

Development of technical heuristics for situated analysis of tools for teaching programming at the secondary level

Maximiliano Diaz¹ and Gisela Confalonieri²

¹ Universidad Nacional de Quilmes, Bernal, Buenos Aires, Argentina

² Departamento de Computación, Facultad de Ciencias Exactas y Naturales,
Universidad de Buenos Aires, Ciudad Autónoma de Buenos Aires, Argentina
`maximiliano.diaz@unq.edu.ar`, `gconfalonieri@dc.uba.ar`

Abstract. This paper presents the progress of an ongoing undergraduate thesis in computer science, focused on developing technical heuristics to evaluate programming teaching tools in technical secondary schools in the southern suburbs of Buenos Aires Province (PBA). The research stems from the lack of technical evidence regarding the actual performance of these platforms under common school conditions, such as the use of limited hardware, unstable networks, and a variety of operating systems. To date, a review of official documents from PBA and the City of Buenos Aires (CABA) has been conducted, from which an initial set of used and recommended tools has been identified. After applying criteria of institutional relevance, technical accessibility, and methodological diversity, Scratch, Pilas Bloques, and PSeInt were selected for in-depth analysis. A methodological framework was also designed through a literature review on heuristic evaluation, defining five dimensions of analysis: accessibility, installation, portability, interface, and support community. Data collection instruments and a mathematical decision rule were also designed to classify the tools as recommended, conditionally recommended, or not recommended. The field application of the proposed heuristics is currently underway, so this paper does not present results.

Keywords: heuristic evaluation, programming tools, evaluation instrument, technical secondary schools, technical-operational feasibility

Desarrollo de heurísticas técnicas para el análisis situado de herramientas para la enseñanza de la programación en el nivel medio

Resumen Este trabajo presenta los avances de una tesis de licenciatura informática en progreso, enfocada en el desarrollo de heurísticas técnicas

para evaluar herramientas de enseñanza de la programación en escuelas secundarias técnicas del conurbano sur de la Provincia de Buenos Aires (PBA). La investigación surge ante la falta de evidencia técnica situada sobre el comportamiento real de las plataformas en condiciones escolares comunes, como el uso de hardware limitado, redes inestables, o diversidad de sistemas operativos. A la fecha, se realizó un relevamiento de documentos oficiales de PBA y CABA, a partir del cual se identificó un conjunto inicial de herramientas utilizadas y recomendadas. Tras aplicar criterios de pertinencia institucional, accesibilidad técnica y diversidad metodológica, se seleccionaron Scratch, Pilas Bloques y PSeInt para el análisis en profundidad. A su vez, se diseñó un marco metodológico mediante una revisión bibliográfica sobre evaluación heurística, definiendo cinco dimensiones de análisis: accesibilidad, instalación, portabilidad, interfaz y comunidad de soporte. Se diseñaron también los instrumentos de recolección de datos y una regla matemática de decisión para clasificar las herramientas como recomendables, condicionales, o no recomendables. La aplicación de campo de las heurísticas planteadas se encuentra en curso, por lo que este trabajo no presenta resultados.

Keywords: enseñanza de la programación, heurística técnica, software educativo, factibilidad técnica, educación media

1. Introducción y motivación

La incorporación de herramientas de programación en escuelas secundarias técnicas plantea un problema de selección que no puede resolverse únicamente a partir de la disponibilidad o popularidad de los entornos. Revisiones de literatura recientes (Gonçalves et al., 2025) muestran un predominio de estudios centrados en dimensiones pedagógicas por sobre las restricciones técnicas y contextuales. No se identificaron, en el contexto argentino, trabajos recientes que aborden de manera directa una evaluación técnico-comparativa de herramientas para la enseñanza de la programación en escuelas secundarias. En estos contextos, la factibilidad de uso depende de condiciones concretas como la instalación, la compatibilidad con el equipamiento disponible, la necesidad de conectividad, la estabilidad de funcionamiento, la facilidad de despliegue en laboratorio y la disponibilidad de soporte.

Una herramienta podría resultar adecuada en términos generales, pero presentar obstáculos relevantes al momento de su implementación efectiva en instituciones educativas con heterogeneidad de recursos tecnológicos y limitaciones de infraestructura. Estas limitaciones adquieren especial importancia en escuelas secundarias técnicas con trayectos afines a la programación, donde la incorporación de herramientas de software constituye un recurso crítico. En este tipo de instituciones, la posibilidad de instalar, ejecutar y sostener técnicamente una herramienta dentro del laboratorio es un aspecto decisivo para su uso efectivo.

El presente trabajo se inscribe en una investigación cuyo objetivo es desarrollar un instrumento de evaluación heurística para la valoración técnico-operativa de herramientas de programación en escuelas secundarias técnicas del conurbano sur de la Provincia de Buenos Aires. Dado que la investigación se encuentra en curso, este artículo no presenta resultados definitivos, enfocándose en describir el proceso de construcción del instrumento, la selección de herramientas y la estrategia prevista para su aplicación en campo.

El resto del artículo se organiza como sigue: la sección 2 presenta la selección del corpus de herramientas y criterios de selección, la sección 3 presenta la fundamentación de las heurísticas y el modelo de evaluación propuesto, y la sección 4 sintetiza los avances logrados y las fases pendientes.

2. Relevamiento y selección de herramientas

Para identificar herramientas de programación con presencia explícita en diseños curriculares y materiales institucionales, se realizó un relevamiento de documentación normativa de la Provincia de Buenos Aires, como los diseños de las tecnicaturas en Programación y en Electromecánica (Dirección General de Cultura y Educación [DGCyE], 2018, 2024). También se consultaron tutoriales oficiales de la Ciudad Autónoma de Buenos Aires (Ministerio de Educación, s.f.) para reforzar la comparabilidad del análisis.

Este relevamiento identificó un conjunto inicial de herramientas, sobre el cual se realizó una valoración preliminar basada en las dimensiones de accesibilidad, instalación, portabilidad, interfaz y comunidad de soporte, registrando la presencia o ausencia de condiciones consideradas relevantes para su posible adopción en contextos escolares (Cuadro 1). Como resultado, se obtuvo un puntaje total por herramienta, entendido como una aproximación comparativa inicial para ordenar el universo relevado bajo un mismo criterio de factibilidad técnica. Para el análisis en profundidad, se seleccionaron las tres herramientas que muestran la mayor cantidad de condiciones favorables para su implementación en escuelas: Scratch³, Pilas Bloques⁴ y PSeInt⁵.

3. Marco metodológico y propuesta técnica

3.1. Fundamentación de las heurísticas

A partir de una revisión bibliográfica, se tomó como base metodológica a la evaluación heurística definida como un método de inspección aplicable por un número reducido de evaluadores expertos. Este enfoque demuestra que un número reducido de evaluadores basta para identificar una proporción significativa de problemas de un sistema, mientras que el aumento excesivo de evaluadores

³ <https://scratch.mit.edu/>

⁴ <https://pilasbloques.program.ar/>

⁵ <https://pseint.sourceforge.net/>

Dimensión	Categoría	Indicador	App Inven.	Light Bot	Pilas Bloq.	Proces sing	PSeInt	Scratch	Thunkable
Accesibilidad	Acceso	Sin registro	no	sí	sí	sí	sí	sí	no
Accesibilidad	Licencia	Gratuito	sí	no	sí	sí	sí	sí	sí
Accesibilidad	Licencia	Cod.Abierto	sí	no	sí	sí	sí	sí	no
Instalación	Instal. local	Sin instalar	sí	sí	sí	no	no	sí	sí
Portabilidad	Conectividad	Uso offline	no	sí	sí	sí	sí	sí	no
Portabilidad	SO	Compatible Windows	sí	sí	sí	sí	sí	sí	sí
Portabilidad	SO	Compatible Linux	sí	sí	sí	sí	sí	sí	sí
Portabilidad	SO	Compatible Mac OS	sí	sí	sí	sí	sí	sí	sí
Interfaz	Idioma	Interfaz en español	sí	sí	sí	no	sí	sí	no
Soporte	Document.	Document. en español	sí	no	sí	sí	sí	sí	no
Soporte	Comunidad	Comunidad activa	sí	no	sí	sí	sí	sí	sí
Total:			9	7	11	9	10	11	6

Cuadro 1. Comparación preliminar de herramientas según criterios de selección.

tiende a producir beneficios decrecientes (Nielsen & Molich, 1990). Esto resulta especialmente relevante para el contexto escolar considerado, ya que permite identificar problemas significativos de uso e implementación con un costo acotado, sin depender de grandes muestras ni de dispositivos complejos.

Sin embargo, las heurísticas generales de usabilidad no alcanzan por sí solas para evaluar entornos de programación, ya que estos presentan problemas específicos vinculados con la comprensión de la lógica del sistema, la claridad de los errores de principiantes, y la visibilidad del estado de ejecución (Kölling & McKay, 2016). Existen antecedentes útiles para considerar dificultades propias del aprendizaje y uso de lenguajes de programación (Pane & Myers, 1996), así como aportes que refuerzan la importancia de evaluar criterios de estabilidad, compatibilidad y funcionamiento seguro (Giang et al., 2019), y la adaptación de la evaluación heurística al tipo de software analizado (Brayshaw et al., 2014).

Sobre esa base, la propuesta organiza los aportes en tres decisiones de diseño: adoptar la evaluación heurística como estrategia experta y de bajo costo para identificar problemas de uso y de implementación técnica, adaptar los criterios clásicos de usabilidad a rasgos propios de las herramientas de programación, e incorporar la factibilidad técnico-operativa (instalación, compatibilidad, conectividad, etc.) como una dimensión crítica del contexto escolar real.

3.2. Diseño del instrumental y métricas de decisión técnica

El modelo propuesto se estructura en cinco dimensiones: accesibilidad, instalación, portabilidad, interfaz y comunidad de soporte. Cada dimensión se desagrega en indicadores observables, puntuables en una escala de 0 a 4, permitiendo valorar la viabilidad técnica de una herramienta en condiciones escolares reales.

En términos operativos, la aplicación prevista se apoya en un conjunto de instrumentos complementarios: una hoja de contexto institucional para relevar

la infraestructura (RAM, SO, conectividad, permisos) de cada escuela, una guía de tareas para estandarizar la interacción con la herramienta (instalación, primer uso, guardado) y asegurar comparabilidad entre pruebas, una planilla de evaluación heurística que registra las valoraciones por dimensión e indicador, un registro de problemas para documentar incidentes observados durante la interacción, y un cuestionario de cierre que consolida las valoraciones obtenidas.

A su vez, se definió una regla matemática de decisión para sistematizar la recomendación. En primer lugar, se calcula el promedio por dimensión (μ_d) considerando sólo los ítems efectivamente evaluados (A_d):

$$\mu_d = \frac{1}{|A_d|} \sum_{i \in A_d} x_i$$

Estos promedios se integran en un puntaje global ponderado (S), donde se asignan pesos (w) según la relevancia para el contexto técnico, priorizando las dimensiones de portabilidad e instalación.

$$S = \frac{\sum_{d \in D^*} w_d \mu_d}{\sum_{d \in D^*} w_d}$$

siendo D^* el conjunto de dimensiones con al menos un indicador aplicable y efectivamente evaluado. Finalmente, se incorpora una medida de cobertura (c) para distinguir evaluaciones sólidas de parciales, calculada como la proporción entre ítems evaluados y aplicables. Además, la regla integra un conjunto de ítems críticos de factibilidad mínima (m).

Una herramienta se clasificará como No recomendable si presenta fallas críticas ($m \leq 1$) o promedios bajos en dimensiones base ($\mu_{1,2} < 2$), mientras que la categoría Recomendable exige un desempeño alto ($S \geq 3,2$), cobertura suficiente ($c \geq 0,70$), ausencia de bloqueos críticos ($m \geq 2$) y promedios razonables en dimensiones base ($\mu_{1,2} \geq 2,5$). Los demás casos se considerarán Recomendables condicional. Así, la propuesta evita que puntajes altos en una dimensión compensen barreras estructurales de implementación en la escuela.

4. Estado actual y trabajo futuro

En la etapa actual del trabajo, se consolidó un marco metodológico y un instrumental de evaluación heurística orientado al análisis técnico-operativo de herramientas de programación en escuelas secundarias técnicas. Se definieron las dimensiones de análisis, se diseñaron los instrumentos de recolección y se formalizó una regla matemática de decisión que permite traducir observaciones situadas en recomendaciones técnicas comparables.

La investigación se encuentra aún en desarrollo, y los avances presentados constituyen el soporte estructural para la fase de aplicación. El trabajo futuro se organiza en tres líneas de acción: aplicación en campo, análisis de sensibilidad y ajuste, y producción de guías de adopción. La aplicación en campo contempla la evaluación de Scratch, Pilas Bloques y PSeInt en tres instituciones del

conurbano sur de la Provincia de Buenos Aires (Avellaneda, Quilmes y Berazategui), con la participación de un referente evaluador por institución. Esta muestra de tres evaluadores responde a los límites temporales de una tesis de grado y es consistente con el enfoque mencionado en la sección 3.1. A partir de los datos recolectados, se realizará un ajuste del instrumento orientado a revisar la pertinencia de los indicadores, el peso relativo de las dimensiones y el comportamiento de la regla de decisión frente a situaciones reales de aplicación. El resultado final no sólo será un ranking comparativo, sino también un conjunto de recomendaciones técnicas situadas, destinadas a identificar en qué contextos una herramienta resulta recomendable, bajo qué condiciones podría adoptarse y qué restricciones deberían considerarse para su implementación.

Referencias

- Brayshaw, M., Gordon, N., Nganji, J., Wen, L., & Butterfield, A. (2014). Investigating heuristic evaluation as a methodology for evaluating pedagogical software: an analysis employing three case studies. *International Conference on Learning and Collaboration Technologies*, 25-35.
- Dirección General de Cultura y Educación. (2018). *Diseño curricular: Técnico en Programación*. Gobierno de la Provincia de Buenos Aires. Consultado el 26 de marzo de 2026, desde <https://shorturl.at/jzt2D>
- Dirección General de Cultura y Educación. (2024). *Diseño curricular para la educación secundaria técnico profesional (ETP): Técnico en Electromecánica*. Gobierno de la Provincia de Buenos Aires. Consultado el 27 de marzo de 2026, desde <https://shorturl.at/JWxXr>
- Giang, C., Piatti, A., & Mondada, F. (2019). Heuristics for the development and evaluation of educational robotics systems. *IEEE Transactions on Education*, 62(4), 278-287.
- Gonçalves, S. C. L., Moreira, R., Backes, A. R., Rodrigues Moreira, L. F., & Martinhago, A. Z. (2025). Programming in brazilian higher education and high school: A systematic literature review. En *Latin American Conference on Learning Technologies* (pp. 287-303).
- Kölling, M., & McKay, F. (2016). Heuristic evaluation for novice programming systems. *ACM Transactions on Computing Education (TOCE)*, 16(3), 1-30.
- Ministerio de Educación. (s.f.). Tutoriales sobre Gestión de Contenidos Multimedia - Biblioteca Digital. *Gobierno de la Ciudad de Buenos Aires*. Consultado el 27 de marzo de 2026, desde <https://shorturl.at/Zg0xf>
- Nielsen, J., & Molich, R. (1990). Heuristic evaluation of user interfaces. *Proceedings of the SIGCHI conference on Human factors in computing systems*, 249-256.
- Pane, J. F., & Myers, B. A. (1996). Usability issues in the design of novice programming systems.