

El LOW-CODE Potenciando la Transformación Digital de los Procesos en Instituciones de Educación Superior

L. Ramos Corchado^{#1}, M. Díaz Rodríguez^{#2}, A. Aguilar Cornejo^{#3}

[#]Tecnológico Nacional de México: Instituto Tecnológico José Mario Molina Pasquel y Henríquez, Unidad Académica Zapopan, Departamento de Investigación y Posgrado, Jalisco, México

¹leon.ramos@zapopan.tecmm.edu.mx, ²miriam.diaz@zapopan.tecmm.edu.mx,

³alejandro.aguilar@zapopan.tecmm.edu.mx

Resumen— El desarrollo de bajo código (LOW-CODE) está transformando la digitalización en las Instituciones de Educación Superior (IES), permitiendo a empleados sin habilidades avanzadas de programación desarrollar soluciones digitales personalizadas. Aunque el LOW-CODE ha ganado impulso en diversos sectores, existe una brecha en la investigación sobre su potencial específico en las IES. El presente aborda esta brecha mediante un enfoque de investigación híbrido que combina revisión de literatura, estudio de caso. Los resultados revelan que el LOW-CODE tiene un potencial significativo para habilitar la digitalización e innovación en procesos específicos de las IES, como gestión de matrículas, recursos humanos, finanzas y sistemas de información estudiantil. El estudio subraya la importancia de un enfoque estratégico para aprovechar el LOW-CODE, incluyendo capacitación del personal y promoción de una cultura de innovación digital. Se identifican áreas para futuras investigaciones, como las sinergias entre LOW-CODE y temas relevantes para las IES: marcos de trabajo Ágiles, analítica de datos educativos, inteligencia artificial y personalización del aprendizaje.

Palabras clave— Low-code, IES, Ágil, Transformación Digital, Procesos, Citizen Developers.

I. INTRODUCCIÓN

El uso de tecnologías emergentes ofrece a las IES oportunidades para optimizar procesos y aumentar la productividad. Un nuevo paradigma en esta área es el desarrollo de bajo código (low-code). Aunque no existe una definición unánime [1,2], el concepto de bajo código puede describirse como una tecnología "que apoya el desarrollo rápido de aplicaciones orientadas al cliente, requiriendo un mínimo de codificación manual y permitiendo nuevas prácticas de desarrollo productivo" según Richardson [3]. Se acuñó este término como respuesta a entornos empresariales cambiantes y la necesidad de un desarrollo de software más rápido y económico [4,5].

El bajo código permite a individuos con habilidades de codificación limitadas desarrollar soluciones digitales para necesidades empresariales específicas. Reduce las barreras técnicas, permitiendo que personas cercanas a las operaciones implementen soluciones digitales. Estos individuos son conocidos como "Citizen Developers" [6-9]. En la práctica, utilizan interfaces gráficas intuitivas con componentes predefinidos para desarrollar aplicaciones,

generando código automáticamente en segundo plano. Así, usuarios sin habilidades avanzadas de codificación pueden crear aplicaciones funcionales sin necesidad de desarrolladores de software [10].

Algunos autores consideran el bajo código como una evolución de la Ingeniería de Software Asistida por Computadora (CASE) [1] o como parte de la Ingeniería Dirigida por Modelos (MDE) [11]. Otros argumentan que su popularidad se debe al rebranding de conceptos existentes [12].

A pesar del debate, el bajo código está ganando impulso. Se pronostica que para 2025, el 70% de las aplicaciones empresariales se desarrollarán con bajo código [13] y más del 90% de las organizaciones lo utilizarán o considerarán [14]. Se estima que para 2023, habrá cuatro veces más Citizen Developers que desarrolladores profesionales [15]. El Project Management Institute (PMI) ha lanzado programas educativos para capacitar en esta tecnología [8], y han surgido proyectos de investigación para preparar ingenieros en su uso [16]. El bajo código ya se ha explorado en diversos escenarios empresariales [2,5,17,18]. Sin embargo, no existe investigación sobre sus implicaciones en las IES. Este estudio pretende llenar ese vacío explorando los beneficios y limitaciones de esta tecnología para la digitalización en las IES.

II. ESTADO DEL ARTE

En la revisión de literatura se identificaron dos líneas principales. La primera, una caracterización general de los principales temas de investigación sobre bajo código, enfocándose en aplicaciones prácticas del bajo código en escenarios empresariales operativos. La segunda, contextualizar los beneficios y limitaciones reportados del uso del bajo código para obtener un marco de beneficios y limitaciones.

Para la caracterización se buscaron trabajos en las bases de datos Web of Science y Science Direct los términos "bajo código (LOW-CODE)" y "Citizen Developers" en títulos de publicaciones, resúmenes o palabras clave de autor. Los resultados se filtraron por años en un rango de 2018 y 2023 dando como resultado 208 artículos donde la tarea principal fue analizar los títulos y resúmenes con el objetivo de ubicar

la literatura en el contexto de interés. Se excluyeron 124 artículos porque tratan el bajo código pero en otras áreas ajenas a la informática o computación. Por ejemplo, código de biomedicina, codificación y decodificación de video y código de diseño sísmico, entre otros. En otros casos, el término bajo código se usaba efectivamente en informática o ingeniería pero refiriéndose a la codificación tradicional. Por ejemplo, encontramos artículos que trataban sobre la calidad, densidad, mantenibilidad y legibilidad del "código bajo", entre otros.

El resultado de lo anterior fueron 84 artículos seleccionados en temas de investigación utilizando análisis temático. El análisis temático es un proceso de investigación cualitativa utilizado para codificar información cualitativa. El análisis temático se usa a menudo en las primeras fases de una investigación para identificar temas, brechas y definir una agenda de investigación futura [19]. Un tema corresponde a un patrón común encontrado en un grupo de artículos que se utiliza para definir los grupos. Al tratar de identificar usos prácticos del bajo código era un objetivo de la revisión, encontramos deliberadamente el tema "aplicaciones empresariales" al inicio del proceso. Los que nos llevó a definir otros temas deductivamente analizando el contenido de los artículos en una primera pasada de revisión. En otra iteración se ajustaron los temas y se aseguraron de que los artículos estuvieran asignados lógicamente a su tema correspondiente.

Dado que no hay investigaciones previas que analicen el uso del bajo código en el contexto de la IES, se obtuvieron los beneficios y limitaciones identificados en trabajos anteriores, independientemente de la industria o contexto empresarial donde fueron reconocidos. La intención de tener los beneficios y limitaciones construido sobre conocimientos previos, que pueda ser validado considerando las ideas derivadas de estudios de casos y las perspectivas de los profesionales capturadas a través de una encuesta. Para lo cual se analizaron los artículos cronológicamente y obtuvieron los beneficios y limitaciones en una matriz (Tabla 1).

TABLA I
MATRIZ DE BENEFICIOS Y LIMITACIONES DEL USO DE BAJO CODIGO
(LOW-CODE)

TABLA 1: MATRIZ DE BENEFICIOS Y LIMITACIONES DEL USO DE BAJO CODIGO (LOW-CODE)																
Beneficios (B) Y Limitantes (L)	Fuente:															
	[3]	[21]	[20]	[5]	[41]	[16]	[9]	[65]	[64]	[1]	[2]	[35]	[74]	[72]	[75]	[73]
B1 - Mayor velocidad/Agilidad																
B2 - Ahorro de costos																
B3 - Reducción de la complejidad																
B4 - Mantenimiento más fácil																
B5 - Participación de perfiles de negocio																
B6 - Requisitos inestables/inconsistentes																
B7 - Aumento de la privacidad																
L1 - Falta de personalización																
L2 - Escalabilidad																
L3 - Fragmentación																
L4 - Dependencia del proveedor																
L5 - Alto costo de licencias																
L6 - Seguridad																

Esto incluyó 3 publicaciones relevantes adicionales encontradas en literatura no convencional [3,20,21] En este proceso, se excluyeron los artículos que citan investigaciones previas. Varios artículos se centran en describir los aspectos técnicos de una plataforma específica de bajo código, en lugar de considerar implicaciones más amplias y posibles desventajas (por ejemplo, [22,23]). Este tipo de artículos no se incluyeron en el marco. También es importante señalar que la literatura no siempre se refiere a los beneficios y limitaciones utilizando una terminología consistente; Ejemplo, en términos de beneficios, algunos autores mencionan a la reducción de la complejidad como "simplicidad". Para el término de limitaciones, algunos autores se refieren a la "fragmentación", como integración limitada o bloqueo del proveedor o como dependencia de terceros. Se procuró racionalmente estos factores y variaciones en la terminología para proporcionar una visión general completa de los beneficios y limitaciones enfatizados en trabajos anteriores.

III. CASO DE ESTUDIO

El caso de estudio aborda la necesidad de optimizar la gestión de inventarios en la división de sistemas computacionales del Instituto Tecnológico José Mario Molina Pasquel y Henríquez, Unidad Académica Zapopan. La investigación propone la implementación de una solución innovadora basada en tecnología LOW-CODE, específicamente una aplicación desarrollada en AppSheet e integrada con hojas de cálculo de Google.

Esta propuesta surge tras un minucioso análisis de los requerimientos institucionales y una evaluación exhaustiva de su viabilidad técnica y económica. El objetivo principal es revolucionar la administración de inventarios mediante la implementación de un sistema integrado que combine la versatilidad de AppSheet con la robustez de las hojas de cálculo de Google.

Para alcanzar este objetivo, se llevaron a cabo las siguientes acciones clave:

1. Diseño de una base de datos adaptada a las necesidades específicas de gestión de inventarios del almacén, cumpliendo con los estándares establecidos.
2. Desarrollo de una integración eficiente entre AppSheet y las hojas de cálculo de Google, garantizando un sistema multimodal cohesivo.
3. Elaboración de un manual de usuario detallado, facilitando la adopción y el uso correcto de la aplicación por parte del personal encargado.

La implementación de esta solución tecnológica ha resultado en mejoras significativas en la gestión de inventarios del Instituto. Entre los beneficios observados se destacan:

- Mayor eficiencia operativa
- Incremento en la precisión de los datos

- Mejora en la visibilidad y control de inventarios
- Optimización de la colaboración y comunicación entre los actores involucrados

Este caso de estudio demuestra cómo la adopción de tecnologías LOW-CODE, como AppSheet da la suite de Google Workspace, puede transformar procesos tradicionales en soluciones modernas y eficientes, adaptadas a las necesidades específicas de instituciones educativas de nivel superior.

Beneficios y Limitaciones del framework.

El análisis de la literatura sobre plataformas de código bajo revela un conjunto de ventajas y desafíos significativos para su implementación en entornos educativos superiores.

Entre los beneficios más destacados se encuentran:

1. Agilidad en el desarrollo: Las IES pueden crear aplicaciones funcionales con mayor rapidez, respondiendo ágilmente a las necesidades cambiantes del entorno académico.
2. Optimización de recursos: Se reduce la necesidad de contratar desarrolladores especializados, lo que puede resultar en ahorros significativos para las instituciones educativas con presupuestos limitados.
3. Simplificación de procesos: El uso de componentes predefinidos facilita el desarrollo de aplicaciones, permitiendo a las IES centrarse en satisfacer las necesidades específicas de estudiantes y personal.
4. Facilidad de mantenimiento: Las aplicaciones de código bajo suelen requerir menos esfuerzo para su mantenimiento, lo que es crucial para las IES con recursos de TI limitados.
5. Participación de personal no técnico: Permite que el personal académico y administrativo participe en el desarrollo de soluciones, aprovechando su conocimiento directo de las necesidades institucionales.
6. Validación eficiente de requisitos: Facilita la creación rápida de prototipos, permitiendo a las IES validar conceptos antes de invertir en desarrollos más extensos.
7. Mayor control de datos: Al desarrollarse internamente, las IES pueden mantener un mejor control sobre la información sensible de estudiantes y personal.

Sin embargo, también se identifican limitaciones importantes:

8. Restricciones de personalización: Las IES pueden encontrar limitaciones al intentar adaptar las aplicaciones a necesidades muy específicas o complejas.
9. Desafíos de escalabilidad: Puede ser difícil expandir las soluciones de código bajo para cubrir todas las necesidades de una institución educativa grande o en crecimiento.

10. Problemas de interoperabilidad: La integración entre diferentes plataformas de código bajo puede ser complicada, lo que podría afectar la gestión de datos en múltiples sistemas institucionales.
11. Dependencia del proveedor: Las IES podrían quedar vinculadas a un proveedor específico, lo que podría limitar la flexibilidad futura.
12. Costos de licencia a largo plazo: Aunque el desarrollo inicial puede ser económico, los costos pueden aumentar significativamente al expandir el uso de la aplicación en toda la institución.
13. Preocupaciones de seguridad: El desarrollo descentralizado de aplicaciones podría plantear riesgos de seguridad para los datos sensibles manejados por las IES.

IV. RESULTADOS

Un factor clave de la solución es que la institución cuenta con una membresía de Google Workspace para educación, asociada al dominio institucional, y proporciona cuentas de correo electrónico a todos los miembros. Este entorno ofrece una capa adicional de seguridad y robustez a la implementación de AppSheet, ya que aprovecha las políticas de seguridad y autenticación proporcionadas por Google como se muestra en la figura 1. Se diseñó una base de datos en hojas de cálculo de Google con campos estandarizados para el almacén. Esta estructura permite un registro eficiente y categorización precisa de los bienes, adaptándose a las necesidades específicas de los programas de sistemas computacionales.

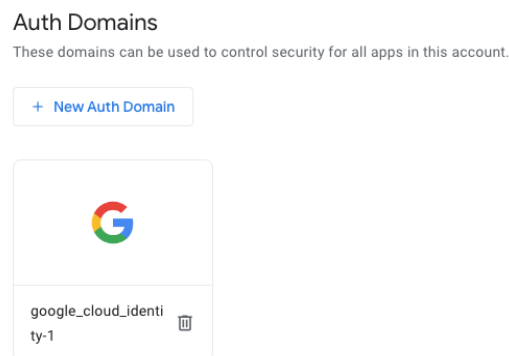


Fig. 1. Autenticación por el dominio institucional.

La aplicación desarrollada ofrece funcionalidades clave como: Registro del inventario de equipos informáticos, componentes y herramientas especializadas, sistema de gestión de salidas, crucial para el préstamo de equipos a estudiantes y profesores para prácticas y proyectos, interfaz para el seguimiento de devoluciones, permitiendo identificar rápidamente responsables de equipos no retornados, capacidad para realizar inventarios periódicos mediante dispositivos móviles, asegurando un control más riguroso del stock, Generación de reportes detallados de inventario,

salidas y faltantes, esenciales para la toma de decisiones administrativas, algunas capturas se aprecian en la figura 2.

La integración con Google Workspace permite además: autenticación segura utilizando las credenciales institucionales, control de acceso granular basado en roles y permisos del dominio institucional, almacenamiento seguro de datos en la nube de Google, con las políticas de seguridad y cumplimiento normativo de Google Workspace para educación, fácil colaboración y compartición de información entre departamentos, respetando los permisos establecidos.

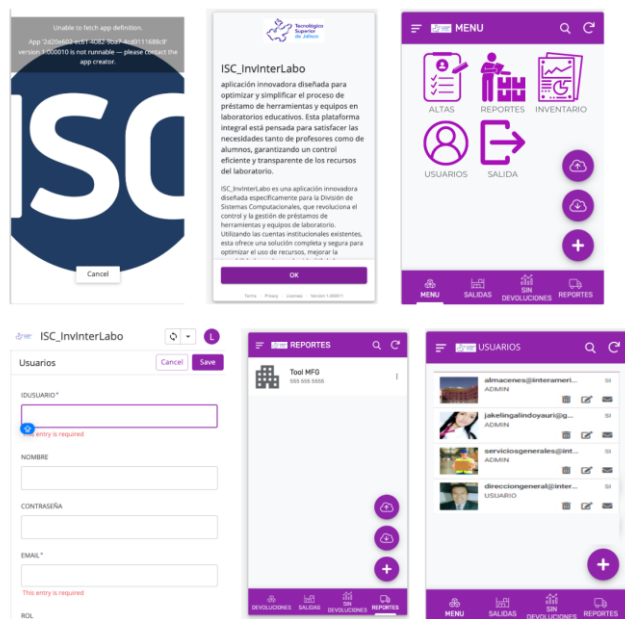


Fig. 2: Capturas de pantalla de la APP ISC_InvInterLabo.

Esta implementación LOW-CODE no solo optimiza la gestión de inventarios, sino que también se alinea con los objetivos de digitalización y modernización del Instituto, preparando el camino para futuras innovaciones en la gestión educativa y administrativa.

La utilización de la infraestructura existente de Google Workspace garantiza una solución costo-efectiva, segura y escalable, perfectamente integrada en el ecosistema digital ya establecido en la institución.

V. CONCLUSIONES

La investigación demuestra cómo la adopción de tecnologías LOW-CODE, como AppSheet, puede transformar procesos tradicionales en soluciones modernas y eficientes, adaptadas a las necesidades específicas de instituciones educativas de nivel superior.

Mientras que las plataformas de código bajo ofrecen oportunidades significativas para que las IES mejoren sus procesos y servicios de manera eficiente, es crucial que estas instituciones evalúen cuidadosamente tanto los beneficios

como las limitaciones en el contexto de sus necesidades específicas, recursos disponibles y objetivos a largo plazo.

AGRADECIMIENTOS

Agradecer al tecnológico por brindarnos el apoyo de seguir trabajando en la comprobación de la metodología IBT en beneficio de los proyectos de base tecnológica utilizando el código bajo.

REFERENCIAS

- [1] A.C. Bock, U. Frank, "Low-code platform," *Business & Information Systems Engineering*, vol. 63, no. 6, pp. 733–740, 2021. DOI: <https://doi.org/10.1007/s12599-021-00726-8>.
- [2] Y. Luo, P. Liang, C. Wang, M. Shahin, J. Zhan, "Characteristics and challenges of low-code development: the practitioners' perspective," *Proceedings of the 15th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM)*, pp. 1–11, 2021. DOI: <https://doi.org/10.1145/3475716.3475782>.
- [3] C. Richardson, J. R. Rymer, C. Mines, A. Cullen, and D. Whittaker (2014) New development platforms emerge for customer-facing applications. Forrester. [En línea]. Disponible en: <https://www.forrester.com/report/New-Development-Platforms-Emerge-For-CustomerFacing-Applications/RES113411>.
- [4] M. Fryling, "Low code app development," *Journal of Computing Sciences in Colleges*, vol. 34, no. 6, pp. 119, 2019. <https://dl.acm.org/doi/abs/10.5555/3344051.3344061>.
- [5] R. Waszkowski, "Low-code platform for automating business processes in manufacturing," *IFAC-PapersOnLine*, vol. 52, no. 10, pp. 376–381, 2019. DOI: <https://doi.org/10.1016/j.ifacol.2019.10.060>.
- [6] J. Everhard (2019) The Pros and Cons of Citizen Development. [Online]. Available: <https://www.forbes.com/sites/johneverhard/2019/01/22/the-pros-and-cons-of-citizen-development/?sh=4d04dfac84fd>.
- [7] C. Olariu, M. Gogan, and F. Rennung, *Switching the center of software development from it to business experts using intelligent business process management suites*, in: *Soft Computing Applications*, pp. 993–1001, 2016. DOI: https://doi.org/10.1007/978-3-319-18416-6_79.
- [8] Project Management Institute (2021) Introducing PMI Citizen Developer. [Online]. Available: <https://www.pmi.org/citizen-developer>.
- [9] M. Woo, "The rise of no/low code software development—no experience needed?" *Engineering*, vol. 6, no. 9, pp. 960–961, 2020. DOI: <https://doi.org/10.1016/j.eng.2020.07.007>.
- [10] A. Sahay, A. Indamutsa, D. Di Ruscio, and A. Pierantonio, "Supporting the understanding and comparison of low-code development platforms," in: *46th Euromicro Conference on Software Engineering and Advanced Applications (SEAA)*, pp. 171–178, 2020. DOI: <https://doi.org/10.1109/SEAA51224.2020.00036>.
- [11] A. Bucaioni, A. Cicchetti, and F. Ciccozzi, "Modelling in low-code development: a multi-vocal systematic review,"

- Software and Systems Modelling*, vol. 21, pp. 1959–1981, 2022. DOI: <https://doi.org/10.1007/s10270-021-00964-0>.
- [12] J. Cabot, “Positioning of the low-code movement within the field of model-driven engineering,” in: *Proceedings of the 23rd ACM/IEEE International Conference on Model Driven Engineering Languages and Systems: Companion Proceedings*, pp. 1–3, 2020. DOI: <https://doi.org/10.1145/3417990.3420210>.
- [13] J. Wong, K. Iijima, A. Leow, A. Jain, P. Vincent (2021) Magic quadrant for enterprise low-code application platforms. Gartner. [Online]. Available: <https://www.gartner.com/en/documents/4005939>.
- [14] Research and Markets (2020), Low-Code Development Platform Market by Component (Platform and Services), Application Type, Deployment Type (Cloud and On-Premises), Organization Size (SMEs and Large Enterprises), Industry, and Region -Global Forecast to 2025. [Online]. Available: <https://www.researchandmarkets.com/reports/5022324/low-code-development-platform-market-by-component>.
- [15] J. Wong, M. Driver, S. Ray (2019) The future of apps must include citizen development. Gartner. [Online]. Available: <https://www.gartner.com/en/documents/3970067>.
- [16] M. Tisi, J. M. Mottu, D. Kolovos, J. D. Lara, E. Guerra, D. D. Ruscio, A. Pierantonio, and M. Wimmer, “Lowcomote: training the next generation of experts in scalable low-code engineering platforms,” *STAF Co-Located Events Joint Proceed*, 2019. DOI: <https://doi.org/10.5281/zenodo.4314656>.
- [17] I. Wolff (2019) Making in-house apps with ash low-code, no-code platforms. Advanced Manufacturing. [Online]. Available: <https://www.sme.org/technologies/articles/2019/october/software-making-in-house-apps-with-low-code-no-code-platforms/>.
- [18] R. Sanchis, O. Garcia-Perales, F. Fraile, and R. Poler, “Low-code as enabler of digital transformation in manufacturing industry,” *Applied Sciences*, vol. 10, no. 1, 2020. DOI: <https://doi.org/10.3390/app10010012>.
- [19] R. E. Boyatzis, *Transforming Qualitative Information: Thematic Analysis and Code Development*, Sage Publications, 1998. ISBN: 0–7619–0961-3.