

Acceptance and appropriation of the AeIE environment by teachers in training

Virginia Brassesco^{1,2}, Gonzalo Pablo Fernández^{3,1,2} and Yamila Ojeda²

¹ Departamento de Computación, Exactas, Universidad de Buenos Aires

² Departamento de Ciencia y Tecnología, Universidad Pedagógica Nacional

³ Departamento de Ciencia y Tecnología, Universidad Nacional de Quilmes

Abstract. It's been ten years since Resolution 263/15 of the Federal Education Council established in Argentina that "the teaching and learning of programming is of strategic importance in the National Education System during compulsory schooling, to strengthen the nation's socio-economic development". It's also been seven years since the approval of the Priority Learning Cores for Digital Education, Programming, and Robotics.

Since then, technology has taken on an even more prominent role in society, boosted primarily by the pandemic and the emergence of generative artificial intelligence systems. Still, we are far from achieving the stated goal. One of the main problems is the lack of qualified teaching staff.

We have made significant progress in the generation of teaching materials and the development of educational environments. A notable example is AeIE, a block-based programming environment for Arduino that stands out in multiple aspects, especially regarding its robustness and versatility. Many other Arduino programming environments simply use blocks as literal translations into the underlying Arduino language, disregarding the multitude of problems that can arise from invalid block combinations. AeIE not only detects and prevents such problems, but also assists the programmer in solving them and learning while doing so.

AeIE was tested on several occasions, used to teach programming to populations of different ages and sociocultural backgrounds. It was included as part of courses and workshops for high school students, teacher training programs, and in prison settings. In most cases, it was combined with other, simpler languages and environments, such as PilasBloques or Scratch, to teach the fundamentals of programming, and experience with Arduino was always left for a more advanced stage of the course due to the additional complications it can entail. In 2023, a programming course was held for high school students using AeIE as the only programming environment, showing its potential for teaching the basics of programming and not just for use at an advanced stage, after having learned the fundamentals with other environments.

In this paper, we present the results of an experience in which a course similar to the 2023 one was taught, using only AeIE, but this time for teachers in training. At the end of the course, acceptance of the environment was assessed using a survey based on the Technology Acceptance Model (TAM). Participants were subsequently contacted to inquire about the impact of both the training and their long-term use of the environment.

Keywords: technology acceptance, learning achievements, teaching resources, robotics, STEM

Aceptación y apropiación del entorno AelE por parte de docentes en formación

Resumen Se están cumpliendo diez años desde que la resolución 263/15 del Consejo Federal de Educación estableció en Argentina que “la enseñanza y el aprendizaje de la “Programación” es de importancia estratégica en el Sistema Educativo Nacional durante la escolaridad obligatoria, para fortalecer el desarrollo económico-social de la Nación”; y siete desde que se aprobaron los NAPs (Núcleos de Aprendizaje Prioritarios) para Educación Digital, Programación y Robótica.

Desde entonces, la tecnología ha ganado un papel aún más preponderante en la sociedad, potenciada principalmente por la pandemia y por la aparición de los sistemas de inteligencia artificial generativa. Aún así, seguimos estando lejos del objetivo planteado. Uno de los principales problemas es la falta de personal docente calificado.

Sí hemos avanzado mucho en la generación de material didáctico y el desarrollo de entornos educativos. Un ejemplo destacable es Arduino en la Escuela (AelE), un entorno de programación por bloques para Arduino que se destaca en múltiples aspectos, especialmente en lo que respecta a su robustez y versatilidad. Muchos de los otros entornos de programación para Arduino simplemente utilizan los bloques para hacer una traducción literal al lenguaje Arduino subyacente, sin tener en cuenta la multiplicidad de problemas que pueden generarse a partir de combinaciones inválidas de bloques. AelE no sólo detecta y evita tales problemas sino que asiste a quien está programando para que pueda resolverlos y además, que aprenda mientras lo hace.

AelE fue puesto a prueba en variadas ocasiones, utilizado para enseñar a programar a poblaciones de distintas edades y contextos socioculturales. Se incluyó como parte de cursos y talleres para estudiantes de secundaria, propuestas de formación docente y en contextos de encierro. En la mayoría de los casos se combinaba con otros lenguajes y entornos más simples, como PilasBloques o Scratch, para enseñar los fundamentos de la programación y la experiencia con Arduino siempre se dejaba para una etapa más avanzada de la cursada debido a las complicaciones adicionales que puede traer. En 2023 se realizó un curso de programación para estudiantes de nivel secundario utilizando AelE como único entorno de programación, demostrando su potencial para enseñar las bases de la programación y no sólo para ser usado en una etapa avanzada, después de haber aprendido los fundamentos en otros entornos.

En este trabajo presentamos los resultados de una experiencia en la que se dictó un curso similar al de 2023, utilizando únicamente AelE, pero esta vez para docentes en formación. Al finalizar la capacitación se evaluó la aceptación del entorno utilizando una encuesta basada en el Modelo de Aceptación de Tecnología (TAM, por sus siglas en inglés). Posteriormente se retomó el contacto con quienes habían participado de la misma para indagar acerca del impacto que tuvieron tanto la capacitación como el uso del entorno a largo plazo.

Palabras claves: aceptación de tecnología, logros de aprendizajes, recursos didácticos, robótica, STEM

Aceptación y apropiación del entorno AelE por parte de docentes en formación

1. Introducción

Se están cumpliendo diez años desde que la resolución 263/15 del Consejo Federal de Educación estableció en Argentina que “la enseñanza y el aprendizaje de la “Programación” es de importancia estratégica en el Sistema Educativo Nacional durante la escolaridad obligatoria, para fortalecer el desarrollo económico-social de la Nación” (de Educación, 2015); y siete desde que se aprobaron los NAPs (Núcleos de Aprendizaje Prioritarios) para Educación Digital, Programación y Robotica (de Educación, 2018).

Ya en esa época se hablaba de cómo la computación atravesaba prácticamente todos los ámbitos de la vida cotidiana. Teléfonos inteligentes, con supercomputadoras en la nube a través de internet, música e imágenes digitales, son solo algunos ejemplos del uso de las tecnologías en la vida cotidiana. Era indispensable educar a la población acerca de los riesgos de abusar de la tecnología. Hoy en día, todo eso parece haberse potenciado. Durante la pandemia dejó de ser exagerado decir que toda la interacción era mediada por la tecnología. Luego aparecieron los sistemas de Inteligencia Artificial generativa, que fueron adoptados inmediatamente por gran parte de la población.

A pesar de lo mucho que avanzó la tecnología y de lo mucho que se incorporó a la vida cotidiana de gran parte de la población, la educación en tecnologías digitales sigue presentando desafíos y la brecha digital de acceso a dichas tecnologías sigue vigente, esto lo podemos ver en los proyectos de desarrollo de los sistemas educativos del mundo, por tomar un ejemplo podemos citar el Plan Europeo (<https://education.ec.europa.eu/focus-topics/digital-education/action-plan>).

Una de las claves que favoreció el crecimiento de la brecha computacional es la falta de capacitación docente. Ya antes del boom de las tecnologías era difícil conseguir personal capacitado para enseñar computación. Como se menciona en la página 3 el informe de la fundación Sadosky del año 2013 (Sadosky, 2013), se generó un círculo vicioso en el que cada vez menos personas se forman para la docencia en informática, y las personas enseñando informática en las escuelas carecen de formación didáctica específica, generando en los estudiantes una visión poco atractiva de lo que es la informática en sí, llevando a su vez a que menos personas elijan informática como alternativa de estudio y por ende menos personas formadas.

En base al análisis que realizaron desde la Fundación Sadosky, se tomaron distintas medidas para romper dicho círculo vicioso. Por ejemplo, por medio de la iniciativa Program.ar (Bonello y Czemerinski, 2015), se ha estado desarrollando material y propuestas de formación, incluso desde antes de que se aprobaran las mencionadas resoluciones, con el objetivo de impulsar la inclusión de contenidos de programación y de ciencias de la computación en general en todos los niveles educativos.

En particular en la Universidad Pedagógica Nacional (UNPE), de la mano de la Fundación Sadosky, se creó la carrera de Profesorado en Informática, presencial, con sedes en la Ciudad de Buenos Aires y Pilar (provincia de Buenos Aires). Además, se realizó un ciclo de complementación curricular virtual para docentes en ejercicio de distintas zonas de la Argentina.

Una de las estrategias más utilizadas para acercar al público general al mundo de las Ciencias de la Computación es enseñar conceptos básicos de programación, sobre todo en los primeros niveles de la educación obligatoria. Gracias a la amplia difusión

que se le dio a las ideas de Wing sobre Pensamiento Computacional se extendió la idea de que era imperativo que todo el mundo aprendiera a programar, no sólo quienes se iban a dedicar profesionalmente a ello (Wing, 2006, 2011).

Por esa misma época apareció el entorno de programación Scratch, orientado a enseñar programación a niños a través de la creación de historias y videojuegos. Su principal característica es que la programación no se realiza escribiendo texto, como en la mayoría de los lenguajes de programación, sino conectando entre sí elementos arrastrables en la pantalla denominados bloques (Resnick et al., 2009). Si bien Scratch no fue el primer entorno de programación por bloques, fue el que más se popularizó e inició una época en la que proliferaron considerablemente los entornos de programación por bloques, entre los cuales se destaca PilasBloques, desarrollado por la Fundación Sadosky (Sanzo et al., 2017).

Por otro lado, también aparecieron variadas propuestas de enseñanza basadas en la robótica y la electrónica, lo que dio origen a la robótica educativa (Sullivan y Heffernan, 2016). Muchas de estas propuestas requieren un kit específico, generalmente privativo. Otras trabajan con placas de desarrollo genéricas, de hardware y software abierto, como Arduino. Estas propuestas fueron adoptadas por diferentes disciplinas, como física y biología para presentar propuestas didácticas STEM.

Al día de hoy existen numerosas alternativas que combinan las ventajas de la programación por bloques con las ventajas de la programación física. Entre ellas se destaca Arduino en la Escuela (Fernández y Cossio-Mercado, 2024b). Como muchos entornos de programación por bloques para Arduino, Arduino en la Escuela está diseñado para facilitar la enseñanza de la programación. Sin embargo, es superior al resto en múltiples aspectos, especialmente en lo que respecta a su robustez y versatilidad. Muchos de los otros entornos simplemente utilizan los bloques para hacer una traducción literal al lenguaje Arduino subyacente, sin tener en cuenta la multiplicidad de problemas que pueden generarse a partir de combinaciones inválidas de bloques. Arduino en la Escuela no sólo detecta y evita tales problemas sino que asiste a quien está programando para que pueda resolverlos y además, que aprenda mientras lo hace.

Arduino en la Escuela fue puesto a prueba en variadas ocasiones, utilizado para enseñar a programar a poblaciones de distintas edades y contextos socioculturales. Se incluyó como parte de cursos y talleres para estudiantes de secundaria, propuestas de formación docente y en contextos de encierro (Macario-Cabral et al., 2024). En la mayoría de los casos se combinaba con otros lenguajes y entornos más simples, como PilasBloques o Scratch, para enseñar los fundamentos de la programación y la experiencia con Arduino siempre se dejaba para una etapa más avanzada de la cursada debido a las complicaciones adicionales que puede traer. En 2023 se realizó un curso de programación para estudiantes de nivel secundario utilizando Arduino en la Escuela como único entorno de programación, demostrando su potencial para enseñar las bases de la programación y no sólo para ser usado en una etapa avanzada, después de haber aprendido los fundamentos en otros entornos (Fernández y Cossio-Mercado, 2024a).

Como su foco es la enseñanza de la programación y no de la electrónica o de otros conceptos más relacionados con el bajo nivel, fue diseñado para ocultar gran parte de los detalles de funcionamiento de las placas, permitiendo que se utilice para enseñar programación sin necesidad de contar con conocimientos previos en electrónica, o de

Aceptación y apropiación del entorno AeIE por parte de docentes en formación

tener que enseñarlos antes de poder empezar a trabajar con él. Su propuesta didáctica está en línea con el enfoque Gobstones (Martínez López et al., 2017) y el método Program.ar (Bonello y Czemerinski, 2015) que ponen el foco de la enseñanza en los conceptos y no en la escritura de código.

Este trabajo presenta los resultados de un estudio longitudinal que evaluó la implementación de Arduino en la Escuela (AeIE) como entorno educativo para docentes en formación del Profesorado en Informática. La investigación replicó y amplió una experiencia previa realizada en 2023, pero con dos innovaciones metodológicas clave: (1) el uso exclusivo de AeIE como plataforma de enseñanza, eliminando herramientas complementarias, y (2) el enfoque específico en futuros docentes como población objetivo.

El estudio adoptó el Modelo de Aceptación de Tecnología (TAM, por sus siglas en inglés) (Chuttur, 2009) como marco teórico para evaluar dos dimensiones centrales: la utilidad percibida y la facilidad de uso del entorno. Inmediatamente después de la capacitación, se aplicó una encuesta estandarizada basada en TAM a los participantes. Posteriormente, siguiendo un diseño de métodos mixtos, se realizó un seguimiento a los cuatro meses mediante una segunda encuesta y entrevistas semiestructuradas. Este enfoque permitió no solo medir la aceptación inicial del sistema, sino también explorar: (a) la retención de conocimientos, (b) la transferencia a la práctica docente, y (c) las barreras para la adopción sostenida.

La investigación buscó responder a tres preguntas clave:

- RQ1 ¿Cómo perciben los docentes en formación la utilidad didáctica de AeIE?
- RQ2 ¿Qué factores influyen en su adopción a largo plazo?
- RQ3 ¿En qué medida la experiencia modifica sus concepciones sobre la enseñanza de programación?

Los hallazgos aportan evidencia empírica relevante para el diseño de entornos educativos adaptados a la formación docente en tecnologías digitales.

2. Metodología

2.1. El taller

El taller se enmarcó en el proyecto de extensión “Domótica en la sociedad”, financiado con el Proyecto Convocatoria Específica 2022 “SIGAMOS ESTUDIANDO”, otorgado a la Universidad Pedagógica Nacional (UNIPe), que consistió en formar estudiantes de la universidad para que lleven a una escuela una propuesta de enseñanza con Arduino. Se realizó en dos encuentros de tres horas y media, con dos semanas de separación entre ellos (los días miércoles 25 de septiembre y miércoles 9 de octubre de 2024, de 13:00 a 16:30). El objetivo del taller era realizar actividades con Arduino, con estudiantes que ya contasen con un mínimo de conocimientos de programación, para que luego pudieran replicar estas actividades en un aula de secundaria. Se requirió para inscribirse tener conocimientos mínimos de programación y haber cursado y aprobado las prácticas de las materias del primer año de alguna de las siguientes carreras de la universidad: Profesorado en Informática, Profesorado en Física y Profesorado en

Matemática. Los profesorados de física y de matemática incluyen en su primer año la materia Taller de Informática. El profesorado de informática incluye en su primer año la materia Programación 1.

Se inscribieron 22 estudiantes de los profesorados de informática, física y matemática de la universidad, de los cuales 16 (la mayoría de informática) asistieron a ambos encuentros. Los encuentros fueron principalmente prácticos. Los estudiantes trabajaron en grupos de dos o tres personas en los proyectos propuestos por el equipo docente. Cada grupo contó con un kit que incluía una placa Arduino UNO, un protoboard de 170 puntos, varios leds, un buzzer, una matriz de leds, un servo, un sonar, un sensor de luminosidad, un joystick, un acelerómetro, un control remoto y varios cables. En la imagen de la figura 1 se muestra una foto del kit completo.

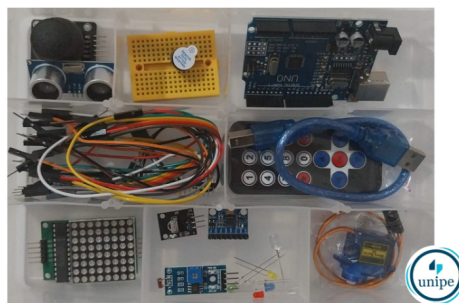


Figura 1: Foto del kit con el que trabajaron los grupos durante los dos encuentros del taller.

La dinámica del taller consistió en la resolución de distintas actividades propuestas por el equipo docente. Las consignas contaban con un objetivo común pero estaban abiertas a diferentes soluciones posibles, permitiendo que cada grupo resolviera el problema de manera distinta, implementara variantes o agregara comportamientos adicionales a elección. Una actividad en particular, por ejemplo, tenía como único objetivo crear una animación usando la matriz de leds, dejando total libertad a los grupos para que experimenten a gusto con qué mostrar y permitiendo la adición de agregados como un sonido que se reproduzca cada vez que cambia la imagen mostrada o la interacción con otros módulos para determinar la velocidad de reproducción o la siguiente imagen a mostrar. En este sentido, la propuesta didáctica se asemeja al esquema del aprendizaje basado en proyectos (ABP). Si bien no se trabajó exactamente con proyectos (en tanto las consignas eran definidas por el equipo docente), la dinámica de trabajo fue similar en tanto la resolución no se limitaba a implementar una solución de manera directa sino que requería primero delimitar el alcance de la misma y se contaba con libertad para modificar ciertos aspectos y agregar otros.

En cuanto al contenido, siendo que se contaba con sólo dos encuentros y que el principal objetivo no era enseñar programación sino motivar a estudiantes del profesorado a usar placas Arduino para enseñar en las escuelas, se partió asumiendo que los participantes ya tenían conocimientos mínimos de programación. Aún así, se aprovecharon

Aceptación y apropiación del entorno AeIE por parte de docentes en formación

las distintas actividades para repasar los elementos programáticos básicos (comandos, expresiones, procedimientos y funciones) así como la legibilidad y la división en sub-tareas. A pesar de haber asumido que este contenido ya era conocido, las puestas en común se usaron para explicarlos como si se estuvieran enseñando a un público no experto. De esta forma, las tuvieron en cuenta como modelos a replicar en el secundario. Además, se introdujeron los sensores y actuadores haciendo un paralelismo entre estos conceptos y los de expresiones y comandos, respectivamente.

2.2. Recolección de datos

Al finalizar el segundo encuentro, se les solicitó a los estudiantes que completaran una encuesta basada en el modelo TAM. Esta encuesta, estandarizada y diseñada para evaluar la percepción de usabilidad y utilidad de herramientas tecnológicas, constó de doce preguntas centradas en dos dimensiones principales: la facilidad de uso y la utilidad percibida de AeIE. Entre las afirmaciones evaluadas mediante una escala Likert (por ejemplo, desde “Muy en desacuerdo” hasta “Muy de acuerdo”) se incluyeron aspectos como la mejora en las habilidades de programación (“AeIE mejora mis habilidades de programación en Arduino”), la eficiencia en el aprendizaje (“AeIE me ayuda a resolver tareas más rápidamente”) y la claridad en la interacción (“La interacción con AeIE es flexible y comprensible”). Además, se indagó sobre la capacidad del sistema para ofrecer asistencia en tareas complejas (“AeIE me ofrece ayuda cuando encuentro una tarea difícil”) y su contribución al aprendizaje autónomo (“AeIE me permite organizar mi tiempo de aprendizaje”). Los resultados de esta encuesta permitieron medir no solo la aceptación inicial de la herramienta, sino también identificar áreas de mejora potenciales, como la curva de aprendizaje o la adaptabilidad a diferentes ritmos de trabajo.

Cuatro meses después del taller, se envió por correo electrónico una segunda encuesta a los participantes que habían respondido la primera, con el objetivo de evaluar el impacto a mediano plazo de la experiencia. Dado el tiempo transcurrido y la ausencia de incentivos directos para los encuestados, la tasa de respuesta fue significativamente menor en comparación con la encuesta inicial. El cuestionario se centró en determinar si los asistentes habían aplicado los conocimientos adquiridos, ya sea en proyectos personales, en su práctica docente o en planes futuros de implementación.

Para complementar y profundizar en las respuestas, se organizaron entrevistas posteriores, en las cuales también se indagó sobre el grado de retención de los contenidos del taller.

Encuesta de aceptación del entorno Como mencionamos anteriormente, esta encuesta se respondió de manera anónima. La primera parte incluyó únicamente algunos datos personales (edad, género, máximo nivel educativo alcanzado por los padres y área laboral en caso de que estuvieran trabajando). La segunda parte consistió en preguntas usando una escala Likert del 1 al 7 donde 1 significaba “Muy de acuerdo” y 7 “Muy en desacuerdo”. Las preguntas sobre las cuales tuvieron que expresar su nivel de acuerdo fueron las siguientes:

- Arduino en la Escuela mejora mis habilidades de programación en Arduino.

- Arduino en la Escuela me permite organizar mi tiempo de aprendizaje.
- Arduino en la Escuela me ayuda a resolver tareas más rápidamente.
- Arduino en la Escuela me ofrece ayuda cuando encuentro una tarea difícil.
- Arduino en la Escuela me ayuda a resolver tareas correctamente.
- Arduino en la Escuela es útil para aprender a programar en Arduino.
- Aprender a usar Arduino en la Escuela me resultó fácil.
- Es fácil conseguir que Arduino en la Escuela haga lo que quiero.
- La interacción con Arduino en la Escuela es clara y comprensible.
- La interacción con Arduino en la Escuela es flexible.
- Me resultó fácil adquirir destreza en el uso de Arduino en la Escuela.
- Creo que Arduino en la Escuela es fácil de usar.

Encuesta de impacto del taller En esta segunda encuesta se optó por recopilar las direcciones de correo electrónico de los encuestados para poder contactarlos para analizar en una entrevista personal algunas de las respuestas que nos parecían podían brindarnos más información sobre el uso de la herramienta. Los datos recopilados sí se usaron de forma anonimizada y agregada.

Para poder analizar las respuestas en base a sus conocimientos previos de programación y docencia, como la encuesta anterior era anónima, se repitieron las preguntas sobre datos personales y experiencia previa, tanto respecto a temas de programación como a temas de docencia. Luego se pidió que establezcan su nivel de acuerdo usando también una escala Likert, eligiendo entre “Totalmente en desacuerdo”, “En desacuerdo”, “Indeciso”, “De acuerdo” y “Totalmente de acuerdo”, con la pregunta “¿Sentís que haber asistido al taller te sirvió para ...”, finalizando la pregunta con cada una de las siguientes opciones:

- Aumentar tus conocimientos en programación?
- Aumentar tus conocimientos de Arduino?
- Revisar tus prácticas docentes?
- Querer usar Arduino para proyectos propios?
- Querer usar Arduino para enseñar programación?
- Querer usar Arduino para enseñar otros contenidos?

Finalmente, se preguntó cuáles de las siguientes opciones corresponden a cosas que hicieron o pensaron hacer después de haber asistido al taller:

- Me compré una placa Arduino (o tengo la intención de hacerlo)
- Seguí usando y explorando el entorno Arduino en la Escuela por mi cuenta
- Incorporé el uso de placas Arduino (o similar) a mis clases (o estoy planificando hacerlo)
- Incorporé el uso de Arduino en la Escuela a mis clases (o estoy planificando hacerlo)
- Armé un proyecto propio usando Arduino en la Escuela
- Diseñé actividades para enseñar usando Arduino en la Escuela
- Ninguna de las anteriores

Aceptación y apropiación del entorno AelE por parte de docentes en formación

3. Resultados

3.1. Aceptación del entorno

La encuesta fue completada por 12 asistentes (9 mujeres y 3 varones), lo que reveló un marcado desbalance de género en la participación. Este dato resulta llamativo, ya que en el Profesorado en Informática de la UNIFE la distribución suele ser más equilibrada, con una proporción cercana al 50 % para cada género. La edad de los participantes osciló entre 23 y 50 años, con una media de 35 años (DE = 8), lo que refleja un interés por la formación continua más allá de la etapa etaria. Cabe destacar que la mitad de los encuestados ya se desempeñaban como docentes al momento del taller, un fenómeno que podría asociarse a factores como la alta demanda laboral de profesores de informática en la zona de Pilar y los requisitos de inscripción (haber aprobado al menos una materia pedagógica y otra de programación). Esta combinación de condiciones, acceso inmediato al mercado laboral y formación especializada previa, podría explicar tanto el perfil de los asistentes como su motivación para participar.

	Promedio	STD
Creo que Arduino en la Escuela es fácil de usar.	1,75	1,76
Arduino en la Escuela me permite organizar mi tiempo de aprendizaje.	1,83	1,85
Arduino en la Escuela mejora mis habilidades de programación en Arduino.	1,92	2,15
Arduino en la Escuela es útil para aprender a programar en Arduino.	1,92	2,15
La interacción con Arduino en la Escuela es clara y comprensible.	1,92	1,83
La interacción con Arduino en la Escuela es flexible.	1,92	1,83
Aprender a usar Arduino en la Escuela me resultó fácil.	2	1,71
Arduino en la Escuela me ofrece ayuda cuando encuentro una tarea difícil.	2,17	1,75
Es fácil conseguir que Arduino en la Escuela haga lo que quiero.	2,17	1,85
Me resultó fácil adquirir destreza en el uso de Arduino en la Escuela.	2,17	1,7
Arduino en la Escuela me ayuda a resolver tareas más rápidamente.	2,25	2,09
Arduino en la Escuela me ayuda a resolver tareas correctamente.	2,42	2,15

Figura 2: Resultados de la encuesta de aceptación de AelE, junto al promedio y el desvío estándar de cada pregunta. Cada columna corresponde a las respuestas dadas por un participante. Tanto las filas como las columnas están ordenadas por puntaje promedio.

Encuesta de Aceptación En el gráfico de la figura 2 se muestran las respuestas dadas por cada participante a cada pregunta de la encuesta. Vemos que 5 participantes respondieron con la calificación máxima (uno, que corresponde a “Muy de acuerdo”) a todas

las preguntas. Hubo un participante que respondió con la calificación mínima (siete, que corresponde a “Muy en desacuerdo”) a todas las preguntas. Lamentablemente, como la encuesta fué anónima, no pudimos indagar en este aspecto. Si bien no descartamos que esta persona haya tenido una muy mala experiencia en el transcurso del taller, creemos que también es posible que haya malinterpretado la escala (asumiendo que siete era el máximo y uno el mínimo) y en realidad haya querido responder con la calificación máxima. Consideramos que daría menos lugar a confusión usar la escala invertida (uno mínimo y siete máximo) y para despejar dudas respecto a las respuestas, se podría agregar un campo de texto libre al querer replicar este tipo de encuestas.

3.2. Impacto del taller

El seguimiento realizado en una encuesta, cuatro meses después del taller, obtuvo una tasa de respuesta del 50 % (8 participantes de los 16 que asistieron a ambos encuentros). Si bien las respuestas recogidas fueron predominantemente positivas, estos resultados deben interpretarse con cautela debido a dos limitaciones principales: (1) el reducido tamaño muestral, y (2) el desconocimiento sobre los motivos de no respuesta (que podrían incluir desde falta de interés hasta dificultades técnicas para recibir o completar el instrumento). Cabe destacar que se realizaron intentos de contacto adicional a través de redes sociales para incrementar la participación, sin lograr obtener respuestas adicionales. Esta situación impide establecer conclusiones definitivas, ya que las respuestas ausentes podrían representar tanto experiencias negativas no registradas como simples limitaciones logísticas en el proceso de recolección de datos.

Encuesta de Impacto Si bien el reducido tamaño muestral ($n=8$) limita la posibilidad de realizar análisis estadísticos robustos, los datos cualitativos obtenidos permiten identificar algunos patrones relevantes.

En cuanto a la utilidad percibida del taller, las respuestas mostraron mayor consenso en las dimensiones relacionadas con el aprendizaje personal de Arduino (5 participantes “Totalmente de acuerdo” y 2 “De acuerdo”) que en aquellas vinculadas a su aplicación pedagógica (4 “Totalmente de acuerdo”, 2 “De acuerdo” y 1 “Indeciso”). Esta diferencia sugiere que, aunque el taller cumplió su objetivo principal de motivar el uso de Arduino para enseñar programación, el impacto fue más significativo en el desarrollo de competencias técnicas que en la reflexión sobre prácticas docentes.

Un hallazgo relevante es la correlación observada entre la experiencia docente previa y la valoración positiva del taller. Los participantes con experiencia en enseñanza de programación mostraron mayor acuerdo con las afirmaciones sobre utilidad pedagógica, mientras que aquellos sin dicha experiencia tendieron a valorar principalmente los aspectos técnicos. Esto podría indicar que la transferencia al ámbito educativo requiere de un bagaje pedagógico previo que no todos los participantes poseían.

Las dos participantes con experiencia en enseñanza de programación mostraron los niveles más altos de acuerdo, aunque con una diferencia significativa:

- La participante que había usado Arduino previamente en sus clases estuvo “Totalmente de acuerdo” con todas las dimensiones excepto en “motivar su uso para enseñar programación” (donde seleccionó “De acuerdo”). Este matiz podría indicar

Aceptación y apropiación del entorno AeE por parte de docentes en formación

un efecto techo (la herramienta no podía motivar lo que ya estaba implementado) o, alternativamente, que su experiencia le permitió identificar limitaciones no evidentes para docentes noveles.

- La participante sin experiencia previa con Arduino pero con práctica docente mostró acuerdo unánime (“Totalmente de acuerdo” en todos los ítems), lo que refuerza la hipótesis de que el taller resulta particularmente valioso para docentes en ejercicio, independientemente de su familiaridad técnica con la plataforma.

Respecto a las intenciones de aplicación posterior al taller, los resultados fueron menos alentadores. Solo 2 de 8 participantes manifestaron planes concretos de incorporar Arduino o Arduino en la Escuela en sus prácticas docentes. No obstante, el interés por continuar explorando la herramienta de manera autónoma (manifestado por 5 participantes) y la intención de adquirir placas (6 participantes, incluyendo 2 que ya las poseían) sugieren que, aunque la transferencia inmediata al aula fue limitada, el taller logró despertar un interés sostenido en la tecnología.

Entrevistas personales Las dos participantes que pudimos entrevistar eran ambas de la carrera del Profesorado en Informática, una de ellas recién había aprobado Programación 1, y estaba en su segundo cuatrimestre de cursada, y la otra era una estudiante más avanzada que ya había dado clases de programación. Los testimonios de las dos participantes entrevistados revelan percepciones contrastantes pero complementarias sobre la experiencia con Arduino en la Escuela:

1. Percepción de usabilidad

- La participante con experiencia previa en programación (Entrevista 1) valoró especialmente la transición desde el IDE tradicional, destacando que AeE resulta “más profesional” que Scratch o Pilas Bloques pero igualmente accesible: “es clara” y “mucho más cómoda para enseñar”.
- La participante sin experiencia (Entrevista 2), aunque inicialmente escéptica, reconoció que AeE le resultó “mucho más fácil” que TinkerCAD y que los bloques le facilitaron posteriormente el aprendizaje de C++.

2. Transferencia pedagógica

- Ambas participantes demostraron comprensión conceptual al explicar soluciones mediante estrategias (no sintaxis), destacando:
 - Importancia de la legibilidad del código (“buenos nombres”, “organizar bien el código”).
 - Proceso iterativo de mejora (de solución funcional a solución óptima).
- Coincidieron en recomendar una progresión didáctica: iniciar con PilasBloques/Scratch en primaria y transitar a AeE en secundaria.

3. Impacto diferencial

- La participante experimentada manifestó intención inmediata de implementación en aula y observó potencial interdisciplinario.
- La participante novel, pese a su evaluación inicial negativa, reconoció que el taller le permitió:
 - Comprender componentes electrónicos (sensores/actuadores).
 - Utilizar TinkerCAD.

- Sentar bases para aprender C++.

4. Recomendaciones emergentes

- Ambas reconocieron la necesidad de andamiaje docente, enfatizando el rol de guía del profesor.
- La participante con experiencia sugirió una adecuación por nivel educativo, recomendando AeIE más para estudiantes de segundo ciclo (más de 12 años).
- La participante sin experiencia destacó particularmente el valor formativo del enfoque por bloques incluso para estudiantes avanzados.

Estos resultados sugieren que AeIE puede servir como puente cognitivo entre entornos visuales (PilasBloques) y textuales (IDE Arduino). Además vemos que AeIE se puede usar para desarrollar pensamiento computacional más allá de la sintaxis y es un recurso versátil adaptable a distintos niveles de experiencia. De todas formas, recordemos que dado que solo hicimos dos entrevistas, estos hallazgos representan sólo dos casos extremos (experto/novel) y requieren validación con muestras más amplias.

4. Limitaciones

Este trabajo presenta varias limitaciones que deben considerarse al interpretar sus resultados. En primer lugar, de las 22 personas que asistieron al taller inicial, solo 12 completaron la encuesta TAM inmediatamente después de la actividad, y apenas 8 respondieron el seguimiento realizado cuatro meses después. Además, las entrevistas en profundidad se llevaron a cabo solamente con 2 participantes. Esta reducida muestra limita significativamente la generalización de los hallazgos, aunque los datos cualitativos obtenidos resultaron valiosos para explorar patrones iniciales de uso y apropiación de la herramienta.

Otra limitación importante se relaciona con el perfil de los participantes. A pesar de que el taller se realizó en la sede Pilar con la intención de convocar a estudiantes de diversas carreras, la mayoría de los asistentes pertenecían al Profesorado en Informática. Este sesgo en la muestra implica que los participantes tenían: (1) un interés particular en la herramienta como recurso didáctico, y (2) conocimientos previos de programación probablemente superiores a los de estudiantes de otras disciplinas. Estas características específicas del grupo podrían afectar la transferibilidad de los resultados a poblaciones con diferente formación académica o motivaciones de aprendizaje.

Estas limitaciones sugieren la necesidad de futuras investigaciones con muestras más amplias y diversas que permitan validar los hallazgos preliminares aquí presentados.

5. Conclusiones

Si bien el reducido tamaño muestral limita la posibilidad de realizar análisis estadísticos robustos, los datos cualitativos obtenidos permiten identificar algunos patrones relevantes. Estos hallazgos tienen importantes implicancias para el diseño de futuras intervenciones:

Aceptación y apropiación del entorno AelE por parte de docentes en formación

- La efectividad del taller parece estar moderada por dos variables clave: (a) experiencia docente previa y (b) conocimiento específico de Arduino.
- La transferibilidad a otras disciplinas STEM podría requerir: (a) módulos específicamente diseñados para cada área o (b) participantes con experiencia docente en esas disciplinas.
- La “experiencia docente” emerge como variable más determinante que la “experiencia técnica” en la valoración pedagógica del taller.

Respecto a las entrevistas realizadas, los resultados sugieren que AelE puede servir como puente cognitivo entre entornos visuales (PilasBloques) y textuales (IDE Arduino). Además vemos que AelE puede usarse para desarrollar pensamiento computacional más allá de la sintaxis y es un recurso versátil adaptable a distintos niveles de experiencia.

Referencias

- Bonello, M. B., & Czemerinski, H. (2015). Program.AR: una propuesta para incorporar Ciencias de la Computación a la escuela argentina.
- Chuttur, M. Y. (2009). Overview of the Technology Acceptance Model: Origins, Developments and Future Directions. *Sprouts: Working Papers on Information Systems*, 9(37).
- de Educación, C. F. (2015). Resolución CFE Nro. 263/15. http://www.bnm.me.gov.ar/giga1/normas/RCFE_263-15.pdf
- de Educación, C. F. (2018). Núcleos de Aprendizajes Prioritarios para Educación Digital, Programación y Robótica. *Anexo I de la Resolución CFE Nro. 343/18*. http://www.bnm.me.gov.ar/giga1/normas/RCFE_343-18.pdf
- Fernández, G. P., & Cossio-Mercado, C. (2024a). ¿Es posible enseñar los fundamentos de la programación sólo con Arduino?: Análisis de un curso introductorio para estudiantes de secundaria usando bloques. *Jornadas Argentinas de Didáctica de las Ciencias de la Computación 2024*.
- Fernández, G. P., & Cossio-Mercado, C. (2024b). AelE: a versatile tool for teaching programming and robotics using Arduino. *50a Conferencia Latinoamericana de Informática (CLEI)*. <https://doi.org/10.1109/CLEI64178.2024.10700288>
- Macario-Cabral, D., Cossio-Mercado, C., & Fernández, G. P. (2024). Iniciación a la programación en contextos de encierro con Arduino. *Jornadas Argentinas de Didáctica de las Ciencias de la Computación 2024*.
- Martínez López, P. E., Ciolek, D., Arévalo, G., & Pari, D. (2017). The Gobstones method for teaching computer programming. *XXV Simposio de Educación Superior en Computación (SIESC'17), dentro de la XLIII Conferencia Latinoamericana de Informática (CLEI'17)*, 1-9.
- Resnick, M., Maloney, J. H., Monroy Hernández, A., Rusk, N., Eastmond, E., Brennan, K., Millner, A., Rosenbaum, E., Silver, J., Silverman, B., & Kafai, Y. (2009). Scratch: Programming for all. *Communications of the ACM*, 52(11), 60-67. <https://doi.org/10.1145/1592761.1592779>

- Sadosky, F. (2013). Una propuesta para refundar la enseñanza de la computación en las escuelas argentinas. <https://fundacionsadosky.org.ar/wp-content/uploads/2023/03/cc-2016.pdf>
- Sanzo, A., Schapachnik, F., Factorovich, P., & Sawady O'Connor, F. (2017). Pilas Bloques: A scenario-based children learning platform. *2017 Twelfth Latin American Conference on Learning Technologies*, 1-6.
- Sullivan, F. R., & Heffernan, J. (2016). Robotic Construction Kits as Computational Manipulatives for Learning in the STEM Disciplines. En *Journal of Research on Technology in Education* (pp. 105-128, Vol. 48). Routledge. <https://doi.org/10.1080/15391523.2016.1146563>
- Wing, J. M. (2006). Computational Thinking. *Communications of the ACM*, 49(3), 33-35. <https://doi.org/10.1145/1118178.1118215>
- Wing, J. M. (2011). Research Notebook: Computational Thinking - What and Why? *The link Magazine*, 6, 20-23.