

Sistema Didáctico de Enseñanza Introdutoria a la Programación usando Scratch, PSeInt, Processing y Arduino

Ángel Montesinos Murillo, Magister¹, Karina Rosas Paredes, Magister², José David Esquicha Tejada, Magister³

¹Universidad Católica de Santa María, Perú. amontesinos@ucsm.edu.pe

²Universidad Católica de Santa María, Perú, kparedes@ucsm.edu.pe

³Universidad Católica de Santa María, Perú, josedet@gmail.com

Abstract-The article describes a new approach to teaching introductory course Programming, applied to engineering careers. The research was divided into 4 phases:

- Phase 1: Language Scratch by comic motivate students acquire the necessary logic to schedule.

- Phase 2: Using flow charts, to enhance logic by a sequence of symbols.

- Phase 3: Programming with Processing, allows to solve arithmetic problems with sequential, selective and repetitive structures; you need only know the syntax of high level programming.

- Phase 4: Programming with Arduino, the student may use the various sensors and actuators that allow interaction with the physical environment.

At the end of the course, the necessary bases have to use the various sensors (infrared light movement, temperature) and actuators (motors, screens), being able to capture information from the outside by sensors and responding to the outside world through actuators, allowing countless design projects. The guidelines will have little robots prototypes.

The use of practice guidelines, allowed a significant learning in students, and by the movement of the limbs of robots, sensors and actuators learning is reinforced.

Keywords: Scratch, flowcharts, Processing, Arduino and learning programming.

I. INTRODUCCION

En la actualidad muchos estudiantes que se inician en carreras de ingenierías, no se sienten motivados en aprender a programar, por el hecho que la enseñanza tradicional de los lenguajes de programación de alto nivel no es el adecuado [1].

El uso de la plataforma Scratch en Colombia, ya tiene resultados satisfactorios que permite incentivar al alumno en aprender a programar [5]. Actualmente existen varias plataformas virtuales, que estimulan a los jóvenes a aprender a programar y desarrollar sus habilidades de abstracción, pensamiento lógico y modelamiento estructurado. Las plataformas de Arduino y Processing conceptualmente simples, actúan directamente sobre el mundo físico mediante sensores. Por ejemplo si se tiene un niño de corta edad al cual se le intenta enseñar a programar e implementar un programa en Lenguaje C, y que escriba en la pantalla “Hola Mundo”, seguramente encontrará el proceso como muy complejo por la cantidad de requerimientos estructurales, semánticos y

sintácticos necesarios. Pero si por el contrario utilizando 3 leds y un zumbador se le enseña a programar el funcionamiento de un semáforo; no solo se le incentiva a la programación, sino que al mismo tiempo está aprendiendo conceptos de electricidad, electrónica y comprenderá el funcionamiento de un objeto de uso cotidiano, seguramente más adelante se le ocurrirá alguna idea extraña para inquietar a sus compañeros de clase, haciendo sonidos a elevadas frecuencias o emitiendo destellos espasmódicos con leds de alta intensidad. [2]

II. ORGANIZACIÓN DEL CURSO

La investigación está organizada en cuatro partes que se muestran a continuación:

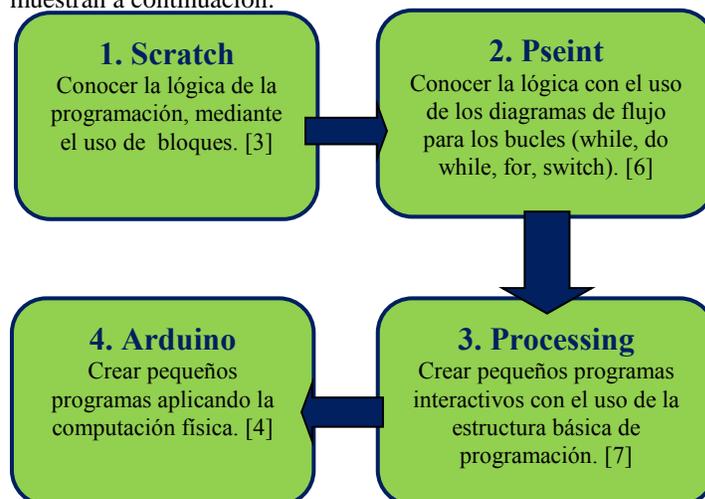


Fig. 1 Ciclo del curso introductorio de Programación

Fuente: [Propia]

Al finalizar los estudiantes mejorarán prototipos básicos ya elaborados como:

- Robot jeep, que posee 4 motores que facilita el movimiento de cada rueda independientemente (uso de instrucciones de control).
- Robot bípedo, posee 6 servomotores permitiendo mover cada pierna de acuerdo a la programación de la placa Arduino (uso de instrucciones iterativas).
- Brazo robot, posee 6 servomotores con el fin de coger objetos mediante su garra que presenta el brazo robótico

(uso de instrucciones anidadas).

- Robot tipo tortuga, posee 9 servomotores que permite mover cada pata de la tortuga como también la cabeza (uso de instrucciones anidadas).



Fig. 2 Prototipos básicos (robots)
Fuente: [Propia]

III. METODOLOGÍA

El desarrollo de la propuesta del sistema didáctico consta de 5 etapas:

- 1.- Análisis.-** Recolección de la información mediante encuestas a los estudiantes que han tenido alguna experiencia en el aprendizaje de programación en carreras de ingeniería.
- 2.- Diseño.-** Se crea el bosquejo de las guías a implantar, generando un curso adecuado para los futuros ingenieros, se debe tener en cuenta la motivación del alumno que desea aprender.
- 3.- Implementación.-** Se inicia la construcción de las guías, que serán usadas para la enseñanza del alumno.
- 4.- Resolución de los ejercicios.-** Con el docente, se implementa la resolución de las guías, dando oportunidad a que el docente se capacite y aprenda de la plataforma libre (Processing y Arduino).
- 5.- Verificación del aprendizaje.-** Posterior a la capacitación a los alumnos del primer año, mediante una encuesta y un examen de 5 preguntas para verificar el aprendizaje significativo a través el uso de las guías elaboradas en este proyecto.

La metodología propuesta sigue un ciclo de retroalimentación de 5 etapas, la cual se mejora con el paso del tiempo a las necesidades que presenta el estudiante de hoy en día. Los cinco pasos son fundamentales para que cumpla con el objetivo de mejorar la enseñanza del curso introductorio de programación

IV. CONCLUSIONES

1. La plataforma Scratch, permitió al estudiante poder crear historietas propuestas a través del uso de las guías prácticas elaboradas, siendo fundamental en el aprendizaje de la programación de computadoras.
2. La herramienta PSeInt, ayudó al alumno en aprender las estructuras iterativas (while, do while, for, switch), fundamentales en cualquier lenguaje de programación, los diagramas de flujo ayudan mediante símbolos de cada bucle en tener una mejor claridad de la estructura.
3. La plataforma Processing, refuerza el uso de las estructuras básicas que existe en cualquier aplicación a desarrollar (variables, bucles y funciones), pero de una manera distinta a la enseñanza tradicional.
4. El uso de la plataforma Arduino, requiere de conocimientos de electrónica, hardware y software, constituyendo un conocimiento actual, que con la guía propuesta se le brindará en forma complementaria.
5. Al usar la placa de Arduino los estudiantes aprendieron a programar de manera simple y divertida, con un alto grado de motivación crearon su propio proyecto. Lo que fue validado con las notas obtenidas (entre 16-20) y la encuesta aplicada indicando el 100% de estudiantes, a los que les pareció bastante sencillo programar.
6. Las plataformas propuestas de este artículo son usadas en las escuelas secundarias en países del primer mundo, para que desde corta edad vayan desarrollando su creatividad en la solución de problemas utilizando la programación de computadoras.
7. La propuesta final del proyecto, permite el uso de prototipos robots con la creatividad y lógica del alumno, crear sus propios proyectos por medio de la programación aplicando las guías de práctica diseñadas.

RECONOCIMIENTOS

A la Universidad Católica de Santa María, Arequipa.

REFERENCIAS

- [1] M. Rodríguez, "Computación Física en Secundaria", pp. 9. Marf Books, 2010.
- [2] D. O'Sullivan, T. Igoe, Physical computing: sensing and controlling the physical world with computers, Thomson, 2004.
- [3] Y. GÜLBAHAR and F. KALELIOĞLU, "The Effects of Teaching Programming via Scratch on Problem Solving Skills: A Discussion from Learners' Perspective", Informatics in Education, Vol. 13, No. 1, pp. 33-50, January 2014.
- [4] Y. Jang, W. Lee and J. Kim, "Assessing the Usefulness of Object-based Programming Education using Arduino", Indian Journal of Science and Technology, Vol 8(S1), pp. 89-96, January 2015.
- [5] Proyecto Scratch Motorola IV.
http://www.eduteka.org/pdfdir/motorola_4_informe_2013.pdf.
- [6] Página oficial de PSeInt.
<http://pseint.sourceforge.net/>
- [7] Página oficial de Processing.
<http://www.processing.org/>