



Tipo de artículo: Artículos originales
Temática: Calidad de software
Recibido: 30/04/2024 | Aceptado: 15/06/2024 | Publicado: 30/09/2024

Identificadores persistentes:
DOI: [10.48168/innosoft.s16.a176](https://doi.org/10.48168/innosoft.s16.a176)
ARK: [ark:/42411/s16/a176](https://nbn-resolving.org/urn:nbn:pe:ulasalle-1-1-42411-s16-a176)
PURL: [42411/s16/a176](https://purl.org/urn:nbn:pe:ulasalle-1-1-42411-s16-a176)

Propuesta de una Metodología basada en DevSecOps y SCRUM para el desarrollo de Software con Calidad

Proposal of a DevSecOps SCRUM-based Methodology for Software Development with Quality

Itzel Xochitototl Cote¹[\[0009-0008-1409-549X\]](mailto:m23370012@apizaco.tecnm.mx)^{*}, José Juan Hernández Mora²[\[0000-0003-2878-7290\]](mailto:juan.hm@apizaco.tecnm.mx), María Guadalupe Medina Barrera³[\[0000-0003-3074-0029\]](mailto:guadalupe.mb@apizaco.tecnm.mx), Elizabeth Cuatecontzi Cuahutle⁴[\[0009-0008-0531-0354\]](mailto:elizabeth.cc@apizaco.tecnm.mx), Juan Ramos Ramos⁵[\[0009-0004-7440-7257\]](mailto:juan.rr@apizaco.tecnm.mx)

¹Tecnológico Nacional de México/Campus Apizaco. México. C.P. 90300. m23370012@apizaco.tecnm.mx

²Tecnológico Nacional de México/Campus Apizaco. México. C.P. 90300. juan.hm@apizaco.tecnm.mx

³Tecnológico Nacional de México/Campus Apizaco. México. C.P. 90300. guadalupe.mb@apizaco.tecnm.mx

⁴Tecnológico Nacional de México/Campus Apizaco. México. C.P. 90300. elizabeth.cc@apizaco.tecnm.mx

⁵Tecnológico Nacional de México/Campus Apizaco. México. C.P. 90300. juan.rr@apizaco.tecnm.mx

*Autor para correspondencia: m23370012@apizaco.tecnm.mx

Resumen

La colaboración en equipos de desarrollo de software a menudo necesita una buena gestión de las actividades; con la aplicación de metodologías ágiles es posible administrar los cambios, sin embargo, una cultura de colaboración transversal en la organización y una responsabilidad compartida en relación con la seguridad en el desarrollo permite una comunicación integral, lo que les da mejores herramientas a las partes interesadas para la toma de decisiones.

El objetivo de este trabajo es describir la propuesta de una metodología para la gestión del desarrollo de proyectos de software con base en DevSecOps y Scrum; posteriormente, se prueba la metodología en el desarrollo de dos casos de estudio y se presentan los resultados, además, identifican las mejoras en los tiempos de entrega y la calidad del desarrollo de software.

Finalmente, se comenta cómo la integración de metodologías ágiles DevSecOps y SCRUM en la gestión de proyectos de desarrollo de software proporciona una estructura robusta para adaptarse a los cambios y mejorar la calidad del producto final.

Palabras claves: Colaboración transversal, DevSecOps, Gestión del desarrollo, Mejora Continua, Seguridad de la información.

Abstract

Collaboration in software development teams often requires effective activity management. Agile methodologies enable managing changes; however, fostering a culture of cross-functional collaboration within the organization and shared responsibility for security in development facilitates comprehensive communication. This provides stakeholders with better decision-making tools.

The aim of this paper is to outline a methodology proposal for managing software development projects based on DevSecOps and Scrum. Subsequently, the methodology is tested in two case studies, and the results are

presented, highlighting improvements in delivery times and software development quality. Finally, it discusses how integrating agile methodologies like DevSecOps and Scrum in software development project management provides a robust framework to adapt to changes and enhance the quality of the final product.

Keywords: *Cross-functional collaboration, DevSecOps, Development management, Continuous Improvement, Information Security.*

Introducción

Según investigaciones en el ámbito de la tecnología y el desarrollo de software, se evidencia que existen diversos desafíos como una mala planeación, falta de comunicación con el cliente, cambio de alcances, personal mal capacitado y fallas en el proceso de desarrollo del software llevan al desecho del trabajo previo [1–4]. Para afrontar estos riesgos, se han propuesto metodologías ágiles que permiten la entrega de software constante y tomando en cuenta siempre las necesidades del cliente y alineando el trabajo a los objetivos de la empresa. Sin embargo, aunque se encuentra amplia información acerca de estas metodologías, poco se habla de su correcta implementación y de los problemas culturales a los que los líderes de equipos de desarrollo se enfrentan al momento de desarrollar software de calidad [5–9]

Por ello, el presente trabajo investiga y analiza la combinación de la metodología ágil DevSecOps y SCRUM en una empresa internacional, en sus equipos de desarrollo de un software, para la gestión del capital humano y tiene como objetivo administrar los requerimientos del cliente a manera de evitar fallas en las entregas de los proyectos [6], [7], [9], [10]. DevSecOps como cultura organizacional permite estar al pendiente de los Feedback que el cliente envía durante el desarrollo del proyecto, mientras que Scrum gestiona las células de desarrollo de software en la conformidad del cliente en cada Sprint [6], [11].

Además, Scrum como metodología ágil, ha probado su funcionalidad, pero también se ha experimentado la falta de compromiso organizacional en las empresas que carecen de cultura DevSecOps alineada a los estándares de Scrum [12], [13].

En el desarrollo de la metodología propuesta, los casos de estudio, procesos y resultados presentados en este proyecto permiten a los lectores la comprensión de las metodologías y la correcta aplicación, al mismo tiempo, se podría interpretar el presente trabajo como guía de aplicación de la metodología denominada ÁgilSegura , y algunos ejemplos de problemas encontrados en las aplicaciones de prueba y posibles soluciones.

Por otro lado, se analizan trabajos relacionados con DevOps y la gestión del talento humano en el desarrollo de software. Según Navarro [14] DevOps implica un cambio de mentalidad y paradigma laboral, lo que re-

quiere adoptar nuevas herramientas, tareas y formas de comunicación. Este cambio en lo personal y grupal es fundamental para la implementación exitosa de DevOps en una organización.

Por su parte, García-Mireles et al. [15] destacan la importancia de la comunicación con los clientes para identificar áreas de mejora en las aplicaciones y monitorear los desechos generados durante el desarrollo. También señalan la necesidad de alentar al equipo de desarrollo desde una perspectiva gerencial. Asimismo, se menciona la interacción entre SCRUM y DevOps en el desarrollo de proyectos, subrayando su relevancia en la gestión del talento humano y los resultados obtenidos.

En otros estudios, recomiendan aplicar un enfoque estructurado para gestionar el cambio y alentar a los empleados a adoptar nuevas prácticas, lo que puede aumentar la velocidad de adopción y la eficiencia del proyecto, reduciendo costos en el proceso. Sin embargo, la mala gestión del cambio puede poner en riesgo tanto a los individuos como al proyecto y la organización en su conjunto, por lo que es crucial diseñar herramientas de prevención de riesgos y asegurar que las personas realicen su trabajo de manera diferente para alcanzar los objetivos del proyecto a tiempo y dentro del presupuesto [12], [13].

Metodología computacional

La metodología propuesta ÁgilSegura basada en DevSecOps y Scrum, permite una gestión integral, ya que, involucra a todas las personas interesadas en el desarrollo del proyecto, con el fin de mejorar los indicadores de desempeño de los desarrolladores para entregar un producto de calidad y obtener ventajas competitivas como se observa en la Figura 1, donde se muestra la integración para desarrollo de software.

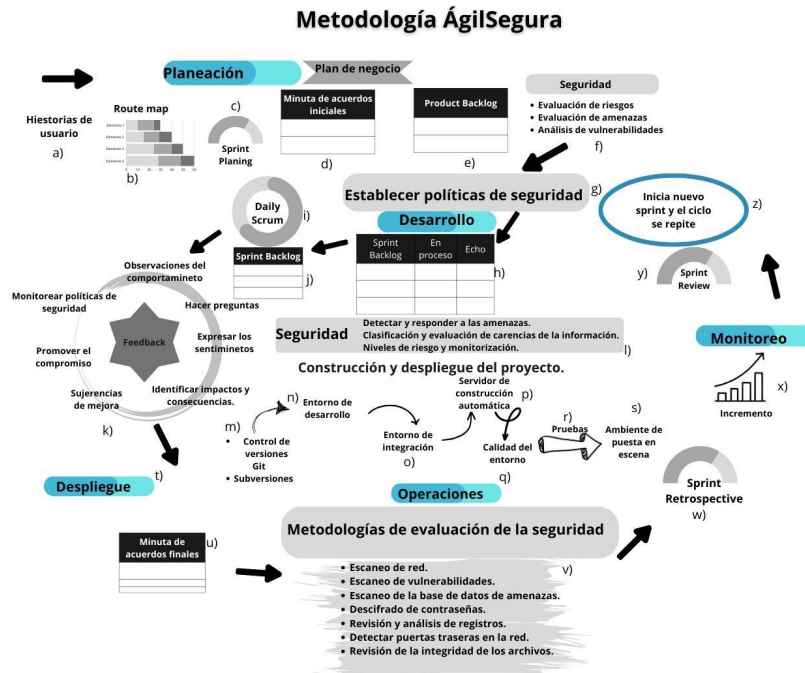


Figura 1. Metodología ÁgilSegura

En la Figura 1, se puede observar cómo gestionar el desarrollo de una aplicación a través de fases como la planeación, el desarrollo, el despliegue, las operaciones y el monitoreo.

Durante la etapa de **Planeación**, se crea el plan de negocio, el cual, establece una relación comercial entre la empresa y el cliente. Se definen las historias de usuario del proyecto, para dar paso a la hoja de ruta (*Road map*), que es un gráfico que muestra las actividades a desarrollar y su tiempo de ejecución.

Además, en *Sprint Planning*, se deben de generar la minuta de acuerdos iniciales y el *Product Backlog*, que son las actividades por desarrollar. También, es importante evaluar los riesgos y amenazas del proyecto, así como, realizar un análisis de las vulnerabilidades de la información para poder elaborar las políticas de seguridad del proyecto.

Después, durante la etapa de **Desarrollo** se lleva a cabo el *Sprint*, que es un evento con tiempo definido, en el que se llevan a cabo diversas actividades, con el fin de codificar ordenadamente el proyecto de software y se debe de tener un tablero Kanban, para ordenar las actividades de *Sprint Backlog*.

En consecuencia, es fundamental llevar a cabo reuniones diarias, conocidas como *Daily Scrum*, donde se dis-

cuten dudas, inconvenientes, retrasos y acuerdos. Durante estas reuniones, se elabora una minuta de acuerdos que permite una comunicación efectiva y una retroalimentación adecuada. En este contexto, durante el *feedback*, se expresan sentimientos, se identifican impactos y consecuencias, se proponen mejoras y se fomenta el compromiso tanto del equipo como del cliente. Además, se monitorean las políticas de seguridad y se observan los comportamientos para garantizar un desarrollo óptimo del proyecto.

Durante el *sprint*, es importante detectar y responder a los riesgos del proyecto. Con el análisis de la seguridad de la información se pueden clasificar y evaluar las carencias que se tienen y de esta manera identificar los niveles de riesgo y monitorearlos.

Luego entonces, el entorno de desarrollo utilizado en la construcción y el despliegue de la aplicación, debe contar con control de versiones; por ejemplo, GITHUB, además debe de estar bien administrado para mitigar errores en la integración del código entre los miembros del equipo y se debe de considerar utilizar un servidor de construcción del sistema; asimismo, permanentemente evaluar la calidad del producto, la integración, realizar documentación de las pruebas y evaluar el ambiente de puesta en escena del proyecto.

Y con el fin de documentar, se integran en forma continua las actualizaciones al software, por lo que se deben de ir corrigiendo errores para mejorar las funcionalidades del sistema con cada iteración.

Durante la etapa de **Despliegue y Operaciones** se genera la minuta de acuerdos finales del sprint, donde se evalúan las funcionalidades del sistema, escaneo de la red, escaneo de vulnerabilidades, revisión de puertas traseras, revisión de la integridad de los archivos y de las actividades que realiza el software desarrollado. Y se lleva a cabo una reunión de retrospectiva (*Sprint Retrospective*), para identificar mejoras y problemas encontrados.

Además, se entrega un producto mínimo viable al cliente, que es un incremento en el desarrollo del proyecto, para ello se debe de cumplir con todas las actividades establecidas en el *Sprint Backlog*.

También es importante **Monitorear** el correcto funcionamiento del sistema entregado en esta etapa y se realiza una reunión de revisión de los incrementos al Proyecto, las que se deben de mantener y las que se deben de modificar. Todo ello, con el objetivo de generar referencias de mejora para implementar el siguiente *sprint*.

El principal objetivo de esta metodología es mejorar el rendimiento del personal del equipo de desarrollo, reduciendo la deuda técnica, logrando la entrega de los proyectos bajo un sello de calidad con enfoque en seguridad de la información. Y con ello, lograr eficiencia económica para la empresa; pero, sin olvidar que el cambio que se sugiere en la propuesta debe ser integral.

Resultados

Para verificar la metodología ÁgilSegura, se implementa en dos casos de estudio.

Caso uno: Se probó la metodología con proyectos a estudiantes de Octavo Semestre de Ingeniería en Tecnologías de la Información y comunicación, donde, se les proporcionó información sobre la metodología, las fases, los procesos y actividades que se utilizaron y la forma en cómo se valida la información de salida.

El objetivo fue que los alumnos aprendieran la metodología y la aplicaran en el desarrollo de un producto en equipo.

Resultados: Durante la ejecución de las pruebas de la metodología, se encontraron patrones de implementación de la misma y se pudieron clasificar en dos grupos, los que aplicaron la metodología y los que no la aplicaron.

Así mismo, se observó que, en un equipo, la falta de interés en la correcta aplicación de la metodología interfirió de forma directa con el logro de los objetivos. Por el lado positivo, los equipos aprendieron a comunicarse y dominar el estrés que provocó la constante entrega de producto de valor para el cliente. Como se puede observar en la Figura 2, donde se muestra la comparación del desempeño de los grupos con y sin la metodología.

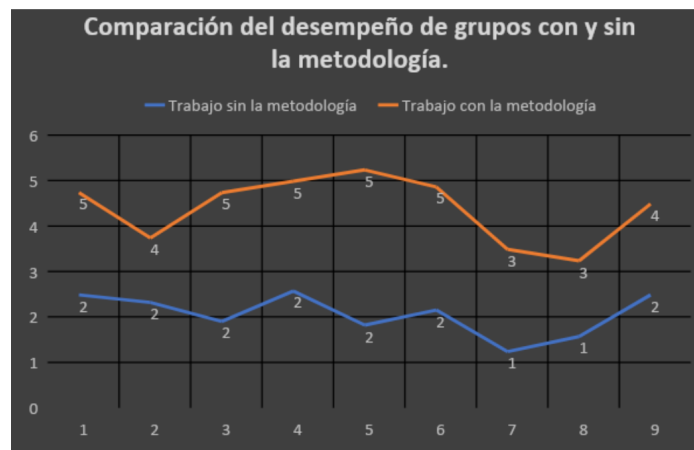


Figura 2. Comparación del desempeño de grupos con y sin metodología ÁgilSegura.

Además, en la Figura 2 se observa que los equipos se evaluaron con una escala de 1 a 5, donde 5 es la calificación más alta y las categorías a evaluar en el desempeño son la forma en que el equipo analizó los requerimientos, cómo realizó la asignación de roles, determinación del backlog, el control del progreso de las actividades, el diseño del producto, la comunicación con el cliente, la entrega de valor y la calidad de retrospectiva que se realizó, con el fin de obtener una muestra completa de la aplicación de la metodología.

Caso dos: Gestión de la aplicación Chat Corporativo en una empresa de desarrollo de software internacional.

Herramienta: la gestión y coordinación de las actividades de este proyecto se llevó a cabo mediante Trello, además, se desarrolló un formato de minutas de acuerdos para poder mantener una comunicación formal entre los miembros del equipo, los gerentes de la empresa y el cliente.

Comentario: Cabe aclarar que la aplicación se encuentra en etapa de pruebas y aún, no cuenta con planes de contingencia, planeación de riesgos, y errores en la estimación de tiempos de entrega. Pero si se pudo observar que la inexperiencia de los desarrolladores, la falta de comunicación entre los miembros de desarrollo del proyecto tienen como consecuencia el retraso en los sprint, por lo que, se podría generar el riesgo de una deuda técnica.

Resultados: En el primer sprint se probó la metodología en el desarrollo del sistema y se observó un completo control, los desarrolladores llevaron a cabo sus roles y sus actividades, las reuniones y las minutas ayudaron a que las actividades se desarrollaran en tiempo y forma como se observa en la Figura 3; pero a medida que los cambios en los requisitos del Sistema y alcance se iban implementando, en los sprint siguientes, la planeación se afectó, los nuevos miembros del equipo no aplicaron la metodología, lo que provocó un descontrol en tiempos de entrega, falta de comunicación como se observa en la Figura 4.

Luego, entonces con la aplicación de la metodología ÁgilSegura en el sprint uno, el equipo no tuvo problemas de entregar los objetivos a tiempo, como se muestra en la Figura 3.

La Figura 3 muestra la evaluación del tiempo de desarrollo planeado, según las actividades del Backlog y lo confronta con el tiempo real del desarrollo; resaltando la entrega del objetivo a tiempo.

Posteriormente, en el sprint dos, los cambios de alcance en el proyecto y nuevos integrantes en el equipo que desconocían la metodología propuesta, provocó que el tiempo de entrega tuvieron un mayor retraso como se puede observar en la gráfica Burndown; que se muestra en la Figura 4. Además, la Figura 4, muestra que, durante el desarrollo del segundo sprint, existe un retraso en el tiempo de entrega de las actividades, desde el día uno hasta el día quince, también, se muestra las actividades que se van realizando a lo largo de los días, lo que indica el retraso en la entrega del proyecto, el cual, es 10 días mayor al planeado.

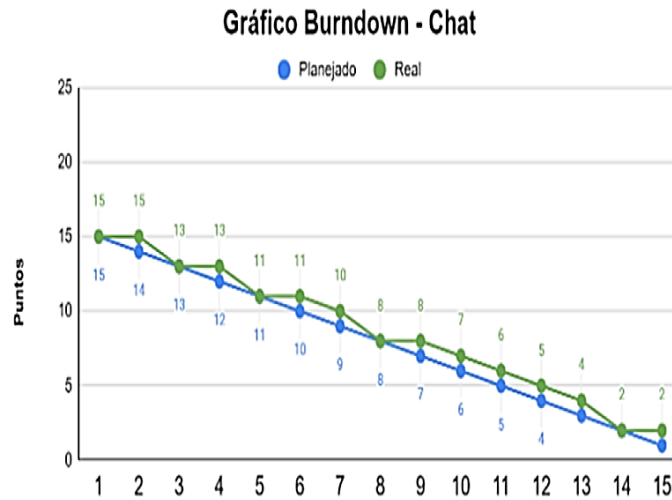


Figura 3. Comparación del desempeño de equipos de desarrollo con la aplicación de la metodología.

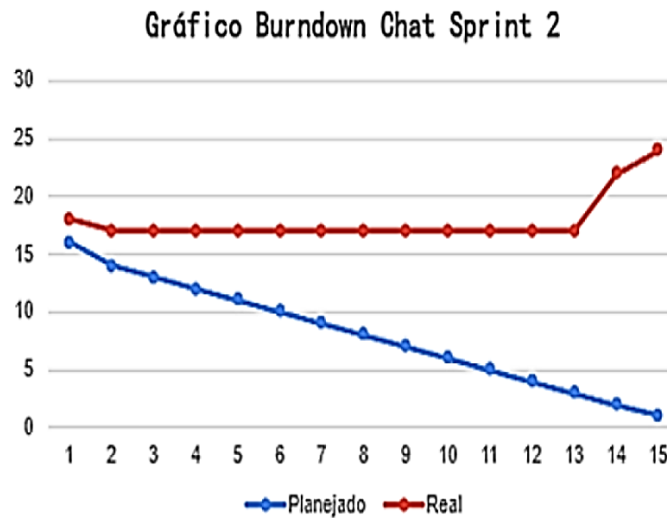


Figura 4. Se muestran los problemas evidentes en el desarrollo del segundo sprint.

Discusión

Los casos de estudio que se presentan revelan varios aspectos interesantes en la aplicación de metodología ÁgilSegura en entornos educativos y empresariales. Ambos casos muestran cómo la implementación y ejecución

de la metodología puede influir significativamente en los resultados finales de los proyectos.

También, se observa que los equipos que no utilizaron la metodología tuvieron problemas con el logro de sus objetivos en tiempo y forma. A pesar de los desafíos, los casos de estudio nos permiten observar cómo los equipos aprendieron valiosas habilidades de comunicación y manejo del estrés, aspectos cruciales en entornos de desarrollo de software donde la entrega continua de valor al cliente es fundamental. Al final, ambos casos de estudio proporcionan lecciones valiosas para estudiantes, profesionales y empresas sobre la implementación práctica de la metodología ÁgilSegura en el desarrollo y gestión de proyectos en entornos reales de desarrollo. Estas lecciones no solo son relevantes desde el punto de vista técnico, sino también en términos de habilidades interpersonales y organizativas necesarias para el éxito en proyectos complejos y dinámicos de creación de software.

Conclusiones

La problemática de la falta de adopción de la cultura DevSecOps se evidencia en la aplicación de esta metodología, cómo se muestra en los casos de estudio 1 y 2; lo cual impacta negativamente en las organizaciones al afectar la comunicación efectiva y en consecuencia el desempeño de los equipos de trabajo. Sin embargo, al analizar la información recabada, se puede observar que los puntos críticos están donde las organizaciones enfrentan problemas de comunicación, seguridad e integración de la información.

Además, identificamos que la aplicación de ÁgilSegura es una solución efectiva para mitigar los problemas de comunicación en el desarrollo de aplicaciones, ya que permite una gestión integral de los proyectos de software. Esta integración proporciona a todos los involucrados en el proyecto la información necesaria para tomar decisiones que beneficien tanto al proyecto como a la organización.

Al final, la adopción de la metodología ÁgilSegura en el desarrollo de un proyecto de software, no solo mejora la eficiencia y la seguridad en las aplicaciones, sino que también fortalece la colaboración y el desempeño de los equipos de trabajo, promoviendo así el éxito organizacional y reduciendo la deuda técnica del proyecto.

Trabajos futuros

Basándonos en la investigación presentada en los casos de estudio sobre la implementación de metodología ÁgilSegura existen varias áreas de trabajo futuro que podrían explorarse:

Estudio de factores que afectan la gestión del tiempo: Dado que el tiempo fue un factor crítico en ambos casos de estudio, sería interesante investigar más a fondo los factores que contribuyen a los retrasos en los

proyectos de desarrollo de software. Esto podría incluir desde análisis de carga de trabajo hasta estudios sobre la efectividad de técnicas de estimación de tiempos.

Implementación de estrategias de gestión de riesgos y planes de contingencia: El segundo caso de estudio resalta la falta de planes de contingencia y gestión ineficaz de riesgos. Investigar y desarrollar mejores prácticas en la identificación temprana de riesgos, así como en la implementación de planes de contingencia efectivos, podría mitigar los impactos negativos en los proyectos.

En resumen, estas áreas no solo pueden ayudar a prevenir problemas conocidos, sino también a mejorar la eficiencia y la calidad en la ejecución de proyectos tecnológicos complejos reduciendo considerablemente la deuda técnica.

Contribución de Autoría

Itzel Xochitototl Cote: Conceptualización, Investigación, Metodología, Software, Validación, Redacción - borrador original. **José Juan Hernández Mora:** Conceptualización, Investigación, Metodología, Análisis formal, Recursos, Visualización, Supervisión, Administración de proyectos, Adquisición de fondos, Curación de datos, Escritura, revisión y edición. **María Guadalupe Medina Barrera:** Conceptualización, Investigación, Metodología, Análisis formal, Recursos, Visualización, Supervisión, Administración de proyectos, Adquisición de fondos, Curación de datos, Escritura, revisión y edición. **Elizabeth Cuatecontzi Cuahutle:** Conceptualización, Investigación, Metodología, Análisis formal, Recursos, Visualización, Supervisión, Administración de proyectos, Adquisición de fondos, Curación de datos, Escritura, revisión y edición. **Juan Ramos Ramos:** Conceptualización, Investigación, Metodología, Análisis formal, Recursos, Visualización, Supervisión, Administración de proyectos, Adquisición de fondos, Curación de datos, Escritura, revisión y edición.

Referencias

- [1] F. Flores-Cerna, V. M. Sanhueza-Salazar, H. M. Valdés-González, and L. Reyes-Bozo, “Metodologías ágiles: un análisis de los desafíos organizacionales para su implementación,” *Revista Científica*, vol. 43, no. 1, pp. 38–49, 2022, doi: 10.14483/23448350.18332.
- [2] C. Bryar and B. Carr, “Have we taken agile too far?” *Harvard Business Review*, 2021. [Online]. Available: <https://hbr.org/2021/04/have-we-taken-agile-too-far>
- [3] K. M. Sánchez, “Proceso de gestión de incidentes y problemas para la mesa de ayuda y de servicios de record S.A, basado en ITIL (information technology infrastructure library),” Montería, Córdoba,

- Colombia: Universidad de Córdoba, 2021. [Online]. Available: <https://repositorio.unicordoba.edu.co/entities/publication/0f600392-dc9a-4a29-b4eb-06e167e07e28>
- [4] R. B. TI, “Los 3 desafíos de implantar una cultura DevOps,” *Revista Byte TI*, 2019. [Online]. Available: <https://revistabyte.es/actualidad-it/devops/implantar-una-cultura-devops/>
- [5] J. S. Q. Monroy, J. V. M. Padua, and I. A. C. Jiménez, “Caracterización y comparación de metodologías ágiles y tradicionales de desarrollo de producto,” *Ciencia e Ingeniería Neogranadina*, vol. 32, no. 2, pp. 9–26, 2022, doi: 10.18359/rcin.5168.
- [6] CLEARED, “Devsecops fundamentals guidebook: Devsecops tools & activities,” DEPARTMENT OF DEFENSE. United States of America, 2021. [Online]. Available: <https://dodcio.defense.gov/Portals/0/Documents/Library/DevSecOpsTools-ActivitiesGuidebook.pdf>
- [7] G. J. A. Zambrano, M. I. D. C. Lazo, D. G. M. Chávez, and D. C. V. Silva, “Gestión de riesgos para el desarrollo de proyectos de sistemas críticos,” *Revista Innovación y Software*, vol. 3, no. 2, pp. 132–139, 2022.
- [8] D. G. C. Carpio, J. F. G. Mendoza, and J. H. Dorado, “Revisión del proceso de mejora de software,” *Revista Innovación y Software*, vol. 3, no. 2, pp. 91–98, 2022, doi: 10.48168/innosoft.s9.a70.
- [9] M. E. C. Arroyo, A. F. V. Cabrera, J. P. S. Fernández, L. E. B. Chavil, and J. L. C. Otero, “Sistema web para mejorar la gestión comercial y de talento humano utilizando la metodología Scrum,” *Revista Innovación y Software*, vol. 5, no. 1, pp. 125–140, 2024, doi: 10.48168/innosoft.s15.a147.
- [10] G. D. L. C. Rodríguez, R. A. M. Fernández, and A. C. M. Fernández, “Seguridad de la información en el comercio electrónico basado en iso 27001 : Una revisión sistemática,” *Revista Innovación y Software*, vol. 4, no. 1, pp. 219–236, 2023, doi: 10.48168/innosoft.s11.a79.
- [11] O. E. C. Uceda and J. M. A. Perez, “Influencia de scrum en los plazos de entrega y rendimiento en los proyectos de las asignaturas de desarrollo de software,” *Revista Iberoamericana de Tecnología en Educación y Educación en Tecnología*, no. 29, pp. 36–42, 2021, doi = 10.24215/18509959.29.e4.
- [12] E. F. Moreno, “La gestión del cambio, la colaboración virtual y la agilidad estratégica organizacional de empresas mexicanas ante los impactos por el COVID-19,” *Nova Sci*, 2021, doi: 10.21640/ns.v13ie.2762.
- [13] P. D. G. Álvarez, “El cambio y su impacto en las organizaciones,” *Academo*, vol. 8, no. 2, pp. 213–220, 2021, doi: 10.30545/academo.2021.jul-dic.10.
- [14] M. Navarro, “Sogeti: “todas las compañías quieren ser devops”,” *Revista Byte TI*, 2021. [Online]. Available: <https://revistabyte.es/entrevistas/sogeti-devops-testing/>

- [15] G. A. García-Mireles, N. P. Olivero, and H. Avila-George, “Devops y la medición de la calidad del producto de software: Hallazgos preliminares,” *RISTI*, no. 53, pp. 37–52, 2024, doi: 10.17013/risti.53.37-52.