

## ***Un entorno para la enseñanza y aprendizaje de programación en la Escuela Secundaria***

- ***Cenich Gabriela***
- ***Facultad de Cs. Exactas / UNICEN***
- ***ECienTec (Educación en Ciencias con Tecnologías)***
- ***gabcen@exa.unicen.edu.ar***
- ***Eje: Escuela Media y prácticas con tecnologías digitales***
- ***Tipo de comunicación: Ponencia***
- ***Abstract***

¿Cómo podemos ayudar a aprender a programar a los alumnos?. Esta pregunta se formula en el contexto de la materia “Laboratorio de Programación” del 4° Año de una Escuela de Educación Secundaria y Técnica de la ciudad de Tandil. Para avanzar en la búsqueda de respuestas se plantea el diseño e implementación de un ambiente para la enseñanza y aprendizaje de programación sobre la base de los lineamientos del modelo para diseñar Entornos de Aprendizaje Constructivista (Jonassen, 2000). Se adopta el enfoque de enseñanza de la programación “unidades simples” descrito por Selby (2011) y se utiliza el lenguaje de programación Robomind<sup>1</sup>. Se propone la integración de un espacio virtual a través de la plataforma Edmodo<sup>2</sup>. Esto posibilita disponer de los materiales para poder llevar adelante las actividades y compartir información y soluciones a los problemas entre todos los miembros de la clase.

---

<sup>1</sup> Lenguaje de programación diseñado para introducir las reglas básicas de las ciencias de la computación. Se utilizó la versión 2.6 de RoboMind por ser gratuita.

<sup>2</sup> Plataforma virtual gratuita orientada a la enseñanza.

En el presente trabajo se describe el entorno de enseñanza y aprendizaje propuesto y resultados de su desarrollo en el aula

- **Palabras Claves**

programación, Informática, enseñanza y aprendizaje, Robomind, Edmodo

- **1-Introducción**

En la actualidad la enseñanza de la programación de computadoras está concitando el interés desde distintas iniciativas que proponen diferentes herramientas y enfoques en el ámbito de la Escuela Primaria y Secundaria. Emprendimientos como Code.org (perteneciente a EEUU), Program.ar y Dale Aceptar (iniciativas nacionales) promueven el aprendizaje de Ciencias de la Computación en jóvenes y docentes.

La orientación “Técnico en Informática Profesional y Personal” de Escuelas de Educación Secundaria y Técnica en su currícula contempla la enseñanza de la programación desde el 4° año. En el año 2014 se llevó a cabo una propuesta de enseñanza de programación en el 4° Año de la EEST N°2 de la orientación citada (Cenich , 2014) guiada por el interés de encontrar respuestas a la cuestión ¿Cómo podemos ayudar a aprender a programar a los alumnos?. Se planteó un diseño flexible para la enseñanza de programación utilizando RoboMind en el marco de una visión socio constructivista del aprendizaje (Castorina y col., 1999) y la Teoría de la Actividad (Engeström, 1987). El estudio de la primera experiencia permitió establecer algunos obstáculos en el desarrollo de las prácticas educativas que permitieron generar feedbacks para repensar la propuesta de enseñanza en el marco de un ambiente con soporte virtual. Entre las limitaciones que se encontraron se pueden mencionar: tiempos diferentes de concreción de las tareas por parte de los alumnos, algunas actividades ofrecían un grado de dificultad muy grande en relación a las resueltas en una instancia anterior y por diversos motivos algunos encuentros presenciales se distanciaron 15 días o más lo que provocó la pérdida de continuidad en el desarrollo de un tema.

En el presente año se reelabora la propuesta para tratar de dar el feedback necesario para poder mejorar el diseño y ofrecer a los estudiantes mejores condiciones para que desarrollen sus aprendizajes. Se tiene en cuenta los lineamientos del modelo para diseñar Entornos de Aprendizaje Constructivista (Jonassen, 2000) que propone que “los problemas dirigen el aprendizaje, en lugar de servir como ejemplos de los conceptos y de los principios previamente enseñados”. Los problemas a resolver son “bien-definidos”

(Jonassen, 1997) considerando el enfoque de enseñanza de la programación “unidades simples” descrito por Selby (2011). Desde esta perspectiva los alumnos solucionan en primera instancia problemas simples antes de aplicar los conocimientos aprendidos a problemas más complejos. Se toma del modelo de Jonassen (2000) la idea de apoyo emocional y cognitivo para facilitar al estudiante el desarrollo de la actividad a través de secuencias de tareas ajustadas en dificultad.

Se propone en el nuevo diseño la integración de un espacio virtual a través de la plataforma Edmodo. Esto posibilita disponer de los materiales para poder llevar adelante las actividades y compartir información y soluciones a los desafíos entre todos los miembros de la clase.

En el presente trabajo se describe el entorno de enseñanza y aprendizaje propuesto y resultados de su desarrollo en el aula.

- **2-Descripción del ambiente de enseñanza y aprendizaje**

En el marco de los lineamientos del modelo para diseñar Entornos de Aprendizaje Constructivista y sobre la base del modelo “Diseño de propuestas de e-formación colaborativa” (Cenich, 2009) se plantea de manera integral los distintos elementos de la propuesta de enseñanza mediante la definición de un Sistema de Actividad Cero (SA 0) (Engeström, 1987). Lewis (1998) describe los componentes de un SA de la siguiente manera: “Un individuo (sujeto) se apoya en herramientas para lograr un objetivo (objeto) y puede aceptar reglas para trabajar en una comunidad que contribuye al objetivo mediante una división del trabajo. A partir de esta actividad, se produce un resultado.” La utilización de este modelo en el diseño de la primera experiencia facilitó la evaluación de los distintos SA y la identificación de contradicciones que contribuyeron a la reformulación del diseño original.

Propósito del SA 0: promover el desarrollo de los aprendizajes en programación estructurada de los alumnos utilizando como soporte a la clase presencial un grupo de trabajo en la Plataforma Edmodo. El sitio online se diseña como complemento al presencial para fortalecer la comunicación y disponer de un espacio común de intercambio entre los miembros de la clase.

Contexto del SA 0: La carga horaria de la materia es de 2 hs semanales y forma parte de la Formación Técnico Específica que corresponde a 12 hs semanales de las 35 hs totales a las que asisten los alumnos. En la escuela actualmente no se dispone de conexión a Internet para el trabajo en el aula, los estudiantes manifestaron tener el servicio en sus domicilios. Esta modalidad de trabajo con apoyo virtual no la habían implementado antes.

Se presenta la descripción de los componentes del SA general del curso (las modificaciones introducidas con respecto al año anterior se resaltan en negrita):

Objeto: promover en los alumnos aprendizajes de conceptos (Secuencia, Bucles, Condicionales, Operadores lógicos) y prácticas de programación estructurada.

Resultado: programas realizados por los alumnos.

Sujeto: alumno.

Comunidad: Alumnos del 4° Año y Docente del curso.

División del trabajo:

-Alumnos: resolver los desafíos propuestos.

-Docentes del curso: acompañar, guiar y animar el proceso de aprendizaje individual y grupal, observando y evaluando el desarrollo general del curso para mediar en consecuencia. Evaluar las producciones finales.

Herramientas: RoboMind, **Edmodo**, Apuntes del profesor.

Reglas:

-Participar en las actividades, proponer, discutir, valorar y negociar significados a partir de las formulaciones del docente y alumnos.

-Tener actualizado el registro completo de las actividades desarrolladas.

**-Participar del espacio virtual creado en Edmodo.**

El SA 0 está compuesto por tres actividades componentes:

-“**Control remoto**” (SA 0.1): tiene como objetivo principal que el alumno se familiarice con el entorno de trabajo y elabore sus primeros programas para que el robot se desplace y pinte en el escenario propuesto. En este SA se plantean los desafíos a resolver en el marco de imaginar que las instrucciones que se le dan al robot pertenecen a un control remoto. La analogía permite al alumno identificar al robot como un objeto que puede realizar ciertos movimientos pero siempre dirigidos por un comando, sin decisión propia.

-“**Yo Robot**” (SA 0.2): se plantea como meta que el alumno logre escribir programas en los que los movimientos del robot estén supeditados a la toma de decisiones. El nombre de este SA representa la idea de que el alumno imagine la resolución de los desafíos desde la perspectiva del Robot. Se abandona la analogía con el control remoto y se propone concebir al Robot con ciertos sensores que le permiten percibir datos de su alrededor y con cierta autonomía en la cantidad de movimientos asociada a una batería. Esta batería se implementa a través de la instrucción repeat(n), n indica la carga expresada en la cantidad de veces que el robot ejecuta las instrucciones.

-“**Yo Robot Plus**” (SA 0.3): tiene como fin profundizar en el planteamiento de las condiciones a través de la utilización de operadores lógicos. Se incorpora además la

instrucción `repetirMientras()` que permite al robot ejecutar un conjunto de sentencias dependiendo de la evaluación de una condición. Esto permite al Robot alcanzar mayor autonomía al no depender de la estimación de un “n” en la instrucción `repeat()`.

- **3- Resultados**

Se describe a continuación a modo de ejemplo los resultados obtenidos en el desarrollo del SA “Yo robot” y la actividad registrada en la plataforma Edmodo.

- **3.1-Desarrollo del SA “Yo Robot”**

Para introducir a los alumnos en el desarrollo del SA se les planteó el desafío de tomar la baliza en tres mapas diferentes con una distancia del robot a la baliza distinta en cada escenario. En la Figura 1 se observan las 3 situaciones con las soluciones propuestas por los alumnos.

		
Tomar()	norte(1) Tomar()	norte (2) Tomar()

Figura 1. Primer desafío

El docente les planteó lo siguiente: Si el robot pudiera ver lo que se encuentra en la casilla al frente, ¿cómo se comportaría en cada situación para tomar la baliza?

A partir del aporte de los estudiantes se escribieron en la pizarra unas instrucciones que luego fueron codificadas (Figura 2):

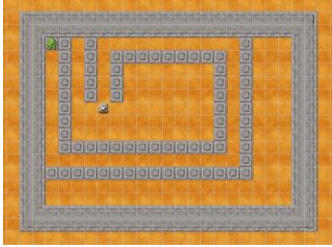
Primera Versión	Versión codificada
Si veobaliza entonces tomarla sino adelante	Si (frenteEsBaliza()){ tomar() } otro { adelante(1)}

Figura 2. Solución al desafío

A continuación el docente destacó la necesidad de ejecutar varias veces las instrucciones para poder resolver con el mismo código las 3 situaciones e introdujo la idea de batería implementada a través de la instrucción `repeat(n)`. Para profundizar en la comprensión del

comportamiento del Robot se analizaron entre todos diferentes alternativas surgidas de darle valores a “n”.

El siguiente desafío permite elaborar distintos algoritmos de acuerdo a las condiciones que se evalúen para el avance del robot:

<p>El robot debe llegar hasta el arbusto en el mapa “selec4”. ¿Qué otra condición podría usar el robot para recorrer el camino hasta el arbusto?</p>	
--	--

Se exponen a continuación 3 soluciones propuestas por los alumnos que permitieron realizar un análisis entre todos los miembros del grupo sobre las diferentes maneras de resolver un problema.

Solución 1	Solución 2	Solución 3
<pre> repetir(46){   si(frenteEsObstaculo() ){     derecha() adelante(1)   }otro{     adelante(1)   } } </pre>	<pre> repetir(46){   si(frenteEsClaro() ){     adelante(1)   }otro{     si(izquierdaEsObstaculo() ){       derecha()adelante(1)     }   } } </pre>	<pre> repetir(50){   si(frenteEsClaro()){     adelante(1)   }otro{     si(derechaEsClaro()){       derecha()     }   } } </pre>

Por último se plantea a los alumnos el siguiente problema que no es obligatorio sino que se propone a modo de un desafío para el que le interese resolverlo:

<p>El robot debe recorrer el laberinto para detenerse delante de la baliza.</p>	
---	--

A medida que los estudiantes iban terminando los desafíos anteriores comenzaron a resolver el problema del laberinto. En el siguiente encuentro la mayoría manifestó haber



intentado resolverlo pero sólo dos alumnos lograron obtener una solución y se compartieron con el resto:

Solución 1	Solución 2
<pre> repetir(){   si(frenteEsBaliza()){     }otro{       si(izquierdaEsClaro()){         izquierda() adelante(1)       }       otro{         derecha()       }     }   } } </pre>	<pre> izquierda() repetir(26){   si(frenteEsClaro()){     adelante(1)   }otro si(frenteEsObstaculo()){     si(izquierdaEsObstaculo()){       derecha()adelante(2)}     si(derechaEsObstaculo()){       izquierda()adelante(1)     }otro si(derechaEsClaro()){       derecha()adelante(1)}     si(frenteEsBaliza()){       }} </pre>

La puesta en común del análisis de los dos algoritmos permitió debatir acerca de las condiciones y el orden en el que las evaluaba el robot para avanzar.

- **3.2-Actividad en la Plataforma Edmodo**

La utilización de una plataforma virtual como apoyo a las clases fue una novedad para los alumnos. No contar con servicio de Internet en la escuela fue un obstáculo. Se explicó oralmente las características de Edmodo y por qué y para qué la íbamos a integrar pero los alumnos debieron comenzar su uso en sus domicilios. Su utilización en el desarrollo de la actividad en el aula hubiese favorecido la apropiación de la herramienta por parte de los alumnos.

De los 14 alumnos<sup>3</sup> del curso sólo 2 (M y N) no participaron en la plataforma, uno manifestó tener interrumpido el servicio de Internet y el otro olvidaba hacerlo.

En el primer trimestre se registraron en la plataforma mensajes de la profesora, publicando información y contestando inquietudes de los alumnos, y mensajes de los estudiantes, preguntando y respondiendo a la profesora y adjuntando problemas resueltos.

Se les preguntó a los estudiantes si habían descargado archivos y ellos manifestaron haber descargado el software, las actividades y ejercicios resueltos por otros compañeros:

...”Descargué para revisar y corregir un ejercicio”..., D.

...”Si, para hacer la tarea”..., B.

<sup>3</sup> Para preservar la identidad de los alumnos se los ha identificado con las letras de la A a la N.

..” Si he descargado Robomind y los trabajos prácticos”... , F.

En el Gráfico 1 se puede observar la cantidad de mensajes con información acerca del desarrollo de las clases que publicó la profesora y la cantidad de mensajes conteniendo preguntas y respuestas publicados por todos los miembros del grupo. De los 12 estudiantes registrados en la plataforma sólo 3 no publicaron mensajes de algún tipo. Sin embargo, uno de ellos manifestó ingresar a la plataforma para leer los mensajes.

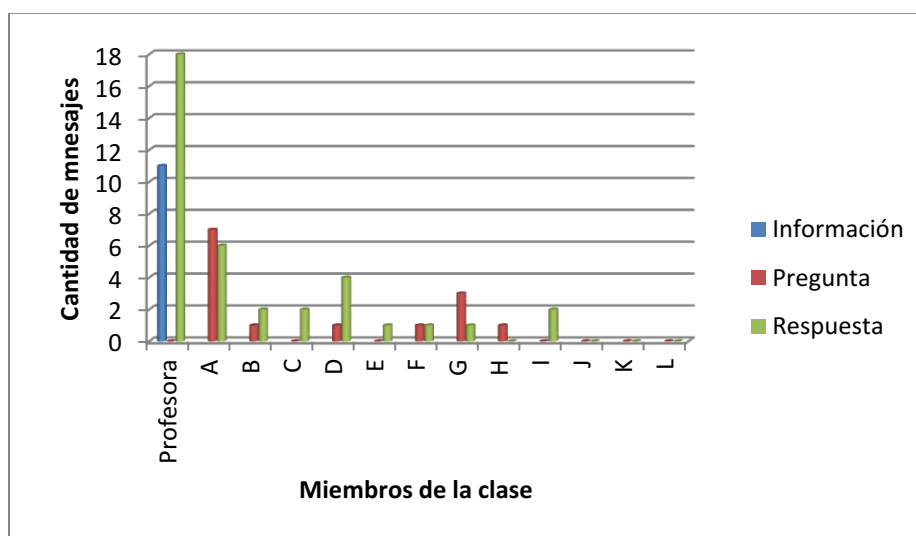


Gráfico 1. Actividad en la Plataforma

Los estudiantes expresaron ingresar a la plataforma una o dos veces por semana para ver las novedades y otros sólo si la profesora se los indicaba.

- **4-Comentarios finales**

El entorno basado en el modelo “Diseño de propuestas de e-formación colaborativa” en el marco de los lineamientos del modelo para diseñar Entornos de Aprendizaje Constructivista permitió plantear el diseño y la evaluación del ambiente desde una perspectiva constructivista (Castorina y col., 1999). Se caracterizó por: el planteo de problemas que requieren para su solución la construcción de nuevos conocimientos o de nuevas relaciones entre los ya elaborados, promover la discusión sobre los problemas planteados y alentar la formulación de conceptualizaciones necesarias para avanzar en el dominio de la programación. El planteo de los SA componentes permitió graduar la complejidad de los desafíos propuestos a los alumnos y ofrecer información y ejemplos en la plataforma. Esto posibilitó que los estudiantes pudieran avanzar en el aprendizaje de la



programación según su ritmo y disponer de los materiales de la clase así como tener una vía de comunicación para consultas con la profesora.

Se espera en el futuro contar con servicio de Internet en la Institución para integrar el uso de la plataforma en el aula. Así favorecer el desarrollo de competencias que les permitan a los alumnos la apropiación de herramientas para la enseñanza y aprendizaje virtual.

- **Bibliografía**

Castorina J., Ferreiro E., Kohl M., Lerner D. (1999). Piaget-Vigotsky: contribuciones para replantear el debate. Argentina: Paidós Educador.

Cenich, G. (2009). Tesis: Diseño de propuestas de e-formación colaborativa: un modelo desde la perspectiva de la Teoría de la Actividad. Magíster en Tecnología Informática Aplicada en Educación, Universidad Nacional de La Plata, La Plata.

Cenich, G. (2014). Una propuesta para la enseñanza de programación en la Escuela Secundaria. I Encuentro Internacional de Educación: espacios de investigación y divulgación, 1a ed. – Tandil, UNICEN. Disponible en <http://encuentroespaciosenblanco.unicen.edu.ar/inicio.xhtml> (Eje 3, Mesa 5)

Engeström, Y. (1987). Learning by expanding: An activity-theoretical approach to developmental research. Helsinki, Orienta-Konsultit.

Jonassen, D. (1997). Instructional Design Models for Well-Structure and Ill-Structure Problem-Solving Learning Outcomes. Educational Technology: Research and Development, 45 (1), 65-95.

Jonassen, D. (2000). El diseño de entornos constructivistas de aprendizaje. En C. Reigeluth (Eds), Diseño de la instrucción. Teorías y modelos. Madrid: Aula XXI Santillana, 225-249.

Lewis, R. (1998). Trabajo y aprendizaje en comunidades distribuidas. En C. Vizcarro y J. A. León (Eds), Nuevas tecnologías para el aprendizaje. Madrid: Pirámide, 191-219.

Selby, Cynthia (2011) Four approaches to teaching programming. In, Learning, Media and Technology: a doctoral research conference, London, UK.