

Programación con la herramienta SCRATCH +CARAMBA. Una experiencia de aprendizaje significativo

Jesennia Cardenas, MSc¹, David Benavides, Phd², Mayra D'Armas, Phd¹, Mariuxi Vinueza, MSc¹
y Jorge Rodas, MSc¹

¹Universidad Estatal de Milagro, Ecuador, jcardenas@unemi.edu.ec, mdarmas@unemi.edu.ec, mvinuezam@unemi.edu.ec,
jrodass@unemi.edu.ec

²Universidad de Sevilla, España, benavides@us.es

Resumen— *El alto índice de reprobados es un problema generalizado en la enseñanza de la programación en los espacios educativos formales, en el caso de las carreras Ingeniería de Sistemas Computacionales e Ingeniería Industrial de la Universidad Estatal de Milagro, Ecuador, se han impulsado iniciativas que van en pro del buen desempeño del estudiante. La herramienta de programación SCRATCH y el sistema de recomendación CARAMBA, se aplican como una alternativa a las estrategias clásicas de enseñanza. La investigación reportada analiza con base en la percepción de los estudiantes del curso pre-universitario, el impacto del uso de herramientas alternativas como el SCRATCH en el proceso de enseñanza y aprendizaje de la programación. Se concluye que debe mantenerse la aplicación de SCRATCH +CARAMBA con los estudiantes del pre-universitario y continuar con el descenso a los niveles académicos inferiores en forma progresiva, es decir colegios y escuelas del sector.*

Keywords— *aprendizaje significativo, ingeniería, programación, SCRATCH, sistema de recomendación.*

Abstract— *The high rate of failures is a widespread problem in the teaching of programming in formal educational places. In the case of the Computer Systems Engineering and Industrial Engineering programs at the State University of Milagro, Ecuador, initiatives to encourage good performance of students have been promoted. The SCRATCH programming tool and the CARAMBA recommendation system are applied as alternatives to traditional teaching strategies. The reported research, which is based on the students' perception of the pre-university course, analyses the impact of using alternative tools such as SCRATCH in the teaching and learning process of programming. It is concluded that the application of SCRATCH + CARAMBA should be kept with pre-university students and also continue progressively with the other lower academic levels, it means in high schools and primary schools in the area.*

Keywords— *meaningful learning, engineering, programming, SCRATCH, system of recommendation.*

I. INTRODUCCIÓN

La reprobación es un fenómeno que impacta en las Instituciones de Educación Superior, sobre todo en las carreras de ingeniería, y si bien es un problema antiguo se considera necesario seguirlo estudiando para tener un acercamiento a la posible solución de este complejo problema [1]. En el caso particular de la Universidad Estatal de Milagro (UNEMI), Ecuador, un estudio realizado en el primer semestre de la

carrera Ingeniería en Sistemas de la UNEMI en el período mayo – septiembre 2015 sobre el rendimiento académico de los estudiantes se reportó una alta tasa de reprobación en la asignatura Fundamentos de Programación, lo que evidencia en este caso en particular, ese fenómeno generalizado de bajo rendimiento en Ingeniería; sin embargo, esta realidad es ignorada por los diferentes actores que participan en el proceso de enseñanza-aprendizaje.

La reprobación en el nivel de educación superior trae como consecuencia el rezago y la deserción, por lo que la reducción de los índices de reprobación estudiantil es tarea que demanda la participación de todos los involucrados en los distintos niveles de la comunidad educativa. Este problema, requiere, entre otras acciones, un cambio en la forma en que los estudiantes visualizan su estadía en las instituciones de educación superior, en la forma en que los docentes realizan su labor y una intervención inmediata de las autoridades educativas con acciones que garanticen la incorporación de elementos que favorezcan el análisis y prevención de las causas de reprobación, mediante la aplicación de mecanismos de control e instancias de toma de decisiones [1].

Por otro lado, el desarrollo de criterios y actividades de programación computacional para los distintos niveles y modalidades del sistema educativo resulta una temática ponderada en la agenda educativa actual [2]. La combinación de elementos de las tendencias pedagógicas contemporáneas y la enseñanza asistida por computadoras aporta medios de enseñanza apoyados en las nuevas tecnologías de la información y las comunicaciones que contribuyen a elevar la calidad del proceso docente educativo de las asignaturas de programación en la carrera de Ingeniería Informática [3].

En el contexto educativo formal iberoamericano, diferentes países están priorizando la introducción de la programación en sus programas educativos [2]; destacándose Argentina, donde, universidades nacionales y privadas han diseñado e implementado diversas propuestas de formación tendientes a facilitar el ingreso y la permanencia de los estudiantes en carreras que demandan la comprensión de lenguajes de programación [2].

A pesar de que en la asignatura se aplican conceptos básicos y algoritmos relativamente sencillos, a los estudiantes se les hace difícil la comprensión de los conceptos tales como

variables, estructuras condicionales, estructuras repetitivas, arreglos, funciones de los lenguajes de programación. Estas dificultades se manifiestan independientes del paradigma y/o lenguaje utilizado. Esto se puede deber a diversos factores, tales como motivación, sintaxis, estilos de aprendizajes diferentes, experiencia previa, entre otros [4].

De la misma manera hay otras causas que impiden el correcto desarrollo de los estudiantes, entre las cuales se destacan: Metodología no acorde que utiliza el docente, Bajo índice de alumnos que estudiaron programación en la secundaria, Mala selección de la carrera, entre otras. A su vez estas mismas causas generan otros problemas en los estudiantes como: Bajas calificaciones, Bajo cumplimiento de tareas de programación, Alto índice de faltas en las horas destinadas a la materia.

Esta situación priva al estudiante el desarrollo de sus habilidades lógicas y de razonamiento, por lo que es necesario aplicar alternativas para que ellos desarrollen sus capacidades de forma óptima. Es decir, es necesario que se innove la manera en las que las clases son impartidas por el docente, mejorando las metodologías ya existentes con el uso de nuevas herramientas que les faciliten la comprensión y resolución de los problemas planteados, para mejorar el rendimiento académico, así como el desarrollo de las capacidades de los estudiantes.

Existen diferentes herramientas para enseñar fundamentos de programación, entre las que se destaca SCRATCH. Se han efectivizado variadas iniciativas en diferentes niveles educativos en torno a la introducción de la programación, en las que SCRATCH aparece como una herramienta destacada [2].

El uso de nuevas metodologías de educación en la asignatura de Fundamentos de Programación apoyados en una herramienta pertinente, permitirá explotar el potencial de los estudiantes, preparando profesionales de calidad y mejorando la aceptación de la carrera y de la universidad.

El objetivo de la investigación reportada es conocer, con base en las percepciones de los nuevos estudiantes, el impacto que tiene el uso de la herramienta SCRATCH en el proceso de enseñanza-aprendizaje de Fundamentos de Programación.

A. SCRATCH

La plataforma SCRATCH, propone un lenguaje de programación basado en bloques, donde la gramática visual de los mismos y sus reglas de combinación tienen el mismo rol que la sintaxis en los lenguajes basados en texto como C, Java o Python [2].

Para el presente proyecto, se ha decidido la utilización de SCRATCH para la creación de las actividades educativas para enseñar a programar por diversas razones: plataforma muy desarrollada, el componente social, el trabajo colaborativo, facilidad de uso, variedad de bibliografía [5].

B. SCRATCH en la Enseñanza de Fundamentos de Programación. Una aproximación al estado del arte

Recientemente se han logrado visualizar diferentes iniciativas asociadas al uso de la herramienta SCRATCH como ayuda al rendimiento académico en estudiantes novatos de carreras relacionadas con la informática. Por ejemplo en [6], se utiliza la herramienta SCRATCH y Lego Mindstorms como ayuda en la enseñanza aprendizaje de la asignatura Fundamentos de Programación, resultando un incremento considerable en la tasa de aprobación de los estudiantes que cursaban la asignatura por primera oportunidad.

Tanrikulu y Schaefer [7] presentan un trabajo donde analizan los aspectos específicos mediante la identificación de las dificultades que experimentan los programadores con SCRATCH durante un taller de cuatro días. Según los resultados obtenidos por los usuarios consideran que esta herramienta es fácil de aprender debido a su interfaz sencilla y visual que hace que sea rápido el aprendizaje para que puedan descubrir y entender la funcionalidad. El Instituto de Tecnología de Massachusetts en julio de 2008 reunió a educadores de todo el mundo que se dan cuenta que SCRATCH es una fascinante puerta en la disciplina de la programación, tomando como referencia a una creciente comunidad que sigue produciendo una gama inesperada de ejemplos desarrollados en forma creativa, producir maravillosos juegos, historias, arte, ciencia simulaciones y modelos matemáticos [8].

Las escuelas secundarias de Sudáfrica, RoboMind, SCRATCH y B #, fueron seleccionadas para la evaluación de un estudio de investigación. Los tres PAT siempre diferentes enfoques mientras que ser capaz de soportar la programación Delphi lenguaje utilizado en las escuelas tuvo una mejor comprensión de los conceptos de programación y motivación hacia la programación de los estudiosos que no utilizaron un PAT. Poco se sabe acerca de lo que aprenden en la programación de un juego [9].

Los resultados de [10] proporcionan la evidencia de que la construcción de juego que involucra tanto las actividades de diseño y programación puede apoyar el aprendizaje de conceptos de informática.

Fessakis y otros consideraron una competencia importante para el desarrollo de orden de pensamiento superior, habilidades para resolver problemas algorítmicos. A partir de entrevistas a los maestros y las notas del investigador se realizó una evaluación realista de la viabilidad, la pertinencia y el valor del aprendizaje de la integración de la programación de computadoras en tal contexto. Resultados de Interés sobre los niños que aprenden, las dificultades, las interacciones, las estrategias de resolución de problemas y el papel del profesor son reportados [11].

Según [12] la resolución de problemas es un proceso complejo, que implica varias operaciones cognitivas tales como la recogida información, la selección de la estrategia heurística y la metacognición, el impacto de la Metodología

Digital Object Identifier: (to be inserted by LACCEI).
ISSN, ISBN: (to be inserted by LACCEI).

Experimento computacional en rendimiento cognitivo alumnos, la resolución de problemas mediante modelos computacionales. Sesenta futuros maestros de primaria se ofrecieron como voluntarios para participar en el estudio, los resultados del experimento muestran que el uso del enfoque experimento computacional tiene un efecto sustancial sobre las experiencias metacognitivas y el uso de indicadores de modelado.

II. METODOLOGÍA

A. Planteamiento del problema

Fundamentos de Programación es una asignatura de suma importancia para la carrera de Sistemas Computacionales que aborda tipos de datos, variables, constantes, declaración de variables, estructuras condicionales, estructuras repetitivas y uso de algoritmos para la resolución de problemas.

En un estudio realizado con los docentes de la asignatura se pudo constatar que la herramienta pedagógica que usan con frecuencia en clases es la resolución de problemas con la aplicación de organizadores gráficos como mapas mentales o mapas conceptuales, siendo el contenido dirigido al lenguaje C++.

Los docentes consideran que una de las principales causas que motiva el bajo rendimiento en la asignatura Fundamentos de Programación es en primer lugar que vienen direccionados por la Secretaría de Educación Superior, Ciencia, Tecnología e Innovación (SENESCYT), es decir la mayoría de los estudiantes fue asignado por la SENESCYT en la carrera de Ingeniería de Sistemas Computacionales; segundo en el Pre-Universitario no reciben la asignatura de Introducción a los Fundamentos de Programación; y en tercer lugar la desmotivación originada por el mismo hecho de que son direccionados a una especialidad que ellos no querían entonces obviamente ellos no le dan mayor interés. Estos son factores fundamentales para que el rendimiento académico sea bajo.

Otro factor que incide en el rendimiento académico de los estudiantes es el conocimiento previo dado en los colegios. En los cursos hay una miscelánea de especialidades y por tratarse de estudiantes de colegios rurales muy pocos de ellos reciben conocimiento sobre ofimática, mucho menos fundamentos de programación, con excepción de unos pocos que provienen del desaparecido bachiller informático. Y en esos casos tienen un esquema cognitivo que es difícil modificarles, convirtiéndose en un problema más que una ventaja.

En general los estudiantes piensan que programar es difícil, de hecho, los cursos de introducción a la programación tienen una alta tasa de abandono y normalmente los alumnos no aprenden a programar bien [13]. Es así como se decidió aplicar la herramienta SCRATCH como parte de las estrategias de enseñanza-aprendizaje de Fundamentos de Programación, así como instrumentos de diagnóstico que permitieran conocer la percepción que tienen los estudiantes de nivelación (Curso Introductorio) sobre la problemática planteada.

B. Marco metodológico

Objetivo de la Investigación: analizar el proceso de enseñanza-aprendizaje en la asignatura Fundamentos de Programación para la generación de conocimiento científico que pueda ser útil en la mejora del rendimiento académico de los estudiantes de Fundamentos de Programación del curso de nivelación (pre-universitario) de las carreras de Ingeniería Industrial y de Ingeniería en Sistemas de la Universidad Estatal de Milagro, Ecuador.

En esta investigación se aplicó la herramienta denominada CARAMBA, la cual es un sistema de recomendación que permite a los estudiantes seleccionar los ejercicios a desarrollar en SCRATCH, los ejercicios son sugeridos a través de la comparación con otros usuarios de perfil similar. Esto permite recomendar ejercicios acorde al nivel de conocimiento y que puedan ir incrementando la complejidad, a medida que vayan resolviendo los ejercicios más sencillos. El estudiante debe evaluar al final de desarrollar los problemas, dos parámetros (gusto y complejidad). Estas dos variables permiten comparar al estudiante que se autentica con todos los usuarios que han evaluado los ejercicios que reposan en una base de conocimiento que administra el sistema de recomendación CARAMBA.

CARAMBA fue aplicado anteriormente a estudiantes de las carreras de Ingeniería en Sistemas e Ingeniería Industrial en la UNEMI, permitiendo evaluar los beneficios de la integración de una herramienta informática en el proceso enseñanza aprendizaje de la asignatura Fundamentos de Programación, dando como resultados un incremento representativo en los resultados académicos reflejados en sus calificaciones y expresados en las preguntas incluidas en la encuesta al final del proceso académico.

Esta experiencia motiva a la investigación que sugiere llevar la herramienta CARAMBA cada vez a un menor nivel académico, para iniciar a una menor edad el proceso de aprendizaje de programación a través de SCRATCH. En ese contexto se decidió aplicar SCRATCH+CARAMBA en los cursos pre-universitario, es decir en el curso de nivelación previo ingreso a las carreras de Ingeniería en Sistemas e Ingeniería Industrial. Estos resultados son los que se presentan en este artículo.

Ficha Muestral: se aplicaron tres (03) modelos de Cuestionario. Los estudiantes debían responder a esos tres (03) cuestionarios, aplicados en diferentes momentos durante el desarrollo del curso de nivelación. El *Cuestionario 1*, fue aplicado antes del desarrollo del curso, o sea, antes de la aplicación de la herramienta SCRATCH como parte de las estrategias de enseñanza-aprendizaje de Fundamentos de Programación. Los *Cuestionarios 2* y *3*, fueron aplicados después del desarrollo del curso.

Con el Cuestionario 1, se buscaba la percepción que tienen los estudiantes sobre el grado de fidelidad en el curso, regularidad, asistencia, interés, capacidad intelectual y expectativa de aprobación del curso de Fundamento de

Programación. Con el Cuestionario 2, se pretendía obtener la Percepción que tienen los estudiantes sobre el grado de influencia en el curso, regularidad, asistencia, interés, capacidad intelectual y expectativa de aprobación del curso de Fundamento de Programación con el uso de la herramienta SCRATCH. El cuestionario 3, se enfocaba a indagar sobre la Percepción que tienen los estudiantes sobre el grado de logro de aprendizaje, facilidad, entusiasmo e interés, aplicabilidad, motivación y obtención de calificaciones en el curso de Fundamento de Programación.

Debían responder si estaban Totalmente en Acuerdo (5), De Acuerdo (4), Parcialmente de Acuerdo (3), Parcialmente en Desacuerdo (2) o Totalmente en Desacuerdo (1). Estas preguntas fueron respondidas por 126 estudiantes. En la Tabla I se presentan las variables de cada cuestionario.

Adicionalmente, en el Cuestionario 1, se les solicitaba a los estudiantes datos demográficos para posterior comparación, que incluían Edad, Sexo, Nivel de instrucción de los padres, Nivel Socio-Económico y Promedio de Calificaciones en la Secundaria.

TABLA I
VARIABLES CONTEMPLADAS EN LOS CUESTIONARIOS

Cuestionario 1	Cuestionario 2	Cuestionario 3
<p>1o Recibí clase de Fundamentos de Programación en el colegio</p> <p>1a Si tuviese la oportunidad de escoger mi opción de estudios, escogería la que estoy cursando actualmente</p> <p>1b Cumplo con regularidad las tareas encomendadas por el docente</p> <p>1c Asisto frecuentemente a clases</p> <p>1d Me interesa la asignatura Fundamentos de Programación</p> <p>1e Considero que tengo capacidad intelectual suficiente para superar la asignatura Fundamentos de Programación</p> <p>1f Considero que aprobaré la asignatura de Fundamentos de Programación</p>	<p>2a Tengo más posibilidades de ser promovido si se utiliza la herramienta pedagógica SCRATCH</p> <p>2b Considero que la herramienta SCRATCH contribuirá a mejorar mis calificaciones en la asignatura Fundamentos de Programación</p> <p>2c Considero que SCRATCH puede contribuir a las metodologías utilizadas por el docente de la asignatura de Fundamentos de Programación</p> <p>2d Fue necesario tener conocimientos previos de programación para poder usar la herramienta SCRATCH</p> <p>2e Me gustaría asistir a cursos donde se profundicen los conceptos y usos de la herramienta SCRATCH</p> <p>2f La inclusión de SCRATCH en el contenido de la asignatura de Fundamentos de Programación cambió positivamente mi perspectiva de la asignatura</p> <p>2g Me gustaría que el software SCRATCH sea considerado como herramienta pedagógica para impartir la clase de Fundamentos de Programación</p> <p>2h Creo que utilizar el SCRATCH frente a otros métodos de enseñanza puede mejorar las calificaciones de los estudiantes de la asignatura Fundamentos de Programación</p> <p>2i Creo que utilizar el SCRATCH frente a otros métodos de programación puede aumentar el interés y la motivación del alumnado por la asignatura Fundamentos de Programación</p>	<p>3a Considero que he logrado aprendizajes sólidos y duraderos en la asignatura Fundamentos de Programación</p> <p>3b Ha sido fácil el aprendizaje de la asignatura</p> <p>3c Estudié la asignatura con entusiasmo e interés</p> <p>3d Creo que lo que he aprendido es aplicable a otras asignaturas de mi carrera</p> <p>3e Creo que lo que he aprendido lo aplicaré en mi futuro profesional</p> <p>3f Me siento motivado a realizar fuera de clases actividades de profundización o complemento de los conocimientos adquiridos en esta asignatura</p> <p>3g Me siento motivado para terminar la carrera</p> <p>3h Obtuve buenas calificaciones en la asignatura</p>

III. RESULTADOS Y ANÁLISIS

De los datos demográficos obtenidos usando el Cuestionario 1, se obtiene que la edad promedio de los estudiantes que aspiran ingresar a la carrera de Ingeniería Informática, es de 18,98 años, con una desviación estándar de 2,42 años. En cuanto al promedio de notas de secundaria, se reporta una media de 8,76 (de una máxima de 10). De los 123 estudiantes, 73 (59,3%) son del sexo masculino y 50 (40,7%) del sexo Femenino. En cuanto al nivel de instrucción de los padres, se destaca que más del 80% no tiene educación universitaria (Técnica/Superior/Postgrado). Con respecto al nivel socio-económico, casi un 85% de los estudiantes se considera de clase media, quedando un 15% que se considera de clase baja.

Con respecto a los resultados de las variables del Cuestionario 1, se destaca que casi un 55% reporta que no recibió clase de Fundamentos de Programación en la secundaria, lo que evidencia que ingresan al sistema de educación superior con pocos conocimientos en el tema;

mientras que más del 40% no escogería o dudaría escoger *Ingeniería Informática* como opción de estudios.

El 87% de los estudiantes dice que asistió frecuentemente a clases, lo que apoya las opiniones expresadas; más del 95% le interesa la asignatura Fundamentos de Programación, mientras que más del 75% considera que tiene capacidad intelectual suficiente para superar la asignatura y más del 80% es optimista y considera que la aprobará.

En la Tabla II se reportan los resultados en Porcentaje para las variables consideradas.

TABLA II
RESULTADOS CUESTIONARIO I EN %

	1o	1a	1b	1c	1d	1e	1f
1	41,5	10,6	0,8	3,3	4,1	0,8	1,6
2	13,0	11,4	0,8	0,0	1,6	1,6	1,6
3	8,9	21,1	11,4	0,8	18,7	17,9	16,3
4	24,4	27,6	56,1	22,0	33,3	42,3	40,7
5	12,2	29,3	30,9	73,2	42,3	37,4	39,8

De los datos obtenidos con el Cuestionario 2 y reportados en la Tabla III, se tiene que más del 60% de los estudiantes considera que tiene más posibilidades de ser promovido si se utiliza la herramienta pedagógica SCRATCH (2a), y casi el 70% considera que su uso contribuirá a mejorar sus calificaciones en la asignatura Fundamentos de Programación (2b). Así mismo, se observa que casi el 70% de los estudiantes considera que la herramienta SCRATCH puede contribuir a las metodologías utilizadas por el docente de la asignatura de Fundamentos de Programación (2c), y más de 40% está de acuerdo en que era necesario tener conocimientos previos de programación para poder usar la herramienta SCRATCH (2d). Adicionalmente, un 78% dice que le gustaría asistir a cursos donde se profundicen los conceptos y usos de la herramienta SCRATCH (2e), y alrededor del 50% considera que la inclusión de SCRATCH en el contenido de la asignatura cambió positivamente su perspectiva de la asignatura (2f).

Es interesante resaltar que a más del 70% de los estudiantes que cursan Fundamentos de Programación, le gustaría que el SCRATCH sea considerado como herramienta pedagógica para impartir las clases de esa asignatura (2g); ese mismo porcentaje, cree que frente a otros métodos de enseñanza, ese software puede mejorar las calificaciones de los estudiantes de la asignatura (2h) y que además, puede aumentar su interés y su motivación por la asignatura Fundamentos de Programación.

TABLA III
RESULTADOS CUESTIONARIO 2 EN %

	2a	2b	2c	2d	2e	2f	2g	2h	2i
1	3,2	3,2	3,2	5,6	4,0	4,0	4,0	2,4	3,2
2	4,8	3,2	1,6	14,3	3,2	2,4	1,6	3,2	0,8
3	31,0	23,8	26,2	38,1	15,1	45,2	21,4	23,8	17,5
4	61,1	69,8	69,0	42,1	77,8	48,4	73,6	70,6	70,8
5	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0

Con respecto al Cuestionario 3, aplicado después del desarrollo de la asignatura Fundamentos de Programación y cuyos resultados se presentan en la Tabla IV, se destaca que más del 70% de los estudiantes considera que ha logrado aprendizajes sólidos y duraderos (3a); sin embargo, solamente alrededor del 30% piensa que ha sido fácil su aprendizaje (3b). Poco menos del 70% estudió la asignatura con entusiasmo e interés (3c), mientras que casi un 60% cree que lo que ha aprendido es aplicable a otras asignaturas de su carrera (3d), lo cual suena lógico por la etapa temprana en la misma.

Importante destacar que casi el 80% cree que lo que ha aprendido lo aplicará en su futuro profesional (3e), lo que evidencia la relevancia que tienen las tecnologías de la información en las ideas de los jóvenes. Se destaca también que, más del 60% de los estudiantes encuestados, se siente motivado a realizar fuera de clases actividades de profundización o complemento de los conocimientos adquiridos en la asignatura; más del 80%, se siente motivado

para terminar la carrera (3g);, sin embargo, solamente poco más del 30% obtuvo buenas calificaciones en la asignatura.

TABLA IV
RESULTADOS CUESTIONARIO 3 EN %

	3a	3b	3c	3d	3e	3f	3g	3h
1	4,8	4,8	5,6	4,8	7,1	4,8	4,8	4,8
2	1,6	17,5	1,6	4,0	0,8	4,8	4,0	14,3
3	20,6	48,4	24,6	32,5	13,5	28,6	9,5	49,2
4	73,0	29,4	68,3	58,7	78,6	61,9	81,7	32,5
5	0,0	0,0	0,0	0,0	0,0	0,0	0,0	0,0

A.- Diseño de las actividades

Esta investigación se realiza en busca de conocer el impacto que genera el uso de una herramienta pedagógica como un nuevo elemento en el proceso educativo del que se pretende fomentar el aprendizaje activo y significativo de la lógica de programación en los estudiantes mediante la herramienta SCRATCH, poniendo a prueba sus conocimientos adquiridos de la secundaria.

Cabe resaltar que muchos de los estudiantes vienen con déficit en el aprendizaje y otros de diferentes especialidades, ya que por lineamientos de la SENESCYT se debe evaluar al estudiante, y dependiendo del puntaje de la evaluación se le asigna la carrera universitaria que deberá cursar. Esto también es una de las causas por la que el estudiante no le pone mucho interés al aprendizaje de esta asignatura y es así que a mitad de periodo educativo se retiran de la carrera, lo que se pretende con esta nueva herramienta es ayudar a muchos estudiantes a salir de la rutina de lo memorístico de conceptos, técnicas y estructuras sintácticas que hacen limitar al docente.

Se intenta probar con los estudiantes y cambiar el enfoque y la forma de enseñanza y aprendizaje, llegando hacia el estudiante donde se le despertará esa curiosidad investigativa y sobre todo la creatividad; al utilizar SCRATCH se generará una gran variedad de necesidades y retos que pondrán a prueba los conocimientos adquiridos.

Se pretende utilizar las mismas variables empleadas en la versión antes de conocer SCRATCH +CARAMBA, pero ahora consultando la percepción que tienen luego de haber utilizado la herramienta.

Las actividades a realizar se distribuirán en 8 sesiones con una duración de dos horas cada una. En cada tarea el docente propondrá su propio mecanismo para relacionarlo con la herramienta SCRATCH. Para esto, los estudiantes tendrán que investigar sobre los conceptos básicos relacionados con fundamentos de programación, tales como: variables, constantes, declaración de variables, declaración de constantes, mensajes, estructuras condicionales, estructuras repetitivas. Los lineamientos a seguir se describen a continuación:

1. Charla con estudiantes novatos de la asignatura de Fundamentos de Programación.

Esta directriz se basa en los puntos de vista de los

estudiantes hacia la asignatura y definir sus ventajas y desventajas al trayecto del periodo educativo sobre cómo asimila la materia de Fundamentos de Programación.

2. Estructuración de las actividades según el sílabo en la asignatura.

En periodos educativos anteriores muchos de los estudiantes vienen de colegios y comparan su sílabo de esta asignatura de la institución con su anterior institución, el cual a pesar de venir de instituciones públicas hay diferencias en el contenido programático de las asignaturas.

3. Implementación de prueba de diagnóstico.

Alineados con el contenido del sílabo que se manejan los docentes se aplicarán una prueba de diagnóstico para los estudiantes en la asignatura de Fundamentos de Programación y de esta manera poder evaluar sus conocimientos adquiridos hasta el momento sobre la asignatura.

4. Aplicación de encuestas e implementación.

Basándose en las hipótesis se definirán las variables a estudiar y se diseñará el instrumento a aplicar a los estudiantes para obtener datos estadísticos sobre lo que ocurre en esta asignatura de Fundamentos de Programación.

5. Presentación de SCRATCH+CARAMBA como herramienta metodológica para que sea de ayuda para los estudiantes.

Este lenguaje utilizado para programar sin preocuparse en los errores de sintaxis como lo es en C++ o Java, las cuales usan las mismas estructuras de control y funciones, lo que hace que la programación sea mucho más fácil e incentive al estudiante por aprender y sean investigativos.

6. Aplicación de la herramienta a los talleres realizados en el aula de clase.

Los ejercicios impartidos por el docente son plasmados en la nueva herramienta resueltos por los mismos estudiantes para notar su aprendizaje adquirido y observar en que parte de la programación se les dificulta pudiendo así reforzar esa duda por parte del estudiante.

7. La mecánica de los juegos realizados con esta herramienta debe ser significativa para el estudiante y debe lograrse que pueda comprender lo que está compilando.

Con SCRATCH+CARAMBA la programación es menos compleja y sobre todo que se maneja bajo objetos como una forma especial de programar el cual se expresaría las cosas en la vida real. Además, la recomendación de ejercicios que le permite avanzar de nivel permite que el estudiante no abandone el aprendizaje.

8. Tratamiento estadístico de los resultados estadísticos. Contrastando la primera prueba de diagnóstico y la final. Así como la incidencia en el rendimiento académico de los estudiantes.

IV. CONCLUSIONES

En este trabajo se ve reflejado el problema del bajo rendimiento académico de los estudiantes universitarios en la

asignatura de Fundamentos de Programación, el cual muestra que muchos ellos vienen de otras especialidades en la secundaria por tal motivo se les dificulta el aprendizaje de dicha asignatura. Otra situación relevante que se observa es la el entusiasmo y motivación que tiene el estudiante por adquirir nuevos conocimientos usando herramientas innovadoras.

En función a esta situación se propone una nueva herramienta pedagógica para que los docentes impartan la asignatura de una manera más práctica y didáctica llamando la atención de los estudiantes en su gran mayoría avanzando más rápido en los contenidos y con la opción de profundizar más en los temas de Fundamentos de Programación.

Se plantea mantener la aplicación de SCRATCH +CARAMBA con los estudiantes del pre-universitario y continuar con el descenso a los niveles académicos inferiores en forma progresiva, es decir colegios y escuelas del sector.

REFERENCIAS

- [1] M. Amado, A. García, R. Brito, B. Sánchez y C. Sagaste, "Causas de reprobación en ingeniería desde la perspectiva del académico y administradores", *Ciencia y Tecnología*, Vol. 14, pp. 233-250, 2014.
- [2] N. Monjela y P. San Martín, "Programar con SCRATCH en contextos educativos: ¿Asimilar directrices o co-construir Tecnologías para la Inclusión Social?," *Praxis educativa*, Vol. 20, No. 1, pp. 61-71, enero-abril 2016.
- [3] Y. Soler y M. Lezcano, "Consideraciones sobre la tecnología educativa en el proceso de enseñanza-aprendizaje. Una experiencia en la asignatura Estructura de Datos", *Revista Iberoamericana de Educación*, Vol. 49, No. 2, pp. 1-9, 2009.
- [4] A. Carbone, J. Hurst, I. Mitchell, and D. Gunstone, "An exploration of internal factors influencing student learning of programming," in *Proceedings of the Eleventh Australasian Conference on Computing Education*, Vol. 95, Darlinghurst, Australia, pp. 25-34, 2009.
- [5] C. España, "Diseño de actividades educativas en SCRATCH para la dinamización del Museo de Informática", Trabajo de Grado, Ingeniería Informática, Universitat Politècnica de València, España, 2015.
- [6] R. Muñoz, T. Barcelos, R. Villarroel, M. Barría, C. Becerra, R. Noel e I. Silveira, "Uso de SCRATCH y Lego Mindstorms como Apoyo a la Docencia en Fundamentos de Programación", *Actas de las XXI Jornadas de la Enseñanza Universitaria de la Informática*, pp. 248-254, 2015.
- [7] E. Tanrikulua y B. Schaefer, "The users who touched the ceiling of SCRATCH ", *Procedia - Social and Behavioral Sciences*, No. 28, pp. 764-769, 2011.
- [8] U. Wolz, H. Leitner, D. Malan y J. Maloney, "Starting with SCRATCH in CS 1", in *SIGCSE '09 Proceedings of the 40th ACM technical symposium on Computer science education*, 2009, pp. 2-3.
- [9] M. Koorsse, C. Cilliers y A. Calitz, "Programming assistance tools to support the learning of IT programming in South African secondary schools", *Computer & Education*, Vol. 82, pp. 162-178, 2015.
- [10] J. Denner, L. Werner y E. Ortiz, "Computer games created by middle school girls: Can they be used to measure understanding of computer science concepts?" *Computer & Education*, Vol. 58, pp. 240-249, 2012.
- [11] G. Fessakis, E. Gouli y E. Mavroudi, "Problem solving by 5-6 years old kindergarten children in a computer programming environment: A case study", *Computer & Education*, Vol. 63, pp. 87-97, 2013.
- [12] S. Psycharis, E. Botsari, P. Mantas y D. Loukeris, "The impact of the computational inquiry based experiment on metacognitive experiences, modelling indicators and learning performance", *Computers & Education*, Vol. 72, pp. 90-99, 2014.
- [13] C. López y R. Sánchez, "SCRATCH y Necesidades Educativas Especiales: Programación para todos", *RED Revista de Educación a Distancia*, Vol. 34, pp. 2-14, 2012.