

LA IMPORTANCIA DE LOS SISTEMAS DE CONTROL DE VERSIONES EN LA GESTIÓN DE LIBERACIÓN DE SISTEMAS WEB

THE IMPORTANCE OF VERSION CONTROL SYSTEMS IN THE RELEASE MANAGEMENT OF WEB SYSTEMS

Jose Azabache Santos  ORCID, Nelson Angeles Piedra , Alberto Mendoza De los Santos 

Universidad Nacional de Trujillo, Trujillo, Perú

Recibido: 08/11/2022 Revisado: 23/12/2022 Aceptado: 10/01/2023 Publicado: 31/01/2023

RESUMEN

En la actualidad el manejo y disponibilidad de los softwares son muy demandados, han demostrado grandes resultados frente a los sistemas tradicionales para las organizaciones, quienes buscan gestionar su versionamiento para mejorar la calidad de sus servicios.

El presente artículo realizó como objetivo determinar el efecto que causan los sistemas de control de versiones en la gestión de liberación de los servicios web. Para lo cual el presente artículo realizó una revisión sistemática de la información comprendida entre los años 2018 y 2022 para focalizar su importancia.

Obteniendo como resultados que las diferentes empresas ya alojan sus servicios en un repositorio de control de versiones como GitHub, BitBucket, GitLab, entre otros; ayudando a los sistemas a tener un mejor control de futuras versiones y liberación, e incluso de poder tener un respaldo de estas versiones que se van dejando atrás en todo el ciclo de vida de un proyecto.

Palabras Clave Gestión de liberación, Gestión de versiones, Sistemas de Control de Versiones, Sistemas Web.

ABSTRACT

At present, the management and availability of software are highly demanded, they have shown great results compared to traditional systems for organizations, who seek to manage their versioning to improve the quality of their services.

The objective of this article is to determine the effect caused by version control systems in the release management of web services. For which this article carried out a systematic review of the information between the years 2018 and 2022 to focus on its importance.

Obtaining as results that the different companies already host their services in a version control repository such as GitHub, BitBucket, GitLab, among others; helping systems to have better control of future versions and releases, and even to be able to have a backup of these versions that are left behind throughout the life cycle of a project.

Keywords: Release Management, Version Management, Version Control Systems, Web Systems.

INTRODUCCIÓN

El presente trabajo nos permitirá analizar la importancia de los sistemas de control de versiones en la gestión de liberación de sistemas web, que desde la globalización tecnológica los servicios de sistemas web se han convertido en algo imprescindible a nivel individual y empresarial.

Lujan Mora (2002), menciona que los sistemas web son un tipo de software en donde usuarios acceden a un servidor web a través de la red ya sea intranet o internet.

Al tomar a David Otero Gutiérrez (2011), nos dice que a diario se desarrollan infinidad de softwares que en el tiempo van a necesitar de modificación, creación y eliminación de documentos continuamente. Por lo que surge la necesidad de gestionar de alguna manera todos los cambios que se van realizando.

Según Nazatul Nurlisa Zolkifli y otros (2018), definen al sistema de control de versiones como un sistema para gestionar el desarrollo de un objeto en evolución o desarrollo, que registra los cambios realizados por los desarrolladores de software, permitiendo el mapeo, seguimiento y coordinación de objetos de diferentes versiones.

Se comprende por el uso de sistemas de control de versiones, el seguimiento de todas las modificaciones en el código del proyecto que permita conservar las versiones garantizando la disponibilidad y calidad servicio.

ITIL define la gestión de liberación de versiones como el proceso de gestionar, planificar y programar el despliegue de los servicios, actualizaciones y versiones de los servicios de TI al entorno de producción.

En este contexto se entiende por "versión" al desarrollo de una nueva versión de un servicio o componente, e "implementación" se refiere al proceso de integrarla al entorno de producción. Por lo que es necesario demostrar la viabilidad de los proyectos de aplicación web haciendo pruebas piloto de las versiones, mucho antes de desplegarlas, los test necesarios que aseguren su correcto funcionamiento.

En este contexto es importante responder a las siguientes preguntas: ¿Qué impacto tienen los sistemas de control de versiones la gestión de liberación de sistemas web?

MATERIALES Y MÉTODOS

Tipo de estudio

Se realizó una revisión sistemática de la literatura científica, apoyándose en la metodología PRISMA (Preferred Reporting Items for Systematic Reviews and Meta-Analyses). La pregunta de investigación establecida para conducir el proceso metodológico fue la siguiente: ¿De qué manera impacta los sistemas de control de versiones en la gestión de liberación de sistemas webs?

Fundamentación de la metodología

La Revisión Sistemática de la Literatura (RSL) es imprescindible en cualquier trabajo de investigación, nos permite un estudio estructurado de información sustentada teórica y conceptualmente por otros investigadores, determinando así la respuesta a una pregunta de investigación. Además, se encuentran al mejor nivel de jerarquía de la evidencia por encima de investigaciones básicas, estudios observables e incluso ensayos clínicos aleatorizados ya que se encuentra respaldada por múltiples artículos de investigación (Julio Villanueva, 2018).

De acuerdo con Raúl Arévalo (2010), el principal propósito de las Revisiones Sistemáticas es minimizar sesgos a través de la aplicación de métodos sistemáticos y explícitos, tales como una cuestión especifican y fuentes inclusivas mediante una estrategia de búsqueda explícita.

Según Miguel Serrano (2019), Prisma ayuda a diversos autores a sus informes de revisiones sistemáticas, el cual permite una mejor redacción de una investigación. Por lo cual esta metodología nos será útil para planificar y realizar nuestra revisión sistematiza, garantizando así una recopilación y sustentación de información adecuada.

Considerando estas definiciones, se evidencia la importancia de desarrollar el estudio siguiendo una metodología actualizada como Prisma, que considere la evaluación del riesgo de sesgos.

Para tener visión clara, se tendrán en cuenta las pautas y procedimientos mencionados por (Laura Arnau Sabatés y Josefina Sala Roca, 2020) quienes nos detallan los siguientes pasos: diseñar una estrategia de búsqueda, identificar y seleccionar la literatura relevante, registrar los resultados de búsqueda para posteriormente organizarla y finalmente analizar e interpretar los resultados seleccionados.

Estrategia de búsqueda

En primera instancia, la pregunta de investigación aborda la intersección de los temas sistemas de control de versiones y gestión de liberación relacionados en sistemas web. Por ello, hemos iniciado nuestro estudio definiendo las palabras clave. De los cuales, se identificaron los siguientes términos: “release management”, “version management”, “version control”, “web systems”.

Para el desarrollo del estudio, delimitamos las fuentes de información o motores de base de datos tales como REDALYC, SCOPUS, SCIENCE DIRECT, ALICIA Y GOOGLE ACADEMICO.

Criterios de inclusión y de exclusión

De acuerdo con (Laura Arnau Sabatés y Josefina Sala Roca, 2020). Los filtros son importantes en el proceso para discriminar entre las referencias, aquellas que son adecuadas para la investigación.

Por lo tanto, tres capas de criterios de inclusión/exclusión se definen para evaluar la relevancia de cada publicación encontrada (Ver Figura 2).

En primer lugar, los artículos resultantes deben cumplir tres criterios de inclusión:

- (1) Se consideraron artículos escritos en inglés y portugués y español.
- (2) Es importante establecer las fechas desde 2018 al 2022.
- (3) Se establecieron los tipos de documentos académicos (por ejemplo, artículos, libros, tesis, conference paper y artículos de revisión).

En segundo lugar, para aumentar la oportunidad de que los artículos seleccionados se encuentran realmente disponibles, el artículo debe estar disponible en línea. Si el artículo no está disponible públicamente, el siguiente procedimiento es activado.

Tercer lugar, como criterio de exclusión se estableció no considerar aquellas publicaciones que abordan experimentos médicos y temas no relevantes a la investigación.

Catálogos y Bases de Datos

En nuestra búsqueda encontramos un total de 56 artículos originales, distribuidos de la siguiente manera: Redalyc: 2 artículos, SCOPUS, 18 artículos; SCIENCE DIRECT, 7 artículos; ALICIA, 9 artículos; Google Académico 10 artículos. Entre los cuales obtuvimos tesis, revistas, libros y artículos, que describen un enfoque relacionado al tema propuesto (Figura 1).

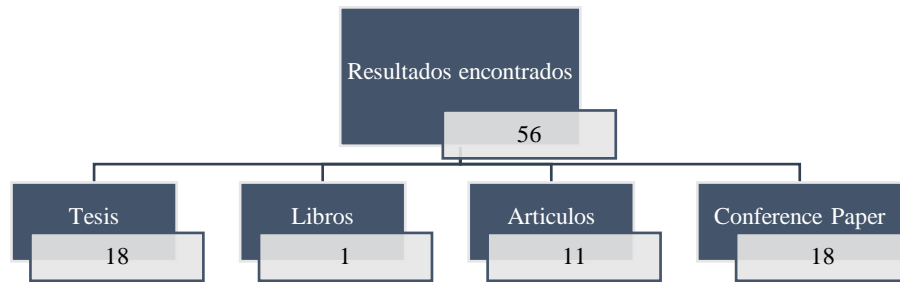


Figura 1. Catálogo y Base de Datos.

REDALYC - 2: ("release management") OR ("version control")

SCOPUS - 28: TITLE-ABS-KEY (("release management") OR ("version control") AND ("web")) AND (LIMIT-TO (PUBYEAR , 2022) OR LIMIT-TO (PUBYEAR , 2021) OR LIMIT-TO (PUBYEAR , 2020) OR LIMIT-TO (PUBYEAR , 2019) OR LIMIT-TO (PUBYEAR , 2018)) AND (LIMIT-TO (EXACTKEYWORD , "Version Control") OR LIMIT-TO (EXACTKEYWORD , "Information Management") OR LIMIT-TO (EXACTKEYWORD , "Version Control System")) SCIENCE DIRECT - 7:

("release management") OR ("version control")

ALICIA - 9: ("release management") OR ("version control") AND ("web")

GOOGLE ACADEMIC - 10: ("release management") OR ("version control") AND ("web")

Por medio de un gráfico (Figura 2), se muestra la cantidad de artículos encontrados, referente a los criterios de exclusión que se establecieron anteriormente, cuyas diferencias fueron analizadas en base a los criterios establecidos, hasta que se obtuvo un total de 08 artículos para la presentación de resultados finales.

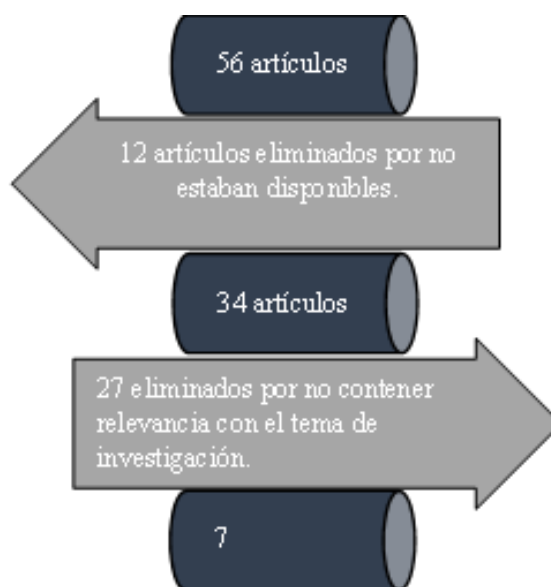


Figura 2. Flujograma de descarte de artículos

RESULTADOS Y DISCUSIÓN

Se procedió a identificar la importancia de sistemas de los sistemas de control de versiones para la gestión de liberación de sistemas web, como lo muestran las publicaciones en las revistas internacionales (Ver Tabla 1).

Tabla 1

Relación de artículos incluidos en la revisión sistemática según los criterios de inclusión.

Autores	Lugar de Procedencia	Año	Título de Investigación	Resultados	Importancia rescatada
Solomon Berhe, Marc Maynard, Foutse Khomhc	Germany	2020	¿Software Release Patterns When is it a good time to update a software component?	Además de los aspectos específicos del producto de software, se debe considerar las recomendaciones del software general, los patrones de lanzamiento de actualizaciones,	La importancia de un sistema de control de versiones controla los errores potenciales que se descubren entre las pruebas; se corrigen antes de

				que pueden reducir aún más el riesgo de romper la actualización del producto de software.	que uno actualice el siguiente componente, lo que reduce el impacto de las correcciones de errores.
Yuxing Ma, StudentMe member, IEEE, Audris Mockus, Fellow, IEEE, Russel Zaretski, Randy Bradley and Bogdan Bichescu	USA	2022	A Methodology for Analyzing Uptake of Software Technologies among Developers	Se proporciono información tecnologías alternativas. De acuerdo con sus hallazgos tienen un valor práctico para los desarrolladores que buscan aumentar la tasa de adopción de sus servicios.	Determinaron tecnologías de control de versiones en muchas forjas, como GitHub, BitBucket, GitLab, Bioconductor, SourceForge, etc. Aprovechando una gran colección de repositorios de control de versiones de código abierto (que contienen más de 4 mil versiones únicas) para construir una cadena de servicio. millones
Pedro Almir M. Oliveira; Pedro A. Santos Neto; Gleison Silva; Irvayne Ibiapina; Werney L.	Brazil	2021	Software Development during COVID-19 Pandemic: An Analysis of Stack Overflow and GitHub	La mayoría de los proyectos durante la pandemia de COVID19 se almacenan en un sistema de control de versión control y hay discusiones sobre ellos en los sitios web de preguntas y	Se reconoce la importancia de los sistemas de control de versiones ante incidentes que afecten la publicación de futuros servicios.

Lira; Rossana M. C. Andrade				respuestas sobre futuras publicaciones.	
Lethbridge, Timothy C.	Canada	2021	Low-Code Is Often High-Code, So We Must Design Low-Code Platforms to Enable Proper Software Engineering	Los sistemas de versión de control, separación de preocupaciones, pruebas automatizadas y programación alfabetizada. Garantizan que permitan el escalado, la comprensión, la capacidad de prueba y la experiencia general del usuario para los desarrolladores.	Se define que los sistemas de control de versiones permiten el escalado, la comprensión, la documentación, la capacidad de prueba de versiones de servicios a publicar.
Eickhoff Thomas, Andreas, Gödel Jens Christian, Eigner Martin	Germany	2020	A Metadata Repository for Semantic Product Lifecycle Management	A lo largo de todo el ciclo de vida de los productos se utiliza Repositorios de metadatos para sus procesos de trabajo diarios, como la Gestión de Lanzamiento dando acceso flexible a la información de los sistemas como el cambio.	Se encontró que, durante el ciclo de producción de un servicio, un pilar importante de la gestión de liberación es tener acceso a la versión de cambios mediante repositorios, en los que encajan los sistemas de control de versiones.
Daniel Alberto MUNIVE ORBEGOS O	Perú	2021	Desarrollo e implementación de mejoras de servicios de comercio exterior para un sistema	La implementación de mejora de servicios hacia nuevas versiones se veía dificultado por la arquitectura heredada	Para una mejora continua de servicios es recomendable migrar a repositorios que ayuden a la

			integrado en un Ministerio del Estado peruano	del sistema integrado. El área de T.I. no tenía implementado un repositorio de librerías o dependencias. Por lo que se recomienda la migración de los componentes a repositorios de dependencias que ayudarán a la construcción y a la gestión de versiones de artefactos de una forma sencilla y estandarizada.	gestión de versiones de artefactos de una forma sencilla y estandarizada.
Marin Diaz, Aymara; Trujillo Casañola, Yaimí; Buedo Hidalgo, Denys	Cuba	2 018	Marco de Trabajo para gestionar actividades de calidad	Se obtuvo como resultado que generalmente os modelos, tanto, así como los estándares que son internacionales agrupan las buenas prácticas propuestas sobre actividades a ejecutar para la detección de defectos.	El autor propone una aplicación de marco de trabajo a la evaluación objetiva de los procesos y los productos, disminuyendo el esfuerzo dedicado a la corrección de defectos a través de la detección de estos en etapas más cercanas a donde se introducen y los defectos en los proyectos de desarrollo de software

La ubicación geográfica del lugar de procedencia, a los que pertenecen las universidades o instituciones identificadas en el estudio, dando a conocer que el tema de sistemas de control de versiones no es de interés global. No obstante, Alemania es el país con mayor seguimiento dentro de nuestra investigación, los demás países cuentan solo un artículo (Ver Tabla 2).

Tabla 2

Relación de artículos incluidos en la revisión sistemática según los criterios de inclusión.

	Cantidad de Artículos	Porcentaj e
Germany	2	28.571%
Brazil	1	14.286%
USA	1	14.286%
Canada	1	14.286%
Peru	1	14.286%
Cuba	1	14.286%
Total	7	100%

Ahora se mostrará una tabla donde se observa la cantidad de tesis, artículos, y libros elegidos finalmente.

Tabla 3

Cantidad de artículos encontrados

Tipo	Cantidad	Porcentaje
Articulo	4	57.143%
Tesis	1	14.286%
Libro	1	14.286%
Conference paper	1	14.286%
Total	7	100%
Articulo	4	57.143%

Sistema de Control de Versiones

Según Trujillo Casañola, Y. y otros (2018), propone ciertas herramientas que permita la planificación y seguimiento de las actividades planificadas y que además permita el registro y seguimiento a las no conformidades y defectos detectados (Ver Figura 3).

Se obtuvo como resultado múltiples tecnologías de control de versiones en muchas forjas, como GitHub, BitBucket, GitLab, Bioconductor, SourceForge, siendo estas las tecnologías más empleadas en la actualidad (Yuxing Ma, 2022).

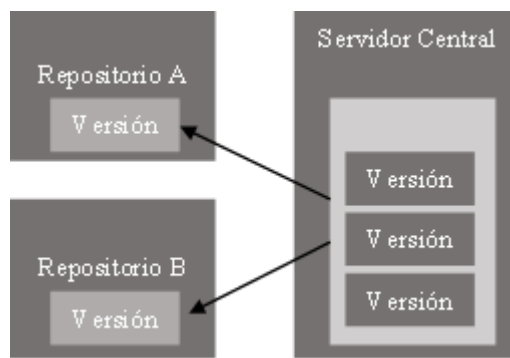


Figura 1. Representación de un Sistema de Control de Versiones

Esto establece que cada desarrollador realiza una copia del repositorio de proyectos completo a su computadora, generándose un repositorio local del proyecto, garantizando así la disponibilidad de los proyectos, ya que es válido señalar que se puede solicitar como entregable cualquier versión en la línea del tiempo del proyecto. Siendo los principales SCV distribuidos de acuerdo a la clasificación presentada en son GIT, Mercurial, BZR y BeetKeeper (Tello Leal, 2021).

Tal es el caso que se rescató la importancia durante la pandemia COVID-2019, en la que se emplearon en gran número control y almacenamiento de versiones con la finalidad de salvaguardar las futuras publicaciones de los servicios (Pedro Almir M, 2021).

Gestión de liberación de sistemas web

En la literatura revisada, se encontró que es necesario gestionar el plan de lanzamiento, la gestión de las tareas, los requisitos ágiles, la gestión del ciclo de vida y la gestión del

cambio (Daniel Munive, 2021). Dentro de ellos el control de versiones en las organizaciones ha resuelto indispensables problemas, dentro de un marco Ágil, evitando fallas en etapas de producción. Por lo que el tamaño, alcance y contenido de cada liberación debería planificarse y gestionarse.

Adicionalmente muchos autores señalaron que el modelo DevOps encaja perfectamente en la gestión de liberación, manteniendo los futuros cambios que puedan desarrollar los servicios, por lo que destacan nuevos términos como DevOps, Gestión de Cambios, Repositorios, entre otros.

La presente revisión sistemática aborda la importancia de los sistemas de control de versiones dentro de las organizaciones, teniendo múltiples sistemas y alojamientos los cuales encajan con las políticas de gestionar un plan de lanzamiento o una gestión de liberación, cumpliendo así las buenas prácticas de cualquier metodología.

No obstante, en nuestro país se encuentran limitantes como la falta de estudios en este tema, a pesar que las grandes instituciones nacionales utilizan gran parte de las técnicas para la gestión de versionamiento.

Según Flores, D. M. (2018), para una mejora continua de servicios es recomendable migrar a repositorios que ayuden a la gestión de versiones de artefactos de una forma sencilla y estandarizada, pese a esto no señala que o cuando migrar a estos repositorios.

Trujillo Casañola, Y. y otros (2018), afirma que, pese a que los estándares internacionales agrupan las buenas prácticas propuestas sobre actividades a ejecutar para la elaboración de proyecto de software, están no especifican técnicas y métodos, roles, responsabilidades, herramientas y guías para llevarlas a cabo. Por lo que escoger cualquier forja de versionamiento es subjetivo a estrategias de la organización o equipo de desarrollo.

Finalmente, otra limitación fueron los años de estudio ya que de la base de datos de SCOPUS Y GOOGLE ACADEMICO, no se obtuvieron óptimos resultados para profundizar detalles de cómo gestionar un plan de versionamiento en las organizaciones. Adicionalmente se identificaron nuevos términos de interés, los cuales servirán para futuras investigaciones en el contexto de versionamiento.

CONCLUSIONES

En el ámbito organizacional las empresas buscan gestionar sus servicios como sistemas web, los cuales presentan nuevas versiones cada periodo de tiempo. La presente investigación indicó la importancia de los sistemas de control de versiones ya que estos permiten el control y las pruebas de lanzamiento, solucionando así posibles errores en etapas de producción e incrementando la calidad de sus servicios. En la praxis las versiones de los sistemas web son alojados en distintos repositorios como GitHub, BitBucket, GitLab, Bioconductor, SourceForge. Permitiendo así que cuenten con una disponibilidad de sus servicios cuando así lo requieran, haciendo que el uso de los sistemas de control de versiones, agregue valor a sus servicios brindados.

Dentro del contexto actual de la pandemia, muchas organizaciones mantuvieron sus servicios alojados en dichos sistemas como una manera de gestionar ágilmente los proyectos, y con la finalidad de salvaguardar sus futuras versiones o conocer todo su transcurso en el tiempo.

Por lo tanto, esta investigación identificó que los sistemas de versionamiento cumplen un papel importante en la gestión de liberación, más no son los únicos dentro de este marco, existen nuevas tecnologías y metodologías que impulsan la planificación, para poder brindar mejores servicios.

REFERENCIAS BIBLIOGRÁFICAS

- Arévalo Barea, R. A., Ortuño, G., & Arévalo Salazar, D. E. (2010). REVISIONES SISTEMÁTICAS (1). Revista médica - Colegio Médico de La Paz, 16(2), 69–80.
http://www.scielo.org.bo/scielo.php?script=sci_arttext&pid=S1726-89582010000200012
- Arnau Sabatés¹, L., & Sala, J. (s/f). La revisión de la literatura científica: Pautas, procedimientos y criterios de calidad. Uab.cat. Recuperado el 19 de junio de 2022, de https://ddd.uab.cat/pub/recdoc/2020/222109/revliltcie_a2020.pdf
- Axelos. (2019). ITIL Foundation: ITIL 4 Edition. Stationery Office Books.

- Bases neurales de la aversión a las pérdidas en contextos económicos: revisión sistemática según las directrices PRISMA: Neurología.com. (s/f). Neurología.com. Recuperado el 19 de junio de 2022, de <https://www.neurologia.com/articulo/2018276>
- Luján-Mora, S. (2002). Programación de aplicaciones web: historia, principios básicos y clientes web. Editorial Club Universitario.
- Moreno, B., Muñoz, M., Cuellar, J., Domancic, S., & Villanueva, J. (2018). Revisiones Sistemáticas: definición y nociones básicas. *Revista Clínica de Periodoncia, Implantología y Rehabilitación Oral*, 11(3), 184–186. <https://doi.org/10.4067/s0719-01072018000300184>
- Otero Gutiérrez, D. (2011). Desarrollo de una aplicación web para control de versiones de software.
- Zolkifli, N. N., Ngah, A., & Deraman, A. (2018). Version control system: A review. *Procedia Computer Science*, 135, 408–415. <https://doi.org/10.1016/j.procs.2018.08.191>
- Eickhoff Thomas, Eiden Andreas, Gödel Jens Christian, Eigner Martin. (2020). A Metadata Repository for Semantic Product Lifecycle Management (B. P. Mpofo K., Ed.). Elsevier B.V. <https://doi.org/10.1016/j.procir.2019.11.006>
- Flores, D. M. (2018). Desarrollo De Un Sistema De Información Basado En Los Requisitos De La Norma Iso/Iec 20000 Para La Mejora De La Calidad De Los Servicios Prestados Por La División De Soporte Informático De La Red Asistencial Juliaca De Essalud 2017 [Universidad Nacional del Altiplano]. <http://repositorio.unap.edu.pe/handle/UNAP/8372>
- Gómez, A. G. (2020). Implantación de Puppet como herramienta de orquestación de sistemas [Universidad Oberta de Catalunya]. <http://openaccess.uoc.edu/webapps/o2/bitstream/10609/107406/6/aguerrero33TFM1219memoria.pdf>
- Hoyos, C. C. (2021). Propuesta Modelo de Gestión de proyectos para la fábrica de software del Ministerio de Educación Nacional [Universidad EAN]. <http://hdl.handle.net/10882/10746>
- Khomh, S. B. M. M. (2020). Software Release Patterns, When is it a good time to update a software component? <https://doi.org/10.1016/j.procs.2020.03.142>.
- Lethbridge, T. C. (2021). Low-Code Is Often High-Code, So We Must Design Low-Code Platforms to Enable Proper Software Engineering. Springer Science and Business Media Deutschland GmbH. https://doi.org/10.1007/978-3-030-89159-6_14
- Oliveira, P. A. M., Santos Neto, P. A., Silva, G., Ibiapina, I., Lira, W. L., & Andrade, R. M. C. (2021). Software Development during COVID-19 Pandemic: An Analysis of Stack Overflow and GitHub. 10.1109/SEH52539.2021.00009.

- Orbegoso, D. A. M. (2021). Desarrollo e implementación de mejoras de servicios de comercio exterior para un sistema integrado en un Ministerio del Estado peruano [Universidad Nacional Mayor de San Marcos]. <https://hdl.handle.net/20.500.12672/18028>
- Yuxing Ma, StudentMember, IEEE, Audris Mockus, Fellow, IEEE, Russel Zaretski, Randy Bradley and Bogdan Bichescu. (2022). A Methodology for Analyzing Uptake of Software Technologies among Developers. 10.1109/TSE.2020.2993758.
- Tello Leal, E., Sosa R., C. M., & Tello Leal, D. A. (2012). Revisión de los sistemas de control de versiones utilizados en el desarrollo de software. *Ingenierías USBmed*, 3(1), 74–81. <https://doi.org/10.21500/20275846.267>
- Trujillo Casañola, Y., Marin Diaz, A., & Buedo Hidalgo, D. (2018). Marco de Trabajo para gestionar actividades de calidad. *Revista cubana de ciencias informáticas*, 12(2), 74–88. <https://www.redalyc.org/articulo.oa?id=378365831006>