemas de la Facultad de Ingeniería, apoyen a este programa, replicando el modelo de enseñanza en otras escuelas públicas, y de esta manera impactar en las nuevas generaciones para tener mejores oportunidades en la vida.

1. MARCO TEÓRICO

1.1. Las ciencias de la computación

“Las ciencias de la computación es la ciencia de usar las computadoras para resolver problemas. Sobre todo se trata de diseño de software (programas informáticos) y abordar cuestiones científicas fundamentales sobre la naturaleza de la computación, pero también implica muchos aspectos de hardware y arquitectura de los grandes sistemas informáticos que forman la infraestructura de las empresas comerciales y gubernamentales. Los científicos en computación (informáticos) trabajan de muchas maneras diferentes: trabajos teóricos en lápiz y papel sobre las bases y fundamentos, trabajo de programación en la computadora y el trabajo en equipo de colaboración en hacer la investigación y resolución de problemas.”¹

Las ciencias de la computación corresponden a una disciplina que consiste en el diseño y desarrollo de aplicaciones informáticas de sistemas, no es sobre el uso del software, tales como una hoja de cálculo como Excel, procesadores de palabras como Word o herramientas de manipulación de imágenes como Photoshop. Muchos paquetes de software son difíciles de dominar como Excel y Photoshop; es cierto que muchos trabajos dependen de la experiencia en el uso de ese tipo de herramientas, pero las ciencias de la computación no se refieren solo al uso de la herramienta.

¹ *Department of computer science, the George Washington University.*
<<http://www.seas.gwu.edu/>> [Consulta: 14 de mayo de 2014].

Las ciencias de la computación se refieren a los principios detrás de la construcción de los paquetes de software como los anteriores mencionados, acerca de los algoritmos utilizados en los juegos de computadora, sobre la tecnología detrás de la internet y sobre la arquitectura de dispositivos informáticos.

Las ciencias de la computación no se refieren a la construcción de teclados o monitores o a hacer los cables de la impresora que conecta a la computadora; estas son importantes para el funcionamiento de la computadora, así como la electricidad. El software informático consiste en interactuar programas donde cada uno de ellos es un conjunto de instrucciones que pueden ser ejecutadas en una computadora.

La computadora es una máquina, es un conjunto de chips y circuitos que no tienen vida, es una máquina “tonta” que sabe cómo ejecutar “instrucciones” (instrucciones como suma y resta; esto multiplica, aquello divide lo otro). Entonces los programas de software son colecciones de instrucciones que permiten alcanzar objetivos finales a nivel superior. En cierto sentido la “inteligencia” reside en el software y es la dificultad de crear software fiable e inteligente que ha hecho de esta joven disciplina de las ciencias de la computación un campo grande y diverso.

1.2. Impacto de las ciencias de la computación en la sociedad

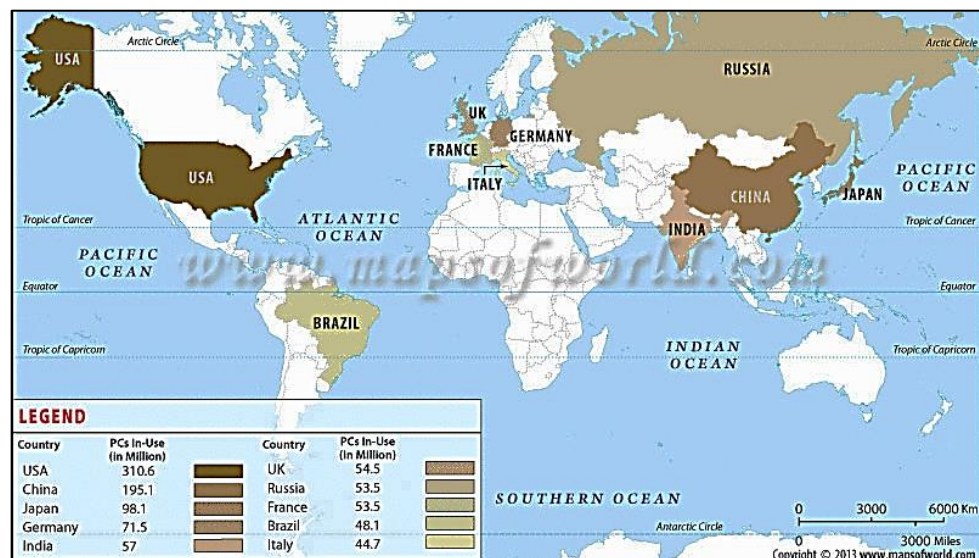
“La amplitud de las ciencias de la computación abarca todo. Es un campo interdisciplinario, con raíces en las matemáticas y la ingeniería y las aplicaciones en muchas áreas diversas.”², el hecho de relacionar las ciencias

² *New Jersey's Science & Technology University*. <<http://catalog.njit.edu>> [Consulta: 14 de mayo de 2014].

de la computación para resolver problemas en muchos aspectos de la vida incluyendo entretenimiento de alta definición como el renderizado de películas como *Toy Story* de Pixar, gráficos y secuencias de *Transformers*, sistemas de control de misión crítica como fábricas, robots, aviones, viajes espaciales, sistemas de información como bancos, *websites*, base de datos médicos, sistemas gubernamentales y herramientas de investigación como simuladores de terremotos, diseño de medicina, bases de datos de la astronomía. Las ciencias de la computación se pueden implementar en cualquier campo o área.

La programación es solo un aspecto de la informática. Los informáticos trabajan para resolver los problemas multifacéticos. Las ciencias de la computación ofrecen una excelente formación en la resolución de problemas y el pensamiento lógico, que son habilidades importantes para el empleo y la investigación.

Figura 1. Los 10 países con más computadoras



Fuente: *mapa del mundo*. <www.mapsofworld.com>. [Consulta: 14 de mayo de 2014].

Las ciencias de la computación han tenido un gran impacto en cómo se vive, piensa y actúa. En 1943, Thomas J. Watson, presidente de IBM declaró: “Creo que en el mundo hay mercado para quizás cinco ordenadores”³. Según el sitio web www.gartner.com y www.mapsofworld.com se presentan algunas estadísticas: “1.1 billones de computadoras en el 2012 y se estima que para el 2016 se tendrán 2 billones de computadoras”⁴.

En la actualidad es difícil caer en la subestimación del señor Watson y la tecnología, el potencial que existe en las ciencias de la computación en nuestra sociedad es como una gran ola de mar que cualquier surfista desea tener. La edad de oro de la informática y de los informáticos apenas ha comenzado. En muchos hogares el poseer una computadora es casi tan normal como tener un televisor.

En una entrevista realizada por Robert X. Cringely a Steve Jobs, este afirma: “Todos en este país deberían saber cómo programar una computadora, porque te enseña a pensar”⁵, programar una computadora es hablar con la computadora en su propio lenguaje y decirle qué hacer; la resolución de problemas puede hacerse de distintas maneras; todas las personas pueden desarrollar un pensamiento computacional distinto a otra; esto implica que hay muchas maneras de resolver problemas. Jobs fue visionario en muchos sentidos, influyó en muchas personas con sus palabras, su tiempo, dedicación y trabajo.

En un país como los Estados Unidos de América se tiene proyectado que para el 2020 habrá un millón más trabajos que estudiantes de informática y de

³ WATSON, Thomas John. <www.wikipedia.org>. [Consulta: 4 de junio de 2014].

⁴ *Technology Research | Gartner Inc.* <www.gartner.com>. [Consulta: 4 de junio de 2014].

⁵ (Jobs, 1995) The Lost Interview. (R. X. Cringely, Entrevistador).

las ciencias de computación. Las ciencias de la computación a nivel de título universitario y programación de computadoras es una de las carreras mejor remuneradas y está creciendo a 2 veces del promedio nacional, menos de 2,4 % de los estudiantes universitarios se gradúan con una licenciatura en ciencias de la computación.

En 2012, de 3,6 millones de exámenes, menos de 3,000 afroamericanos e hispanos tomaron el examen de colocación avanzada en ciencias de la computación de la escuela secundaria (preuniversitario), mientras que el 57 % de las licenciaturas son aprobados por mujeres, solo el 12 % de las titulaciones universitarias en ciencias de la computación se otorgan a las mujeres. Estos números son el reflejo de que 9 de cada 10 escuelas ni siquiera se ofrecen clases de programación de computadoras⁶.

Las ciencias de la computación han impactado la sociedad en nuestra manera de vivir. Ahora no es necesario salir de casa o viajar grandes distancias para realizar una compra particular, ya que por medio de las herramientas como Amazon, eBay, MercadoLibre, los cuales brindan el servicio de comercio electrónico ofreciendo productos como música, software, videojuegos, ropa, electrónica, muebles, libros, entre otros, puede efectuarse cualquier compra. El ser humano actual, haciendo uso de las nuevas tecnologías, puede recibir en un solo día, una cantidad de información similar a la que recibían los individuos en el siglo XVII durante toda su vida.

En los Estados Unidos de América, la tecnología de la información representa más del 6 % del PIB, comparado con el 3,1 % de la industria automotriz. Además, ésta ha contribuido a una mayor eficiencia y productividad

⁶. *Infographic code.org*. Consulta: 20 de junio de 2014.

en las organizaciones, reduciendo la tasa inflacionaria hasta en un 30 % durante 1997⁷.

Israel, desde la década del 90 impuso políticas sobre la enseñanza rigurosa de la computación. Este país posee un programa sobre dicha enseñanza, que es considerado como el más riguroso existente en el mundo.

Nueva Zelanda, en sus reportes ha rediseñado sus curso bajo el lema “Programación y Ciencias de la Computación“. Se provee material y educación para profesores como principales herramientas de cambio.

Alemania, en el 2008 adoptó estándares que claramente distinguen la enseñanza de TIC y ciencias de la computación en los diseños curriculares, que comienza de forma temprana en la escuela secundaria.

India ofrece ciencias de la computación en escuelas secundarias, con asignaturas para chicos de 12, 15 y 17 años. Para todos los niveles presentan distintos temas como redes, programación, arquitectura de computadoras, entre otros, además de manejo de software de oficina. Igualmente, continúan trabajando en un nuevo diseño curricular de ciencias de la computación.

Corea del Sur actualmente incluye algunos aspectos de ciencias de la computación, principalmente programación y algoritmos. Nuevos diseños curriculares están siendo propuestos, con menos énfasis en programación y más conceptos fundamentales de ciencias de la computación⁸.

⁷ Sadosky, Fundación. *Una propuesta para refundar la enseñanza de la computación en las escuelas Argentinas*, 2013.

⁸. *Ibíd.*

1.3. ¿Por qué es importante la enseñanza de ciencias de la computación?

Las ciencias de la computación en los próximos años producirán muchos empleos, es decir, generarán múltiples oportunidades profesionales, como se ha mencionado desde arte y entretenimiento, hasta telefónica celular y medicina, desde trabajadores de fábricas hasta dueños de pequeñas tiendas, todos deben entender sobre computación para ser competitivos en sus diferentes campos.⁹

En Corea del Sur, por ejemplo, la economía informal que se conoce aquí en Guatemala realiza sus transacciones por medio de POS, las tiendas y puestos de venta de productos en la calle, en su mayoría maneja un sistema de cómputo para llevar el registro de inventarios, proveedores, ventas y ganancias.

Las ciencias de la computación proveen un factor importante como la innovación en los negocios, los profesionales de las ciencias de la computación ocupan la mayoría de su tiempo trabajando con expertos del negocio, diseñando y construyendo sistemas de cómputo para muchos aspectos de nuestra sociedad como las artes, medicina e investigación científica, es decir, se apoya y se relaciona con otras ciencias. Aquellos que han sabido integrar, adaptar y utilizar estas nuevas tecnologías se están beneficiando de lo que tales herramientas pueden lograr.

Las ciencias de la computación son importantes ya que acrecientan el pensamiento humano. Enseña a solucionar problemas, cada necesidad es importante resolverla de diversas perspectivas, con la finalidad de obtener la

⁹. Sadosky, Fundación. *Una propuesta para refundar la enseñanza de la computación en las escuelas Argentinas*. 2013.

mejor solución. Alex Osborn, autor del *Brainstorming* (lluvia de ideas) dijo: “Todo el mundo posee cierta capacidad creativa, pero la mayoría de la gente no ha aprendido a utilizarla”¹⁰. “Una persona creativa es capaz de demostrar mayor originalidad que otras. La originalidad significa, en este contexto, la aptitud para dar respuestas insólitas a las preguntas, o soluciones atípicas a problemas específicos. La persona creativa es la que da vida a cosas nuevas, la que ve las cosas de manera original”.¹¹

Las ciencias de la computación permiten que el alumno sea el protagonista de su propio aprendizaje, a diferencia de una clase tradicional en la que el maestro transmite conocimiento y los estudiantes escuchan la lección. Y es por medio de esta metodología en que las ciencias de la computación dan la libertad al estudiante a abrir su horizonte y resolver el problema de muchas maneras.

Las ciencias de la computación son tan importantes como las matemáticas, ciencias sociales y ciencias naturales, ya que además de acrecentar el pensamiento computacional, ayuda a facilitar el trabajo y la persona que domine la tecnología tendrá un gran poder en sus manos.

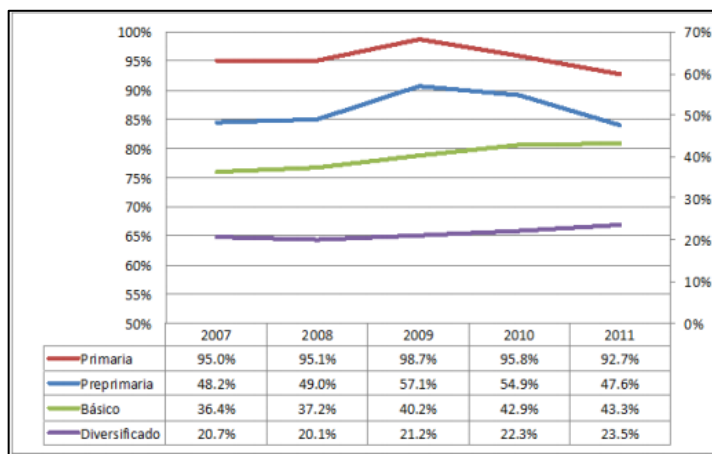
1.4. Guatemala, la educación y las ciencias de la computación

Se necesita adquirir recursos didácticos para la enseñanza de la informática y ciencias de la computación, con una sólida y actualizada formación científico-pedagógica y con una visión integrada de las ciencias, que facilita el trabajo interdisciplinario.

¹⁰ Napoleon Hill And W. Clement Stone, 1960.

¹¹ CORTESE, Abel. *Técnicas de estudio*. 140 p.

Figura 2. **Tasa educativa, por nivel educativo, por año**



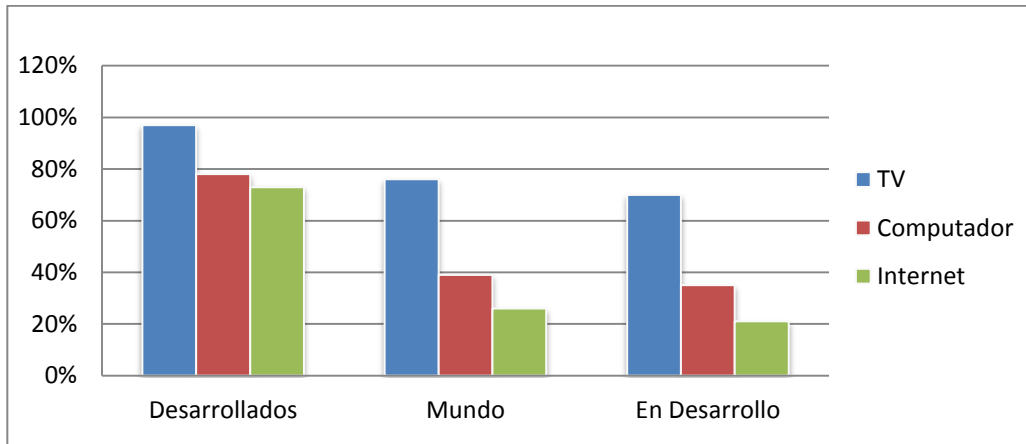
Fuente: Ministerio de Educación. *Informe de transición*. 2011.

Las estadísticas del país indican que para el 2012 la tasa de analfabetismo fue de 16,62 %, la tasa neta de escolaridad primaria fue de 89,10 %, la tasa de promoción fue de 68,5 %, la tasa de repitencia fue de 24,3 %, la tasa neta de matriculación de escuela secundaria fue de 46 %, la tasa de diversificado fue de 23 %. El índice de desarrollo humano es del 0,58, el cual evalúa los parámetros de vida larga y saludable, educación y nivel de vida digno.¹²

Las estadísticas brindadas por el Ministerio de Educación permiten ver la realidad del país, si bien es cierto en Guatemala existen barreras para la educación tales como la política, la corrupción, el presupuesto para el Ministerio de Educación, organizaciones sindicalistas, más de 20 idiomas mayas, la falta de infraestructura para llegar a las aldeas más remotas del país, el bajo nivel de calidad educativa, y la más dañina, el desinterés por parte de la misma sociedad guatemalteca; esto no implica que no se pueda cambiar.

¹² Ministerio de Educación.

Figura 3. Hogares con TV, computador e internet, 2012



Fuente: base de datos de la UIT sobre indicadores mundiales de las telecomunicaciones/TIC.

Con el pasar de los años se ha mejorado en la cantidad de alfabetismo, se ha disminuido la tasa de deserción y la tasa de repitencia a un ritmo bajo, pero desde la incursión de las telecomunicaciones y de las computadoras a la sociedad, un país en vías de desarrollo, se ha notado un cambio en las regiones donde la tecnología tiene presencia; la tendencia es que eleva el nivel de índice de desarrollo humano. Año tras año se va incrementando el porcentaje de población guatemalteca que posee un computador en su hogar.¹³

Actualmente la gran mayoría de universidades carreras o licenciaturas relacionadas con las ciencias de la computación, algunos colegios privados también han incursionado en el tema de las ciencias de la computación; la gran mayoría posicionados en el área urbana. Como se ha mencionado anteriormente, la infraestructura y la tecnología son recursos que ayudan al desarrollo, al no tenerlas, el avance al desarrollo se queda estancado o con un crecimiento lento.

¹³ Unión Internacional de Telecomunicaciones, 2013.

1.5. Ciencias de la computación en las escuelas públicas de Guatemala

La problemática de la educación en Guatemala se puede visualizar desde varias perspectivas; una de ellas es de que, se espera que el Gobierno sea el que se encargue de todo lo relacionado con la educación, ya que es quien rige las normas y procedimientos, quien recauda los impuestos, quien distribuye el presupuesto al Ministerio de Educación. Es el Gobierno quien debe de hacerse responsable de la educación en todo el país, quien está obligado a erradicar el analfabetismo.

Si se orienta la atención a este tipo de pensamiento realmente no se avanzará mucho en los próximos años, realmente se necesita mucho más que eso, John F. Kennedy expresidente de los Estados Unidos de América en su discurso inicial del 20 de enero de 1961 dijo: “No preguntes qué es lo que tu país puede hacer por ti; pregunta qué es lo que tú puedes hacer por tu país”¹⁴.

Realmente si se quiere ver una Guatemala desarrollada, mejorando la tasa de alfabetismo y la calidad de educación, entonces la sociedad guatemalteca se tiene que involucrar teniendo una actitud de servicio, brindando ayuda a los que la necesitan. Muchas personas dicen “yo no tengo el talento para ayudar”, quizás no tenga el talento pero cuenta con recursos para ayudar, otras dicen: “yo quisiera ayudar pero no tengo recursos”, quizás no se tenga recursos pero tiene el tiempo para ayudar, uno puede ser generoso con el tiempo, el talento o el recurso para ayudar a las personas, la pregunta sería entonces ¿estoy dispuesto a ayudar?

¹⁴ KENNEDY John F. *Presidential Library and Museum* - Columbia Point, Boston MA, 2014.

En Guatemala han surgido entidades no gubernamentales en pro del desarrollo tales como “Fundación Sergio Paiz Andrade”¹⁵ con el programa Escuelas Digitales, el cual tiene como objetivo mejorar la infraestructura básica de las escuelas donando laboratorios de cómputo, “Edulibre”¹⁶ cuenta con la visión de brindar una oportunidad a los niños a tener acceso a una educación a través de la tecnología de la información, de igual forma con la donación de computadoras y software instalado en las mismas, que ayuden a los estudiantes.

Muchas de estas organizaciones están enfocadas en brindarles la infraestructura necesaria; asimismo cuentan con un pequeño programa educativo, pero en muchas ocasiones sin darles el seguimiento correspondiente; es por ello que teniendo la infraestructura necesaria es más fácil poder implementar un programa que aumente la calidad de educación y que sea autosostenible como este EPS.

Puede empezarse a realizar un cambio, como ciudadanos, todos pueden involucrarse en ayudar a estas organizaciones no gubernamentales con recurso (donaciones de equipo que ya no se utiliza) y tiempo (actividades de mantenimiento de computadoras como lo hace Edulibre).

Como universitario se puede brindar tiempo para enseñar a estos niños y jóvenes y lo más importante influir y sembrar en ellos el deseo optar por una carrera universitaria. Como estudiante de la Escuela de Ciencias y Sistemas de la Facultad de Ingeniería de la Universidad de San Carlos de Guatemala, se puede dedicar tiempo, recurso y talento compartiendo con las escuelas la visión de la importancia de las ciencias de la computación, dando cursos de ciencias

¹⁵ Fundación Sergio Paiz Andrade, 2014.

¹⁶ Edulibre-Software y Educación, 2014.

de la computación a los estudiantes de la escuela y capacitando a su personal para que el programa sea auto sostenible.

1.6. Implementación de las ciencias de la computación como EPS

Implementación del programa de capacitación y desarrollo en ciencias de la computación utilizando el modelo de enseñanza “después de la escuela”, en la Escuela Oficial No. 53 David Vela, es el ejercicio profesional supervisado que consistió en capacitar a estudiantes de quinto y sexto primaria, en total 118; desarrollando en ellos el pensamiento computacional, dándoles así la oportunidad de comprender las ciencias de la computación, también se capacitó al maestro de computación; para que el proyecto sea autosostenible de manera que en los próximos años se pueda replicar esta enseñanza a las nuevas generaciones de dicha escuela.

Los niños son el futuro de la nación, son fundamentales puesto que el futuro del país les pertenece; el país de Guatemala puede morir o vivir según la educación y las oportunidades que se les brinde. Los niños tienen la capacidad de absorber el conocimiento rápidamente; se vive en la época de la información, los niños manejan los celulares como un usuario profesional, se adaptan de una manera increíble a la tecnología, pueden ir a un café internet y conectarse al mundo exterior.

La Unión Internacional de Telecomunicaciones (ITU) (International Telecommunication Union, por su siglas en Inglés) en el informe *Medición de la Sociedad de la Información* del 2013 llama a esta generación "nativos digitales", es decir, jóvenes nacidos en la era digital que crecen utilizando tecnologías de la información y la comunicación en su vida cotidiana.

Si bien no se ha llegado a un consenso en las publicaciones sobre el impacto exacto que las TIC tienen en los jóvenes, se conviene en general en que los medios digitales están cambiando la manera de aprender, jugar, alternar y participar en la vida civil. Los nativos digitales son la población de jóvenes conectados, de 15 a 24 años de edad, con cinco o más años de experiencia en línea. “En el informe de la ITU indica que en Guatemala existen 528,839 nativos digitales, 3,5 % de la población total, 17,2 % de la población joven total para el 2012 y 2013”¹⁷.

Esto quiere decir que si a los nativos digitales se les brinda oportunidades, el país se desarrollará de una manera impresionante; hoy en día existen herramientas como Scratch, Alice, Kids with Ruby, Kodu, Hackety Hack, Robomind y Tynker, herramientas que ayudan a aprender a programar y que son fáciles de utilizar. Scratch es una herramienta que ayudó a cumplir los objetivos en la ejecución del EPS, permite poner en práctica los conceptos de secuencias, ciclos, eventos, paralelismo, condicionales, operadores y manejo de variables.

Nada está perdido, mucha gente durante la ejecución del EPS dijo “¿cómo se va a cambiar a una nación si solamente dos niños se interesan en ese tipo de educación?”. Quizás para muchas personas hacer la diferencia implica cambios rápidos; se vive en la época del microondas, la comida rápida, los platos desechables, todo se quiere rápido, al instante.

La respuesta es la siguiente: “Si de un total de 120 niños se logra impactar en la vida de al menos dos niños, entonces se da por satisfecha la labor, el esfuerzo, la paciencia y la dedicación, ya que estos niños no tenían ningún tipo

¹⁷ International Telecommunication Union ITU Medición de la Sociedad de la Información , 2013.

de educación de ciencias de la computación y ahora si la tienen, esto quiere decir que estos dos niños pueden tener mejores oportunidades, inclusive pueden replicar sus conocimientos a los demás siendo generosos, dedicando su tiempo para que otros puedan tener las mismas oportunidades”, realmente vale la pena dedicar el tiempo a las personas que lo necesitan, e impartir conocimientos a la nueva generación, sin esperar recibir nada a cambio.

Tras finalizar este EPS, queda demostrada, la factibilidad de implementar las ciencias de la computación en escuelas públicas del país, con el apoyo de la Universidad de San Carlos de Guatemala y la Escuela de Ciencias y Sistemas, por medio de proyectos de EPS de seis meses, y así producir un impacto en la sociedad guatemalteca.

1.7. Duración del EPS

El EPS tuvo una duración de 6 meses, empezando en marzo y finalizando en septiembre del 2014, ajustando el calendario escolar y sus distintas actividades con el calendario de las clases del EPS.

1.8. Atribuciones en el desempeño del EPS

Durante el desarrollo del curso impartido, el epesista es responsable de:

- Transmitir la visión del proyecto, a los directores, maestro y estudiantes involucrados.
- Impartir clase para exponer el contenido del curso basado en el material “Programación con Scratch” cuaderno de trabajo para estudiantes; grados 3º a 6º de Juan Carlos López Garcia, cuarta edición.
- Elaborar enunciados de problemas.

- Realizar y calificar ejercicios.
- Realizar y calificar una evaluación final.

2. CURSO DE SCRATCH

El contenido del curso se basó en el documento *Programación con Scratch* del autor Juan Carlos López García, un cuaderno de trabajo dirigido a estudiantes de grados 3° a 6°, cuarta edición (4.01, agosto 2011) de la fundación Gabriel Piedrahita Uribe que otorga permiso para utilizar el documento bajo la licencia Creative Commons *Reconocimiento-NoComercial-CompartirIgual 3.0 Genérica*¹⁸.

- La licencia permite realizar lo siguiente:
 - Compartir: copiar, distribuir y comunicar públicamente el cuaderno de trabajo.
 - Remezclar: hacer obras derivadas.
- Bajo las siguientes condiciones:
 - Reconocimiento: reconocer los créditos de la obra mencionando al autor, a Eduteka y a la Fundación Gabriel Piedrahita Uribe.
 - No comercial: no utilizar la obra para fines comerciales.
 - Compartir bajo la misma licencia: si se altera o transforma la obra, o se genera una obra derivada, distribuir la nueva obra generada bajo una licencia idéntica.

¹⁸ Commons, Licencia Creative, s.f.

“Las segunda, tercera y cuarta edición del cuaderno de trabajo se elaboraron gracias al apoyo de Motorola Solutions Foundation y la gestión de la ONG Give to Colombia”.¹⁹

2.1. ¿Qué es Scratch?

“Es un entorno de aprendizaje de lenguaje de programación, que permite a las personas que no tengan conocimientos de programación, obtener resultados sin tener que aprender a escribir de manera sintácticamente correcta primero; esto facilita el aprendizaje autónomo y ayuda a los jóvenes a aprender a pensar creativamente, razonar sistemáticamente, y trabajar colaborativamente habilidades esenciales para la vida en el siglo XXI”.²⁰

“Fue desarrollado como un proyecto del Grupo Lifelong Kindergarten del Laboratorio de Medios del MIT (Instituto Tecnológico de Massachusetts) dirigido por Mitchel Resnick; este software apareció por primera vez en el verano del año 2007. Está diseñado especialmente para edades entre los 8 y 16 años, pero es usado por personas de todas las edades. Millones de personas están creando proyectos en Scratch en una amplia variedad de entornos, incluyendo hogares, escuelas, museos, bibliotecas y centros comunitarios”.²¹

El nombre de Scratch se deriva de la técnica de “*scratching*” usada por los DJs, utilizada para producir sonidos característicos a través del movimiento de un disco de vinilo hacia delante y hacia detrás sobre un tocadisco, y por medio del cual generan música uniendo las piezas musicales. La similitud con el “*scratching*” musical es la fácil reutilización de piezas: en Scratch todos los

¹⁹.LÓPEZ GARCIA, Juan Carlos. 2011 *Programación con Scratch*. Colombia.: Fundación Gabriel Piedrahita Uribe Motorola Solutions Foundation Give to Colombia.

²⁰.Scratch | Lenguaje de Programacion, 2014.

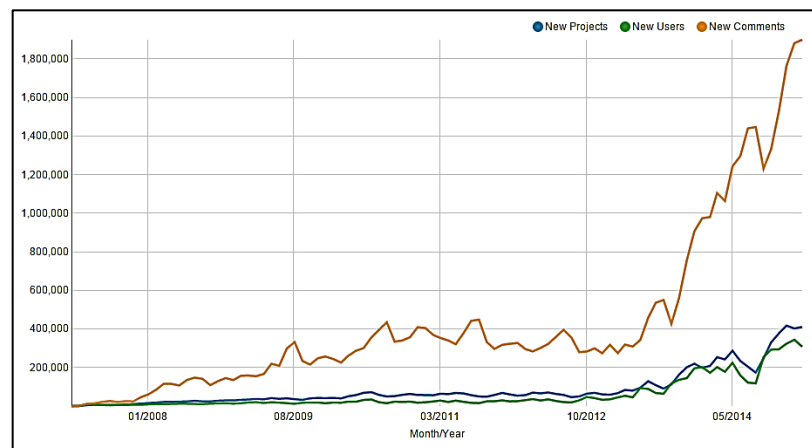
²¹.Acerca de Scratch | scratch.mit.edu, 2014.

objetos, gráficos, sonidos y secuencias de comandos pueden ser fácilmente importados a un nuevo programa y combinados de diversas maneras, permitiendo a los principiantes a conseguir resultados rápidos y estar motivados para intentar más.

2.2. Estadísticas de Scratch

La comunidad de Scratch está aumentando significativamente; ahora existen más de 8 millones de proyectos, más de 5 millones de usuarios registrados, más de 42 millones comentarios publicados, y más de 700 mil estudios creados; esto quiere decir que la tendencia está creciendo exponencialmente gracias a la difusión del programa a todo el mundo.

Figura 4. **Tendencias mensuales de actividad**



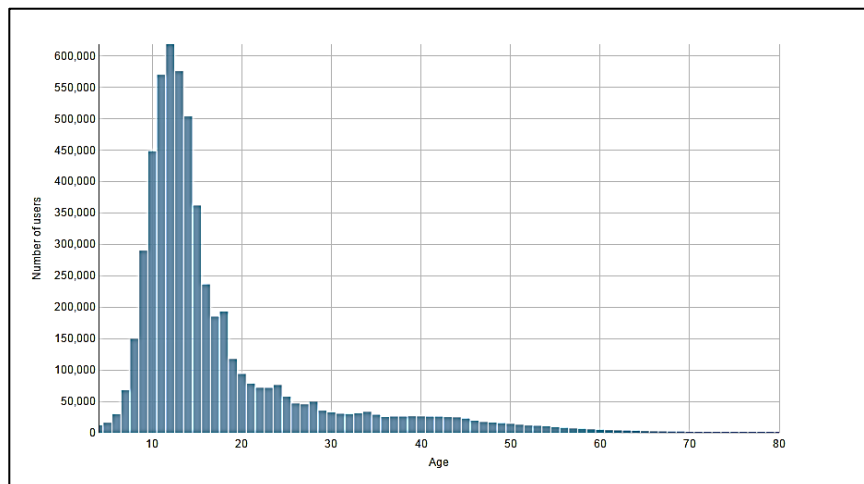
Fuente: *Tendencias mensuales de actividad*. <http://scratch.mit.edu/statistics/>. Consulta: 14 de junio de 2014.

La tendencia mensual de Scratch respecto de nuevos proyectos, usuarios y comentarios, han tenido un crecimiento que se había mantenido normal desde

el 2007 hasta el 2012. En el 2012 fue cuando tuvo un crecimiento de gran magnitud; este crecimiento se ha producido debido a que la organización code.org ha promocionado un movimiento llamado “*One hour of code*” (una hora de codificación), el cual consiste en que todas las personas puedan programar durante una hora diariamente, realizando ejercicios y tareas que han publicado en su portal. Esto ha despertado el interés de utilizar Scratch y otras herramientas en todas partes del mundo, ya que puede utilizarse en línea como fuera de línea.

Según el registro del uso de Scratch, la distribución por edades de los nuevos usuarios de este lenguaje tiene una representación significativa entre las edades de 8 a 16 años. Los niños y adolescentes a menudo ponen en práctica la creatividad, y esto se acopla muy bien en el ámbito educativo para aprender jugando.

Figura 5. **Distribución por edades de los nuevos scratchers**

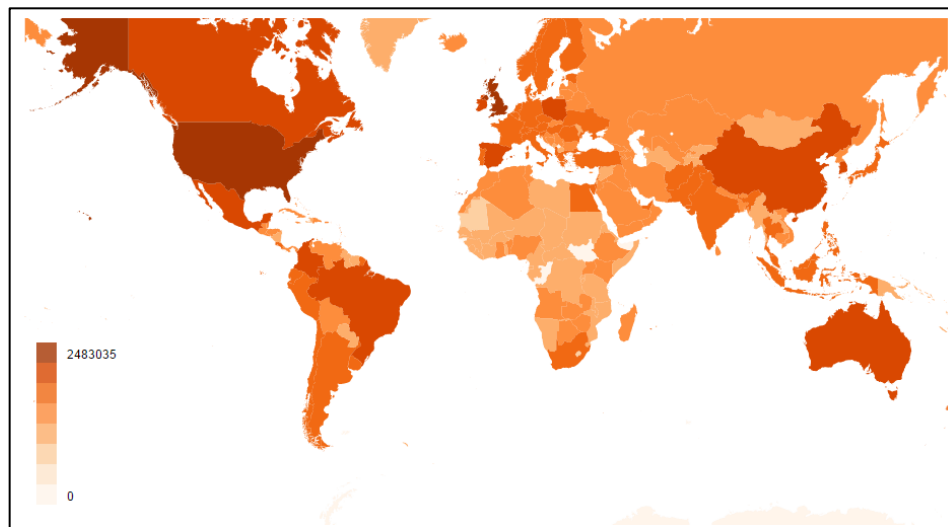


Fuente: *Tendencias mensuales de actividad*. <http://scratch.mit.edu/statistics/>. Consulta: 14 de junio de 2014.

Estas edades según el modelo educativo propuesto, comprenden los grados de cuarto, quinto y sexto primaria, el ciclo básico y diversificado; esto podría ayudar a promover y ejecutar un plan para la inserción de programas para el aprendizaje de las ciencias de la computación por medio de Scratch en la escuelas públicas y de esta manera poder influir en estos niños y adolescentes.

Geográficamente las barreras se rompen por medio de la tecnología y la internet, la gran mayoría de países de todo el mundo, por medio de escuelas, colegios y la entidad educativa encargada de regular el pénsum estudiantil en cada país, han introducido programas para la enseñanza de programación por medio de la herramienta de Scratch.

Figura 6. **Scratchers alrededor del mundo**

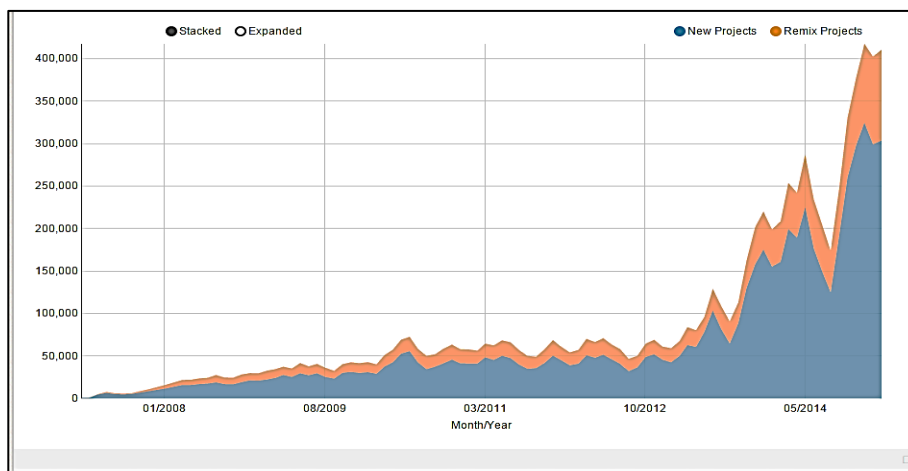


Fuente: *Tendencias mensuales de actividad*. <http://scratch.mit.edu/statistics/>. Consulta: 14 de junio de 2014.

Scratch ha tenido un gran impacto en los Estados Unidos de América, Inglaterra, Australia, España y China, ya que han empezado la enseñanza de la educación de las ciencias de la computación y programación a una edad muy temprana; esto ha impulsado a las personas a buscar carreras y trabajos relacionados con la tecnología. Además permite a las personas pensar de una forma diferente, buscando varias alternativas para la resolución de problemas; esto también influye en su diario vivir.

El interés que las personas están tomando respecto de la tecnología se ve reflejado en los proyectos compartidos de Scratch. Estos son fundamentales ya que muestran que ha culminado un proyecto y lo suben a la web, estos proyectos pueden ser vistos por todas las personas; de esta manera se puede visualizar el resultado del proyecto como el código mismo; esto ayuda también a otros a intercambiar el conocimiento para poder resolver un problema, o llevar a cabo un proyecto similar.

Figura 7. **Proyecto compartidos mensualmente**



Fuente: *Tendencias mensuales de actividad*. <http://scratch.mit.edu/statistics/>. Consulta: 4 de julio de 2014.

Se ha formado una comunidad de Scratch donde se intercambian ideas y conocimiento y con el pasar del tiempo se ha logrado establecer comunidades específicas por región que le dan seguimiento a los proyectos y a los avances de un grupo de personas o incluso a entidades educativas.

2.3. Descripción del curso de Scratch

El objetivo general de este curso es proporcionar el conocimiento de ciencias de la computación a nivel primario, dirigido fundamentalmente a niños y niñas de quinto y sexto primaria de la Escuela David Vela, como parte del EPS.

El curso tuvo una duración de 6 meses que se divide en 3 partes. En la primera parte, con una semana de duración, se introdujo al estudiante el conocimiento y la diferencia entre computación y ciencias de la computación, la aplicación de cada una de ellas en la vida real, la tendencia mundial en cuanto a la educación en ciencias de la computación y las diferentes oportunidades y beneficios que traen consigo el aprender ciencias de la computación. En la segunda parte, con tres semanas de duración, se capacitó a los estudiantes en el uso de la herramienta de trabajo “Scratch”, su entorno de trabajo, la funcionalidad de cada uno de los bloques, tales como el bloque de control, apariencia, sensores, sonido, operadores, lápiz, variables.

En la tercera parte con una duración de 5 meses, consistió en el uso del cuaderno de programación con Scratch, el cual trae consigo ejercicios de adiestramiento, laboratorios y tareas para trabajar en clase y en casa, poniendo en práctica los conocimientos obtenidos de una manera sencilla y práctica. También cuenta con un apartado llamado desenchufado, en el cual se hacen

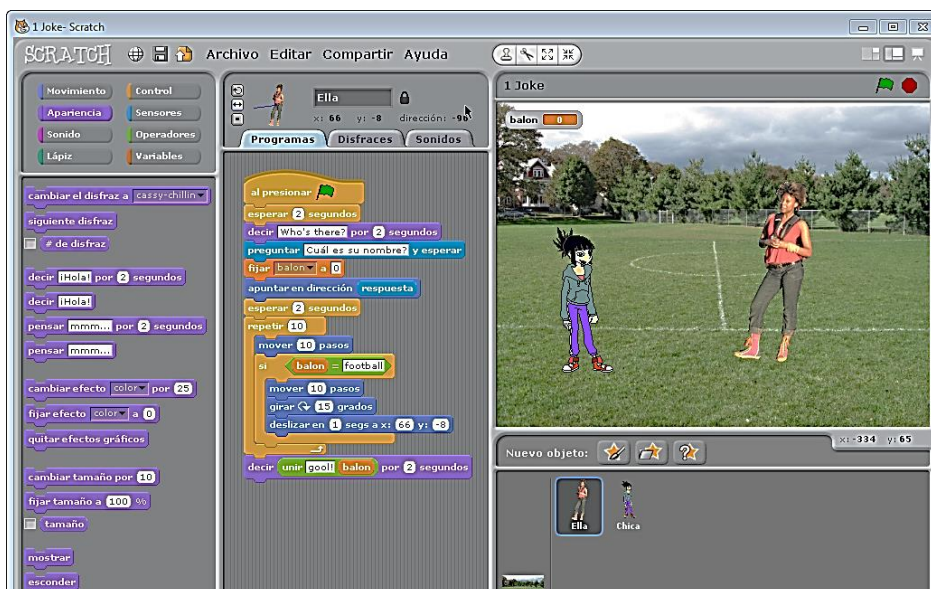
ejercicios y tareas sin un computador y les ayuda a comprender conceptos básicos mediante el uso de estos ejercicios.

El curso también cuenta con evaluaciones periódicas que miden el avance de los alumnos, el cual es un indicador de la comprensión de los alumnos y la transmisión del conocimiento de parte del maestro.

2.4. Entorno de trabajo de la herramienta Scratch

Scratch permite poner en práctica lo relacionado con el conocimiento obtenido del cuaderno “Programación con Scratch”; el entorno de trabajo es fácil de usar ya que permite crear programas de una manera sencilla y agradable.

Figura 8. Entorno de trabajo de Scratch



Fuente: elaboración propia, con programa Re-Digital.

“La gramática de Scratch se basa en un conjunto de bloques gráficos de programación que pueden ensamblarse uno a uno para crear programas, tal como los bloques de Lego lo hacen para crear figuras, casas, edificios, entre otros, estos conectores permiten ensamblar y crear cualquier cosa y hacer que el programa funcione, es simple de utilizar ya que solamente se arrastra el bloque al área de trabajo”.²²

Figura 9. Programas en bloques de Scratch



Fuente: elaboración propia, con programa Re-Digital.

2.5. Contenido del cuaderno “Programación con Scratch”

Se contó con la ayuda del material didáctico del cuaderno de “Programación con Scratch” para cada estudiante, el cual fue reproducido gracias a las donaciones económicas realizada por personas que tenían el conocimiento y la intención de ayudar a los estudiantes de la escuela y de este proyecto.

El cuaderno cuenta con una estructura definida que encamina al estudiante al aprendizaje gradual con dificultad creciente en los conceptos y la aplicación de cada uno de ellos para programar en Scratch, poniéndolos en práctica de distintas formas. Este cuenta con varios apartados que sirven como

²².LÓPEZ GARCIA, Juan Carlos. 2011. *Programación con Scratch*. Colombia.: Fundación Gabriel Piedrahita Uribe Motorola Solutions Foundation Give to Colombia.

guía para realizar distintas actividades que proponen nuevos retos con distintos propósitos, identificándolos con un sello o marca, que le indica al estudiante el propósito de lo que tiene que realizar en dicha actividad.

El cuaderno también cuenta con un apartado llamado “Metas al volante” donde se indica que el problema a resolver es muy grande, y este se divide en varias actividades o pasos a seguir para completar con la resolución del problema o reto; de esta forma las metas volantes permiten avanzar de manera ordenada hasta llegar al resultado final.

La actividad puede tener asociado un sello, el cual indica su propósito, es decir, que indica qué tipo de actividad va a realizarse tales como: imaginar, crear, jugar, compartir, entre otros.

2.6. Sellos y apartados del cuaderno “Programación con Scratch”

- Sello “Imagina”: este sello indica a los estudiantes que es el momento de poner en práctica su imaginación, relacionada con lo que se le pide que haga en Scratch, es decir, poner al estudiante a pensar en qué es lo que quieren llegar a hacer y cómo hacerlo, colocando en sus mentes los pasos a seguir para llegar al resultado de lo que dice el enunciado y colocarlos por escrito en el área que está en el cuaderno. Por ejemplo: “Imagina un diálogo corto entre dos personajes y escríbelo a continuación, indicando los segundos (seg) que dura el mensaje.”
- Sello “Crea”: este sello indica que es el momento que los estudiantes deben poner en práctica lo aprendido y utilicen los bloques de Scratch para crear un proyecto basado en las ideas de los estudiantes y que cumplan con los requerimientos dados por el maestro; este sello o

apartado va muy relacionado al de “Imagina” ya que primeramente deben de imaginarlo y luego crearlo. Por ejemplo: “Crea en Scratch una representación del diálogo que acabas de elaborar. Los objetos/personajes deben dialogar, es decir, hablar uno a la vez, respetando tiempos prudenciales para mostrar cada intervención en el diálogo.”

- Sello “Juega”: este sello indica que es el momento de jugar con las ideas y creaciones ya realizadas, es decir, que se deja libertad a los estudiantes a modificar los proyectos, agregando la funcionalidad que ellos crean conveniente en su propio proyecto o en el de su compañero. Por ejemplo: “Llegó el momento de jugar con los proyectos de tus compañeros. Deja abierto tu proyecto en el computador y te paras para ubicarte en el computador que tu maestro(a) te indique. Juega con el proyecto de tu compañero(a)”.
- Sello “Comparte”: este sello indica que es el momento de compartir las ideas y creaciones con otros estudiantes de la comunidad de Scratch y permite mostrar lo que se ha hecho, al tiempo que posibilita que los demás puedan dar nuevas ideas que se apliquen al proyecto, ya que pueden cumplir con el objetivo o resultado esperado haciéndolo de distinta forma. Por ejemplo: “comparte este proyecto con la comunidad de usuarios alrededor del mundo. No olvides agregar las notas al proyecto (archivo/notas sobre el proyecto). Aquí puedes explicar de qué se trata tu proyecto e incluir instrucciones para utilizarlo. Recuerda que cuando compartes un proyecto, es muy posible que otras personas se interesen en él, por lo tanto, debes contarles a esas personas cómo usarlo”.

- Sello “Reflexiona”: este sello indica que es el momento en que el estudiante debe reflexionar sobre las experiencias con Scratch, verificar si la programación cumple con el resultado esperado, ¿Qué se podría agregar al programa para cumplir con otros objetivos? ¿Qué pasaría si utilizara otro bloque? de manera que lleve a los estudiantes a imaginar nuevas ideas, nuevos proyectos y nuevas formas de aplicar lo aprendido. Por ejemplo: “Reflexiona sobre esta actividad: ¿Qué fue lo que más te gustó? ¿Qué fue lo más fácil? ¿Qué fue lo más difícil? ¿Qué te gustó de compartir este proyecto con tus compañeros? ¿Qué aprendiste?”
- Apartado “Problemas inesperados”: este apartado indica que siempre hay formas de mejorar los proyectos, de hacerlos más funcionales, con lo cual es tiempo que se experimenten y mejoren los proyectos; a lo ya realizado se le pide que se le agregue funcionalidad para cumplir con otro objetivo, este apartado va de la mano con los sellos “Imagina”, “Crea”, “Juega” y “Comparte”.
- Apartado “Análisis de problemas”: este apartado es muy importante, ayuda al estudiante a ver cada proyecto Scratch como un problema a solucionar; este apartado es donde el estudiante pone por escrito las posibles soluciones que podrían ayudar a resolver el problema, plasmando las ideas resultantes y los pasos a seguir. Por ejemplo: “Participa activamente en la lluvia de ideas dirigida por tu maestro(a) sobre el análisis del problema. Apunta aquí las ideas interesantes”.
- Apartado “Diseño”: este apartado le indica al estudiante que es momento de elaborar un diseño ideado un plan para elaborar un proyecto, y se debe de traducir al formato que “entiende” Scratch (escenario, fondos, objetos, disfraces, sonidos, variables, entre otros), es decir, que se

coloca en el apartado lo que debe realizar para cumplir con el objetivo el cual lleve a la solución correcta. Por ejemplo: “Ahora que completaste la actividad, utiliza tanto la información de la plantilla de análisis, que llenaste en la meta volante 1, como las acciones que llevaste a cabo durante la desarrollo del proyecto Scratch “Carrera de autos” para llenar las siguientes plantillas de diseño.”

- Apartado “Ejercicios”: este apartado indica al estudiante que es el momento de hacer un ejercicio de aprestamiento y debe realizarlo para estar mejor preparado al momento de realizar la actividad correspondiente. Los ejercicios de refuerzo sirven para consolidar los aprendizajes alcanzados durante la actividad. Por ejemplo: “Discutamos en parejas el ejemplo de la bombilla y propongamos algunas mejoras. Luego, un voluntario debe pasar al tablero y escribir el algoritmo con la participación de todos.”
- Apartado “Prácticas”: este apartado indica al usuario que es el momento de realizar tareas cortas que pueden realizar en pocos minutos en una clase, Así se busca intercalar elementos lúdicos a lo largo del proceso de aprendizaje de Scratch. Ejemplo: “Elabora un proyecto Scratch en el que utilices el bloque “numero al azar entre __ y __”?”
- Apartado “Scratch desconectado”: indica que todos los ejercicios y prácticas en este apartado, se deben realizar sin utilizar un computador, es decir, que son actividades que se realizan en el cuaderno como un trabajo manual. Ejemplo: “¿Qué elementos incluirías en un aviso publicitario sobre Scratch? ¿Qué argumentos utilizarías para convencer a un estudiante de un grado inferior al tuyo para que utilice Scratch?”

- Apartado “Conceptos fundamentales”: son recuadros de color naranja en los cuales se encuentran definiciones de conceptos fundamentales en programación y formas de uso de estructuras de control, las cuales son muy importantes. Por ejemplo: “estructura repetitiva”. Los bloques repetitivos permiten ejecutar uno o varios bloques, un número determinado de veces o, indefinidamente, mientras se cumpla una condición. Esta estructura repetitiva ayuda a ahorrar tiempo y espacio en los programas de Scratch.”
- Apartado “Evaluación”: es una matriz con la cual se evalúa el trabajo del estudiante realizado con Scratch, evalúa los distintos aspectos como nombres significativos para las variables, funcionamiento del programa, manejo del editor de pinturas, manejo de objetos, animación básica, uso de bloques repetitivos, entre otros. Se utilizan para comparar lo que se espera que hagan los estudiantes y sea considerado como sobresaliente. Por ejemplo: “En la siguiente tabla encontrarás la forma cómo se va a evaluar tu trabajo, qué se considera sobresaliente y qué no es aceptable.”
- Apartado “Sugerencias”: este apartado presenta una serie de trucos a lo largo del cuaderno que proponen al estudiante probar nuevos comandos o formas diferentes de realizar alguna tarea. Por ejemplo: “Programa los objetos incluyendo el siguiente par de bloques para lograr que al presionar un objeto, este envíe un mensaje que indique qué fondo debe activarse.”
- Retos en familia: es un apartado de ejercicios retadores que los estudiantes deben realizar en compañía de sus padres o hermanos. Se debe prestar mucha atención al razonamiento que hacen los mayores

para llegar a la respuesta. Por ejemplo: “Pídeles a tus padres y hermanos mayores que te ayuden a resolver el siguiente problema: Debes cambiar de lugar solamente tres cerillos de forma que construyas tres cuadrados. No puedes romper ni sobreponer cerillos (fósforos).”

Con la explicación de cada uno de estos sellos y apartados, el estudiante utiliza el cuaderno con entusiasmo y decidido a aprender.

3. EVOLUCIÓN DEL CURSO

3.1. Metodología

Para el curso se contó con el cuaderno de programación que sirvió de guía para impartir las clases de forma verbal y visual; adicionalmente se tomó en cuenta para elaborar el cronograma inicial y así delimitar las clases por periodo.

También se contó con un proyector que permitió dar las clases de una forma dinámica. Las clases fueron impartidas según el cronograma inicial; también se capacitó al profesor de computación con el contenido del curso; varias clases fueron impartidas por el profesor de computación, para cumplir con uno de los objetivos.

3.1.1. Primeros pasos en Scratch

En este módulo se desarrolló el aprendizaje del entorno de trabajo de Scratch en cada una de sus secciones como el área de bloques, y de dibujo, el segmento de coordenadas y propiedades del objeto, la forma en que se arrastran los bloques al área de trabajo, en cómo unir los bloques uno seguido del otro; como si fueran piezas de lego para construir algo.

A los estudiantes se les enseñó el concepto de algoritmo, el cual permite describir claramente una serie de instrucciones para lograr un resultado, y se les dio varios ejemplos como cambiar una bombilla o freír un huevo con mantequilla; los estudiantes se dieron cuenta que para realizar esas tareas

había que detallarlas de una manera ordenada, y con los pasos necesarios, ya que el obviar algunos pasos o instrucciones, podría no obtenerse el resultado esperado.

Se les pidió a los estudiantes realizar varios ejercicios para describir detalladamente el orden y los pasos lógicos a seguir para comprar una revista, botar la basura, empaçar un regalo, construir un avión de papel y pescar. De esta forma se ejemplificó que los algoritmos se tienen que detallar, así como los pasos para cumplir con el objetivo.

Se estudió el concepto de los comentarios en programación, cómo es que los comentarios ayudan a identificar una porción de código y brindan información vital de lo que puede realizar esa porción de código de programación.

El ejercicio de aprestamiento es Aquarium, el cual permite visualizar lo que se puede llegar a hacer con Scratch; en este ejercicio los estudiantes pudieron determinar el comportamiento de cada objeto en pantalla; a los estudiantes se les pidió que cambiaran una porción de código y que visualizaran el resultado en pantalla, los estudiantes se encargaron de documentar en su cuaderno lo que pudieron observar con los cambios realizados.

Se realizó una práctica desconectada, en la cual tenían que indicar detalladamente los pasos a realizar para movilizarse de su escritorio hasta la puerta principal del salón; en cuanto más detalle pudo colocar los estudiantes, más se acercaban a la realidad para realizar la acción. De esta manera se les mostró que una computadora es tonta en sí, ya que se le debe decir detalladamente lo que tiene que hacer, a diferencia de los seres humanos que

sí comprenden conceptos e instrucciones; se dieron por sentado los pasos a seguir para llegar al resultado esperado.

3.1.2. Animando un paisaje

En este módulo se enseñó la opción que Scratch tiene para el editor de pinturas; en este editor el usuario puede crear pinturas, es decir, crear paisajes, objetos como un tren, un payaso, entre otros. Este editor de pinturas cuenta con las herramientas básicas como una brocha, la cual permite manejar con el *mouse* como si fuera un lápiz en un cuaderno, un borrador que permite borrar lo realizado en el editor, una cubeta para llenar de algún color un espacio definido, y la opción para pintar un rectángulo, elipse, línea y texto.

En el ejercicio de aprestamiento se les pidió a los estudiantes realizar un tren y un payaso para que practicaran con el editor de pinturas y así ellos podrían crear muchos objetos y paisajes. Adicionalmente se les pidió que crearan un paisaje, luego se les pidió que animaran el paisaje colocándole movimiento a cada uno de los objetos.

A los estudiantes se les enseñó el concepto de Identificadores; el nombra los objetos, variables y métodos en un programa es de gran ayuda, ya que se puede identificar de una manera clara y tomar una idea de lo que puede hacer un método, lo que logre almacenar una variable; al igual que cada persona tiene un nombre, cada elemento del programa tiene un identificador sobre el cual se puede trabajar.

A los estudiantes se les enseñó a crear bloques de programas con las instrucciones mover, girar, ir a, deslizar, repetir, por siempre, rebotar si está tocando un borde, cambiar disfraz, siguiente disfraz, al presionar. Para cada

instrucción se les explicó su funcionalidad, dando uno o más ejemplos de cómo utilizarlos, los parámetros que pueden llegar a tener, y el orden en el cual se deben poner.

A los estudiantes se les pidió realizar un ejercicio, que imaginaran cómo realizar una animación en el que un gato caminara cierta distancia en la pantalla, que lo hiciera 10 veces, y en otro programa que lo hiciera indefinidamente, los estudiantes pusieron a prueba sus conocimientos y los practicaron respecto de bloques repetir y por siempre; durante este ejercicio, se tuvieron varias actividades como metas al volante y reflexionar, para el cual se les pidió que agregaran la funcionalidad de cambiar de disfraces y que pudieran girar hacia una dirección durante cierto tiempo.

A los estudiantes se les pidió elaborar la matriz de valoración de la actividad anterior, calificando así los nombres significativos para los objetos, disfraces, escenarios, entre otros. La funcionalidad del programa, el manejo del editor de pinturas, animaciones básicas y el uso de los bloques repetitivos al realizar esta matriz de valoración, los estudiantes se pudieron dar cuenta de sus errores y corregirlos, también observaron que para completar un proyecto es necesario realizar las pruebas necesarias para verificar su correcta funcionalidad.

Para el ejercicio de refuerzo se les pidió crear un robot con figuras geométricas básicas (círculo, cuadrado y rectángulo) y luego colorearlo; en el cuaderno ellos tenían un apartado de identificación del problema, otro sobre las restricciones, y otro apartado para colocar los pasos a seguir para la elaboración del robot.

Otro ejercicio de refuerzo fue dibujar por medio de un conjunto de instrucciones de repetición y tipo de figura como caracol o laberinto para el cual únicamente tenían que utilizar los bloques bajar la piza, borrar, mover X pasos, girar X grados, presionar, ir a X, Y.

3.1.3. Diálogos animados

Para este módulo se enseñó una manera de interactuar con el usuario, por medio de mensajes, preguntas y respuestas, haciendo uso de una historieta grafica en el cual se cuenta con varias escenas de dos personas intercambiando una conversación. De esta manera se pudieron crear historietas animadas con Scratch, y poner a prueba su imaginación.

Se les enseñó el concepto del diálogo el cual es necesario entre las personas, de igual forma un programa puede interactuar con el usuario para manifestar sus ideas y pensamientos. Estos diálogos deben tener un orden lógico para entender el mensaje; si no tuviera un orden lógico sería un caos, ya que se interpretaría de una forma incoherente que no tiene sentido.

A los estudiantes se les pidió crear por medio de Scratch un diálogo entre una persona y un perro, una pequeña historia en el cual manifestaran un conversación entre estos dos personajes, utilizando la instrucción decir, pensar, esperar X segundos, entre otros. Esto fue muy importante manejarlo, ya que permitió a los estudiantes darse cuenta que la interacción con el usuario es fundamental y que debe tener un orden lógico.

Se les enseñó el concepto de estructuras repetitivas, el cual permite ejecutar uno o varios bloques, un número determinado de veces o,

indefinidamente, mientras se cumpla una condición. Estas estructuras repetitivas ayudan a ahorrar tiempo y espacio en los programas de Scratch.

Los estudiantes se pudieron dar cuenta de que es mucho más simple utilizar estructuras repetitivas, ya que acortan el número de instrucciones que el programa utiliza al ponerlo en práctica; también se dieron cuenta que se tardaban menos tiempo en realizar el programa, y si había que cambiar algo, era mucho más sencillo; se les enseñó que tenían que reconocer los patrones para utilizar las instrucciones repetitivas.

Se les dio a conocer el concepto de análisis de problemas; esto se consideró como la piedra angular, ya que el análisis permite formular el problema, identificar los resultados esperados, los datos disponibles, y las restricciones y definir los procesos necesarios para convertir dichos datos en la información requerida.

A los estudiantes les pareció algo de otro mundo, lo cual no les iba a servir para nada, pero conforme se fue desarrollando el curso, se dieron cuenta de la necesidad de seguir estos pasos para resolver los enunciados y problemas que se les dejaba. Comprendieron que realmente esto no solamente les ayuda a la programación con Scratch, sino que también les sirve para la vida misma, y también pudieron determinar que el proceso para la resolución del problema podía realizarse con varas alternativas.

Se les enseñó el concepto de lenguaje pseudocódigo, ya que para la solución del problema se puede expresar en varias maneras, y una de ellas es en pseudocódigo que consiste en representar las instrucciones por medio de palabras comunes y corrientes en forma de frases, lo cual permite un estándar para las instrucciones.

Se les pidió a los estudiantes realizar cálculos de suma, resta, multiplicación y división utilizando operadores y variables; esto permitió realizar programas con instrucciones de asignación de valores, desplegando así el resultado en pantalla, así también conocer más sobre los operadores y cómo utilizarlos.

Para la práctica se les pidió a los estudiantes crear una historia y que dentro de la misma incluyeran operadores matemáticos y desplegar los resultados, también se realizó la evaluación de la matriz de valoración de la actividad.

3.1.4. Carrera de autos

En este módulo se realizaron varios ejercicios de aprestamiento y se pusieron en práctica los conceptos que se habían aprendido hasta ese momento en el ejercicio de aprestamiento: medio día, tarde, noche, y reemplazar bandera verde; estos ejercicios ayudaron a reforzar, ya que constantemente utilizaban los bloques que aprendieron anteriormente.

3.1.5. Ampliando el juego Pong

En este módulo se les enseñó el juego de los bloques de azar, conectores lógicos y proposiciones condicionales, por medio de los ejercicios de refuerzo analizar bloques, analizar programas y dibujos en Scratch.

3.1.6. Tablas de multiplicar

En este módulo se les enseñó sobre el concepto de interactividad con el usuario, variables, diagramas de flujo y acumuladores, las sentencias de

decisión bloque de sí, no y los ejercicios de aprestamiento fueron los proposiciones, partes de la proposición, conectores lógicos y proposición condicional.

Los estudiantes pudieron realizar una calculadora que efectuaba cálculos de área y perímetro, dependiendo de la figura geométrica que el usuario seleccionaba, y también dependía de la información de los valores que el usuario ingresaba.

3.2. Dificultades encontradas

Las dificultades encontradas fueron las siguientes:

- Equipo de cómputo antiguo: la mayoría del equipo de cómputo es muy antiguo, para empezar dificultó la instalación del programa de Scratch; adicionalmente a esto, la velocidad con la que respondían algunas máquinas era muy lenta; esto perjudicaba al estudiante ya que no podía avanzar al ritmo de los demás estudiantes.
- Pocas computadoras: al iniciar el proyecto: se contaba con máquinas antiguas, algunas no servían, a otras les faltaba el disco duro, a otras la memoria ram; al inicio había un total de 14 computadoras funcionando; hubo que conseguir patrocinadores para completar las máquinas con disco duro y memoria ram; al final se logró tener 22 máquinas funcionando. Sin embargo no era suficiente, ya que había grados que tenían más de 25 estudiantes y por esa razón se dividía el grupo en dos, y para recibir el curso.

- No contar con una cañonera o medio visual: al no contar con un medio visual se dificultaba la parte de la enseñanza, ya que en el pizarrón es difícil transmitir la idea por escrito a lo práctico, también dificultaba al momento de que los estudiantes tuvieran dudas, ya que era difícil la movilización de lugar en lugar, lo cual requería tiempo y no se podía cumplir con el calendario establecido; se logró obtener la cañonera gracias a la donación de personas para la escuela.

3.3. Casos especiales

Durante una clase se les pidió realizar un ejercicio de meta volante con la instrucción “ir a X:___ Y:___” y la instrucción “girar ___ grados” el cual está en las páginas 29, 30 y 31 del cuaderno: en este caso los estudiantes realizaron la meta volante, pero no comprendieron realmente de qué se trataban las instrucciones, colocando números positivos o negativos, se cree que el libro asume que los estudiantes deben de tener conocimientos del plano cartesiano, sin embargo ellos no tenían tal conocimiento. La solución fue explicarles lo que es el plano cartesiano, y el concepto de grados para formar la circunferencia.

Suspensión de actividades en la escuela: en ocasiones se suspendían las actividades escolares ya sea por paro general en el país, que la escuela no contaba con agua potable, por lo tanto no se tenían habilitados los servicios sanitarios, actividades especiales en la escuela como entrega de notas, celebraciones, prácticas, entre otros.

3.4. Resultados

Los resultados fueron sorprendentes, los estudiantes lograron adaptarse a la metodología de estudio y comprender cada una de las instrucciones que

tiene Scratch, y lo más importante es que lograron desarrollar el pensamiento computacional. Cuantos más ejercicios realizaban, más aprendían, ya que podían ver las alternativas para resolver los problemas y de distintas formas, llegando al mismo objetivo; podían resolver los problemas compartiendo y discutiendo con sus compañeros, realizándolo en menor tiempo.

En el grado de sexto sección A, se contó con la participación de 32 estudiantes; de estos 6 resultaron con calificaciones excelentes (96 % - 100 %), 15 estudiantes resultaron con calificaciones promedio alto (80 % - 95 %), 7 estudiantes resultaron con calificaciones promedio (60 % - 79 %), 2 estudiantes por debajo del promedio (1 % - 50 %), y 2 estudiantes no lograron completar sus evaluaciones (0 %).

En el grado de sexto sección B se contó con la participación de 27 estudiantes; de estos 2 estudiantes resultaron con calificaciones excelentes, 10 con calificaciones promedio alto, 8 con calificación promedio, 6 estudiantes por debajo del promedio, y 1 estudiante que no logró completar sus evaluaciones.

Estos números indican que la sección A fue más efectiva que la sección B; un factor importante es que la sección A tuvo una enseñanza mucho más personalizada, ya que se tuvo que dividir en dos grupos para usar el laboratorio de cómputo, esto hacía que cada grupo tuviera 16 estudiantes por periodo; estos estudiantes seleccionaban las computadoras que tuvieran la velocidad más rápida y que sus componentes como el *mouse* sirvieran, a diferencia de la sección B, cuyo número total de estudiantes recibía el curso en un periodo, es decir, los 27 estudiantes estaban en el laboratorio, lo que causaba distracciones, ya que se generaba más ruido.

Algunos estudiantes no ponían de su interés para resolver los problemas y también las computadoras antiguas no ayudaban al estudiante a realizar su ejercicio en el tiempo correspondiente.

4. PROPUESTA PARA MEJORAR LA IMPLEMENTACIÓN DEL FUTUROS PROYECTOS SIMILARES

4.1. Metodología

Luego de impartir el curso durante seis meses, se tiene una perspectiva más completa del mismo. A continuación se presentan algunas propuestas para lograr un mayor impacto en la implementación de futuros EPS similares.

4.2. Escuela candidata antes de realizar el anteproyecto

Un factor muy importante es contar con la escuela candidata antes de entregar el anteproyecto, para no retrasar la entrega y elaboración del mismo, ya que muchos factores dependen de la escuela, y otros aspectos a considerar y que deben plasmarse en el anteproyecto.

4.3. Coordinar el calendario

El calendario de actividades debe conocerse antes de realizar el cronograma propuesto, ya que de ello depende si se cumplirá con los objetivos para impartir las clases; las escuelas tienen ya un calendario escolar, el cual puede descargarse desde la página del Ministerio de Educación para verificar las fechas de evaluación, fechas de asuetos y feriados.

La escuela también posee fechas de actividades especiales; esto es importante considerar en el cronograma. Es necesario preguntar a la escuela

por las fechas de dichas actividades; tales como actos cívicos, mañanas deportivas, entregas de notas, y otras fechas específicas.

El laboratorio de cómputo también posee fechas en las cuales se les hace mantenimiento a las mismas; es importante conocer para dichas fechas, tomarlas en cuenta en la elaboración del cronograma o seguimiento del mismo, ya que si no se conocen desde el inicio, por lo menos una semana antes, y para redistribuir el contenido del programa en los demás días y cumplir con el cronograma.

4.4. Proyector

El uso del proyector facilita en gran manera el avance del contenido a impartir, y la comprensión en los estudiantes, ya que brinda un medio visual y aprenden más rápido. Reduce el tiempo para exponer; durante la implementación de este EPS al inicio no se tenía proyector y las clases se volvían monótonas, no se lograba transmitir bien el conocimiento ya que la gran mayoría del contenido consiste en realizar ejemplos; desde el momento en que se tuvo proyector las clases fueron más dinámicas, y fueron más fáciles de transmitir el conocimiento, ya que los estudiantes podían ver de forma visual lo que se estaba realizando en el programa.

4.5. Material de apoyo al estudiante

El material de apoyo por estudiante es muy importante, ya que le brinda una guía ya definida, no tienen que escribir los enunciados, y tienen diversos apartados para contestar y poner en práctica sus conocimientos; en este material existen ejercicios para llevarse a casa y son importantes para que el estudiante amplíe sus conocimientos y la forma de resolver los problemas.

El material que se pudo conseguir no alcanzó para todos los estudiantes y se pudo notar la diferencia cuando no utilizaban el material didáctico.

4.6. Incluir el contenido de plano cartesiano y grados

Dentro del contenido del cuaderno “Programación con Scratch” no se explica el plano cartesiano ni los grados; es muy importante que los estudiantes conozcan este contenido ya que al momento de utilizar Scratch y controlar los bloques, deben poseer estos conocimientos. (En el anexo de este informe se genera dicho contenido para utilizarlo).

CONCLUSIONES

1. El estudiante comprendió el significado de ciencias de la computación y el pensamiento computacional.
2. El estudiante aprendió y puso en práctica las sentencias, bloques con instrucciones de Scratch, funciones y algoritmos de programación, al igual que el diagrama de flujo.
3. El estudiante desarrolló el pensamiento computacional, poniendo en práctica la formulación del problema, ver el problema desde varias perspectivas y seleccionar la solución más óptima.
4. Los estudiantes encontraron una motivación en las ciencias de la computación, pudiendo su funcionalidad en muchos campos; ahora pueden relacionarse con todas ellas, y cuentan con un panorama más completo.
5. Los estudiantes demostraron ser capaces de resolver problemas, ya que se evaluó mediante ejercicios y prácticas los conocimientos adquiridos; esto indica que no importa la condición económica o social; la tecnología ayuda al desarrollo cuando se le brinda la oportunidad a la persona.

RECOMENDACIONES

1. Seguir con este tipo de proyecto en EPS, ya que de esta manera se ayuda al desarrollo del país, debido a que son las pequeñas cosas que hacen que sucedan grandes cambios.
2. Tener una visión clara del proyecto, transmitirla constantemente a los estudiantes; esto hace que los estudiantes se enfoquen y presten atención al proyecto decididos a aprender y no solo a ganar una clase.
3. Compartir los avances con los maestros, el director y con los estudiantes del establecimiento; esto ayuda a crear un ambiente positivo y motiva a los estudiantes a seguir adelante.

BIBLIOGRAFÍA

1. *Acerca de Scratch*. [en línea]. <<http://scratch.mit.edu/about/>>. [Consulta: 10 de junio de 2014].
2. CORTESE, Abel. *Técnicas de estudio*. [en línea]. <<http://www.tecnicas-de-estudio.org/inteligencia/creativa1.htm>>. [Consulta: 10 de julio de 2014].
3. Commons. *Licencia Creative*. [en línea]. <<http://creativecommons.org/licenses/by-nc-sa/3.0/deed.es>>. [Consulta: 10 de junio de 2013].
4. Edulibre. *Software y educación*. [en línea]. <<http://edulibre.net/>>. [Consulta: 10 de julio de 2014].
5. Fundación Sadosky. [en línea]. <www.funsepa.org>. [Consulta: 10 de julio de 2014].
6. HILL, Napoleón; W. STONE, Clement. *Success through a positive mental attitude*. New York: Pocket Book, 1960. 359 p.
7. International Telecommunication Union -ITU-. *Medición de la sociedad de la información*. Ginebra, Suiza: Place des Nations CH-1211, 2013. 44 p.
8. KENNEDY John F. *Presidential Library and Museum - Columbia Point*, Boston MA. [en línea]. <<http://www.jfklibrary.org/Education/>>

Teachers/CurricularResources-Image-List/Elementary- School-Curricular-Materials/Ask-not-wh at-your-country-can-do-for-you. as px14>. [Consulta: junio de 2014].

9. LÓPEZ GARCÍA, Juan Carlos. *Programación con Scratch*. [en línea]. <www.eduteka.org.>. [Consulta: 10 de junio de 2014].
10. Ministerio de Educación. [en línea]. <http://estadistica.mineduc.gob.gt/>. [Consulta: 4 de julio de 2013].
11. New Jersey's Science & Technology University. [en línea]. <http://catalog.njit.edu/undergraduate /programs/ computer science.php>. [Consulta: 14 de mayo de 2014].
12. Fundación Sadosky. *Una propuesta para refundar la enseñanza de la computación en las escuelas Argentinas*. Buenos Aires: 2013. 39 p.
13. RESNICK, Mitchel.. *Scratch lenguaje de programacion*. [en línea]. <http://es.wikipedia.org/wiki/Scratch_>. [Consulta: 10 de junio de 2014].
14. Technology Research Gartner Inc. Gartner Says More than 1 Billion PCs in use Worldwide and Headed to 2 Billion Units by 2014. [en línea]. <http://www.gartner.com/newsroom/id/703807>. [Consulta: 4 de junio de 2014].

15. The George Washington University. *Department of Computer Science*, [en línea]. <<http://www.seas.gwu.edu/~simhaweb/misc/cscareers.html>>. [Consulta: 14 de mayo de 2014].
16. Unión Internacional de Telecomunicaciones. *Medición de la sociedad de la información* . Suiza: 2013. 8 p.

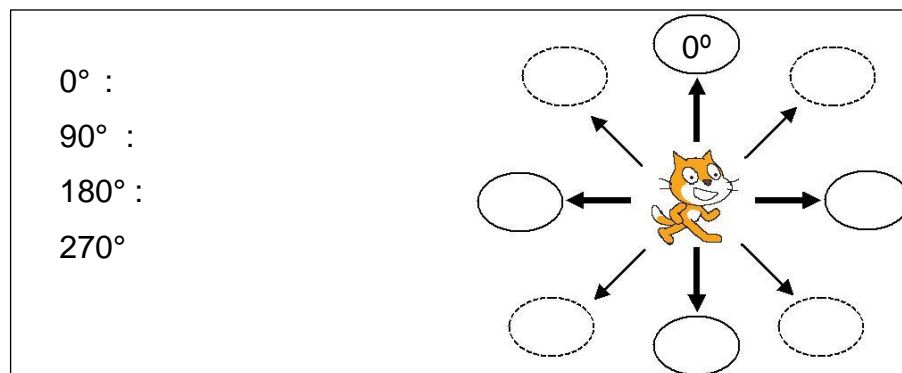
APÉNDICES

Al iniciar Scratch, este siempre comienza con el Objeto Gato en un Escenario blanco, a partir de ahí podrá desarrollar sus proyectos. Debemos tener en cuenta lo siguiente:

El escenario: es donde usted ve cómo sus historias, juegos y animaciones cobran vida. El escenario tiene 480 unidades de ancho y 360 unidades de alto y está dividido en un plano cartesiano $x - y$. El centro del escenario corresponde a las coordenadas $x:0, y:0$.

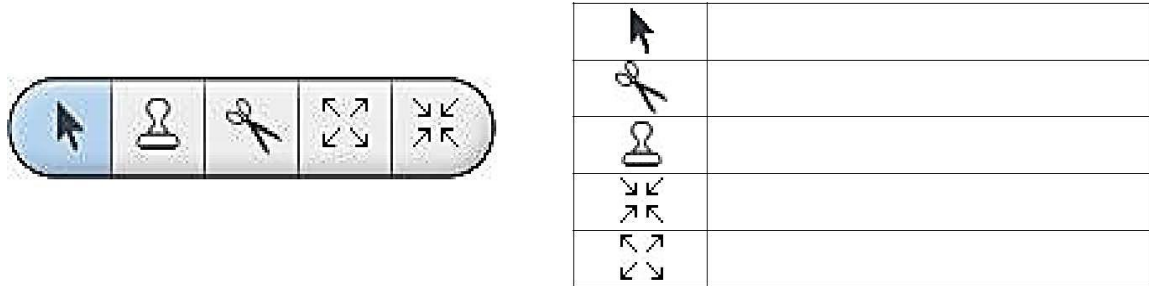
Test de autoevaluación

1.- Descubre y explica los giros del gato:



Fuente: elaboración propia.

2.- Describe las acciones que se ejecutan con los siguientes botones de la barra de herramientas



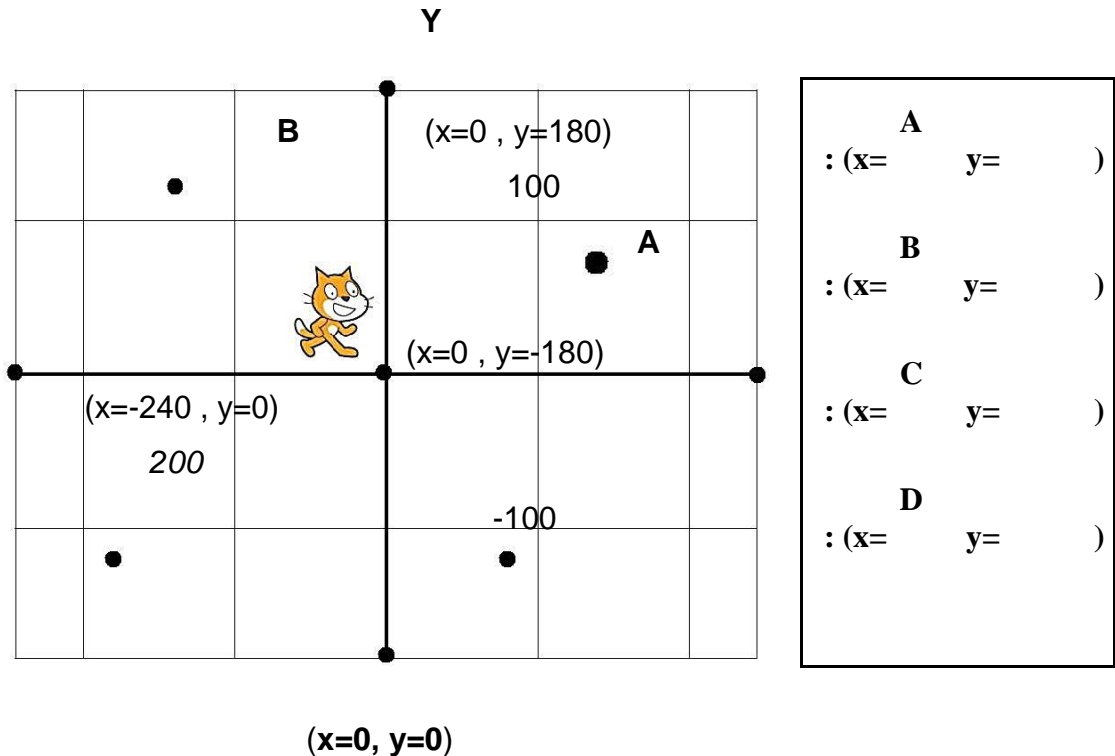
Fuente: elaboración propia.

3.- Explique la diferencia que hay en la lectura de las siguientes coordenadas:



Fuente: elaboración propia.

4.- Ubicar las coordenadas aproximadas de los puntos señalados en el Escenario:



4. Describe el algoritmo para preparar un huevo frito :inicio

_____	_____
_____	_____
_____	_____
_____	_____
_____	_____

FIN

Fuente: elaboración propia.

Actividades prácticas

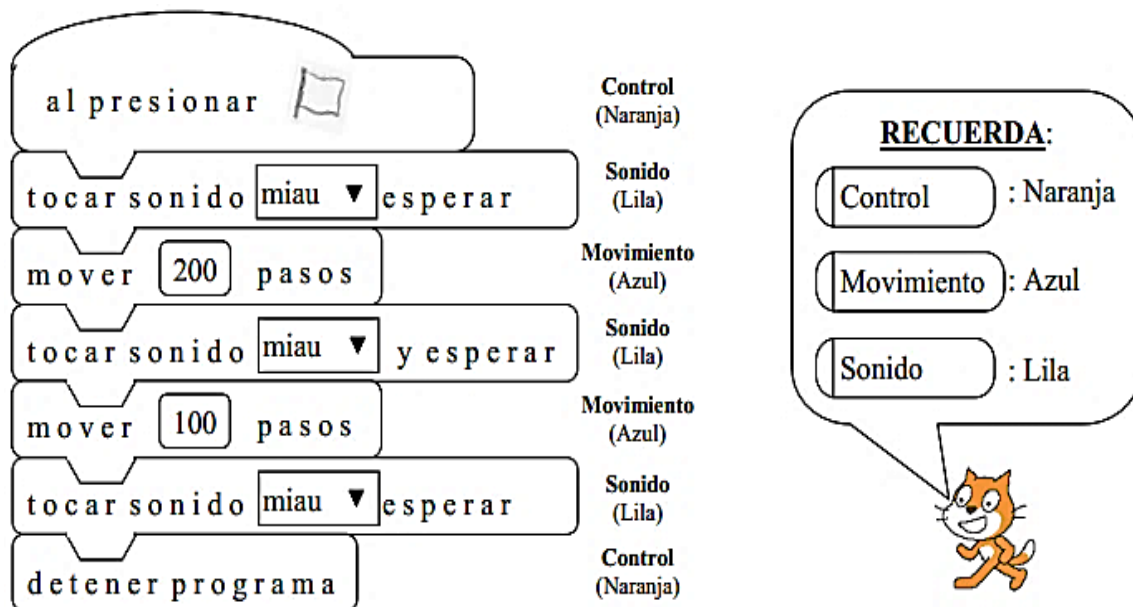
En cada programa se debe ejecutar lo siguiente:

1. Colorear cada orden con sus colores respectivos
2. Ejecutarlo en la PC

Guardarlo con nombre extraído del título de cada programa

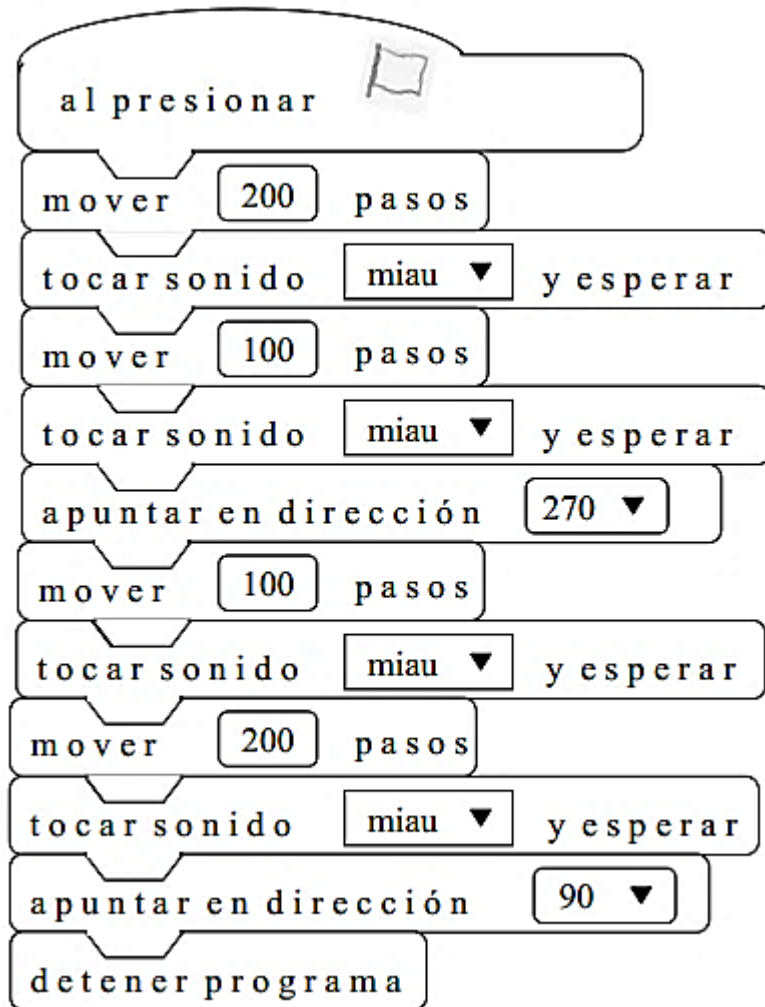
Programando 1: iniciando en scratch (Control–Movimiento-Sonido)

Programa 1: primeros pasos: gato debe avanzar maullando.



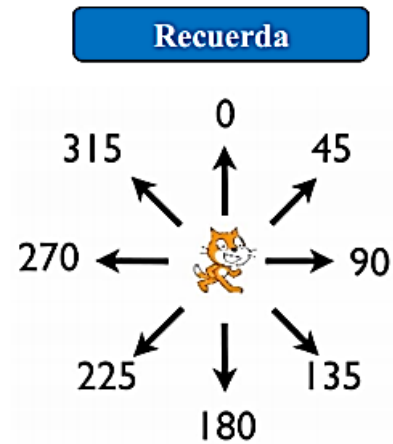
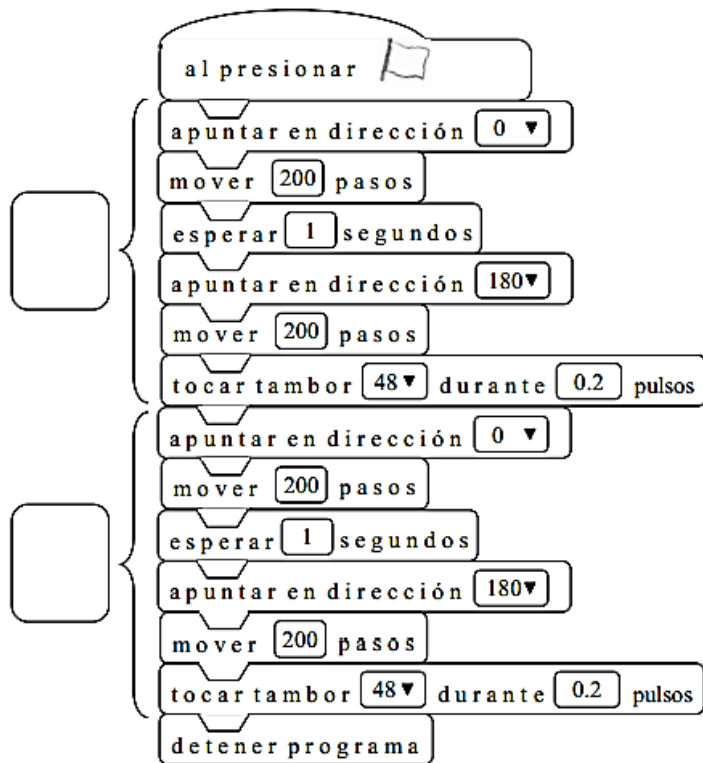
Fuente: elaboración propia.

Programa 2: Ida y vuelta: gato debe avanzar maullando pero debe regresar.



Fuente: elaboración propia.

Programa 3: Pelota rebota dos veces: Hacer que una pelota añadida al escenario rebote (arriba-abajo) tocando sonido de tambor.



Fuente: elaboración propia.