

Microservicios para la modernización: Ventajas, Seguridad y Resiliencia

Maryury González Bonilla

Ficha: 2694679

Instructor: Jesús Ariel González Bonilla

04 de diciembre de 2024

Servicio Nacional de Aprendizaje

Resumen

Los micro servicios han emergido como una de las arquitecturas más relevantes para la modernización de sistemas en la actualidad, especialmente en un entorno tecnológico donde la escalabilidad, la flexibilidad y la resiliencia son factores críticos. Esta arquitectura ofrece la capacidad de dividir sistemas complejos en servicios independientes que pueden ser diseñados, implementados, desplegados y mantenidos de manera autónoma, permitiendo que las organizaciones aborden con mayor eficiencia los desafíos que enfrentan las arquitecturas monolíticas tradicionales. Este artículo explora en detalle cómo los microservicios han transformado la forma en que se construyen, operan y optimizan los sistemas modernos, enfocándose en tres aspectos fundamentales: sus ventajas inherentes, las estrategias de seguridad más comunes, y los mecanismos de resiliencia que aseguran la continuidad operativa.

En la primera parte, se analizan las principales ventajas de los microservicios, como la escalabilidad independiente, la modularidad que facilita el mantenimiento y la capacidad de trabajar con tecnologías heterogéneas. En la segunda parte, se aborda la importancia de garantizar la seguridad en entornos distribuidos mediante herramientas como OAuth2, TLS y patrones como API Gateway, que centralizan la autenticación y autorizaciones. Finalmente, se examinan los patrones y técnicas que fortalecen la resiliencia de los sistemas, incluyendo Circuit Breaker, retries y estrategias de monitoreo que permiten mantener la estabilidad ante fallos.

El estudio incluye una revisión exhaustiva de 30 artículos académicos y casos de estudio que ilustran cómo diferentes organizaciones han implementado microservicios para superar limitaciones tecnológicas, mejorando sus capacidades operativas y estratégicas. Si bien los microservicios presentan desafíos técnicos y organizativos, como la complejidad en la gestión y la necesidad de herramientas avanzadas, sus beneficios superan con creces estas dificultades cuando se implementan correctamente. Este artículo concluye que los microservicios no solo representan una solución viable para la modernización tecnológica, sino que también ofrecen un marco para construir sistemas más seguros, flexibles y resilientes.

Abstract

Microservices have emerged as one of the most significant architectures for modernizing systems in today's technological landscape, where scalability, flexibility, and resilience are critical factors. This architecture enables complex systems to be divided into independent services that can be designed, implemented, deployed, and maintained autonomously, allowing organizations to address the challenges posed by traditional monolithic architectures more efficiently. This article provides an in-depth analysis of how microservices have transformed the construction, operation, and optimization of modern systems, focusing on three key aspects: their inherent advantages, common security strategies, and resilience mechanisms that ensure operational continuity. The first section analyzes the primary advantages of microservices, such as independent scalability, modularity that facilitates maintenance, and the ability to work with heterogeneous technologies. The second section highlights the importance of ensuring security in distributed environments using tools like OAuth2, TLS, and patterns like API Gateway, which centralize authentication and authorization. Finally, it examines patterns and techniques that enhance system resilience, including Circuit Breaker, retries, and monitoring strategies that maintain stability in the face of failures. This study includes a comprehensive review of 30 academic articles and case studies illustrating how various organizations have implemented microservices to overcome technological limitations and improve operational and strategic capabilities. While microservices present technical and organizational challenges, such as complexity in management and the need for advanced tools, their benefits far outweigh these difficulties when implemented correctly. This article concludes that microservices not only represent a viable solution for technological modernization but also provide a framework for building more secure, flexible, and resilient systems.

Introducción

En el panorama tecnológico actual, caracterizado por demandas constantes de escalabilidad, flexibilidad y velocidad, las organizaciones enfrentan el desafío de modernizar sus sistemas de información para responder a las necesidades de los usuarios y del mercado. Las arquitecturas monolíticas, aunque sólidas durante décadas, presentan limitaciones como la dificultad de escalar módulos específicos, el impacto de fallos centralizados y la rigidez para integrar nuevas tecnologías. Ante esta situación, los microservicios han emergido como una solución disruptiva, transformando la manera en que se diseñan, despliegan y mantienen las aplicaciones modernas. Esta arquitectura divide los sistemas en pequeños servicios autónomos que interactúan mediante APIs o sistemas de mensajería, lo que permite escalar y actualizar componentes individuales sin afectar al sistema completo. Los microservicios no solo representan una alternativa técnica, sino también un cambio conceptual en el desarrollo de software. Empresas como Netflix, Amazon y Spotify han adoptado esta arquitectura para mejorar su resiliencia, tiempo de respuesta y capacidad de innovación, consolidándose como casos de éxito en la industria tecnológica. Sin embargo, la implementación de microservicios plantea retos significativos, como la complejidad de la comunicación entre servicios, la gestión de transacciones distribuidas y la necesidad de herramientas avanzadas para monitoreo y seguridad. Además, garantizar la resiliencia de estos sistemas ante fallos requiere la adopción de patrones avanzados como Circuit Breaker y estrategias de

recuperación automática. Este artículo tiene como objetivo analizar cómo los microservicios contribuyen a la modernización de sistemas, enfocándose en tres áreas clave: sus ventajas inherentes frente a las arquitecturas monolíticas, las estrategias de seguridad necesarias para proteger entornos distribuidos y los mecanismos de resiliencia que aseguran la continuidad operativa. Para ello, se realiza una revisión exhaustiva de 30 artículos académicos que abarcan desde conceptos fundamentales hasta casos prácticos de implementación, con el propósito de proporcionar una visión integral que facilite la adopción de esta arquitectura en diferentes contextos organizativos. Finalmente, se destacan las herramientas y patrones más relevantes, así como los desafíos técnicos y organizativos asociados, concluyendo que los microservicios representan una solución viable y transformadora para la modernización tecnológica.

...

Metodología

La metodología empleada en este artículo se fundamenta en un enfoque cualitativo orientado a comprender cómo los microservicios contribuyen a la modernización de sistemas empresariales. Para ello, se realizó una revisión exhaustiva de fuentes académicas, estudios de caso y documentación técnica que abordan los aspectos fundamentales de esta arquitectura. El análisis se estructuró en tres áreas clave: las ventajas que ofrecen los microservicios, las estrategias de seguridad aplicadas en entornos distribuidos y los mecanismos que fortalecen su resiliencia. El proceso de investigación se desarrolló en varias etapas. En primer lugar, se identificaron las fuentes relevantes utilizando plataformas reconocidas como Google Scholar, conferencias especializadas y documentos técnicos de la industria. A continuación, las fuentes seleccionadas fueron clasificadas según su contribución a las áreas de interés del artículo, lo que permitió organizar la información en categorías específicas para un análisis más detallado. Durante este proceso, se priorizó la extracción de conceptos fundamentales, patrones arquitectónicos, herramientas tecnológicas y estudios prácticos que ilustraran la aplicación de los microservicios en diferentes sectores. El análisis se complementó con la comparación de enfoques y resultados obtenidos en diferentes casos de estudio, lo que permitió identificar patrones comunes y estrategias exitosas en la implementación de microservicios. Adicionalmente, se consideraron los desafíos técnicos y organizativos asociados, como la gestión de la comunicación entre servicios, la implementación de protocolos de seguridad y las soluciones adoptadas para garantizar la resiliencia ante fallos. Esta metodología busca proporcionar una visión integral y estructurada sobre cómo los microservicios pueden ser implementados eficazmente, abordando no solo sus beneficios, sino también los retos que implican. A partir de este análisis, se presentan recomendaciones prácticas y herramientas clave que facilitan su adopción en diferentes contextos empresariales, con el objetivo de apoyar la modernización tecnológica de las organizaciones.

...

Marco Teórico

La arquitectura de microservicios ha emergido como una solución clave para afrontar los retos de escalabilidad, flexibilidad y seguridad en el desarrollo de software mo-

dermo. En contraste con las arquitecturas monolíticas tradicionales, los microservicios permiten descomponer aplicaciones en componentes pequeños, autónomos e independientes que pueden ser desarrollados, desplegados y escalados individualmente. Este marco teórico explora las ventajas, los retos y las aplicaciones de esta arquitectura, apoyándose en un análisis de 20 estudios relevantes.

Los microservicios son conocidos por su capacidad para escalar de manera eficiente. Estudios como el de Mejía y Noel (2022) destacan que esta arquitectura mejora significativamente la eficiencia operativa al reducir los tiempos de respuesta y facilitar la adaptación a cambios en la demanda. Por ejemplo, empresas como Amazon y eBay han migrado exitosamente hacia microservicios, obteniendo mejoras en el mantenimiento y el desarrollo de sus sistemas (Colombero et al., 2024). La independencia de los componentes es otra ventaja crucial. Según Suárez et al. (2017), esta modularidad permite que los equipos de desarrollo trabajen en diferentes partes del sistema sin afectar el funcionamiento global. Además, facilita la implementación de actualizaciones y nuevas funciones sin interrumpir el servicio. El uso de herramientas modernas como Docker y Kubernetes mejora la agilidad en el despliegue de aplicaciones basadas en microservicios (Moreno Belinchón, 2018). La posibilidad de implementar y gestionar estas aplicaciones en entornos de nube también aumenta la portabilidad y resiliencia del software (Jiménez Aliaga, 2019). Aunque los microservicios ofrecen muchas ventajas, también presentan retos significativos en seguridad. La comunicación entre servicios y el uso de datos sensibles requieren sistemas robustos de protección. Modelos como TLS y OAuth2 son esenciales para garantizar la autenticación y la confidencialidad de los datos (Rienda Iáñez, 2019). Guerrero Martínez (2023) destaca la implementación de estrategias de aprendizaje automático para la detección y prevención de ciberataques en arquitecturas de microservicios. Estas soluciones son especialmente relevantes en contextos como las ciudades inteligentes, donde la seguridad es crucial para la gestión de grandes volúmenes de datos. Los patrones de resiliencia, como el uso de Circuit Breakers y Retry, son fundamentales para mantener la estabilidad del sistema ante fallos. En un estudio aplicado a PedidosYa, Suarez (2023) demostró que estos patrones ayudan a gestionar errores de red y picos de tráfico, aumentando la robustez del sistema.

La integración de herramientas de automatización y monitoreo, como DevOps, es crucial para la implementación exitosa de microservicios. Estas herramientas garantizan la entrega continua y la adaptación a los cambios del mercado (Rodríguez et al., 2020). En el ámbito bancario, los microservicios han demostrado ser efectivos para mejorar la calidad del software, especialmente en sistemas que requieren alta disponibilidad y escalabilidad (Mejía y Noel, 2022). En telecomunicaciones, la implementación de microservicios ha permitido optimizar procesos de cotización y gestión de recursos, utilizando tecnologías como Kafka y Azure para garantizar la escalabilidad y disponibilidad del sistema (Sánchez Matos y Ore Quintana, 2024). El uso de microservicios para implementar modelos de aprendizaje automático en la seguridad de ciudades inteligentes ha demostrado ser una solución escalable y eficiente para gestionar datos en tiempo real (Guerrero Martínez, 2023).

Seguridad en Microservicios: Un Pilar Fundamental

Desafíos de Seguridad en Microservicios

En una arquitectura monolítica, la seguridad generalmente se concentra en proteger el perímetro del sistema. Sin embargo, en microservicios, cada servicio constituye un punto potencial de vulnerabilidad. Entre los principales desafíos destacan:

Comunicación entre Servicios

La interacción constante entre microservicios crea un flujo de datos continuo que puede ser interceptado si no se emplean protocolos seguros.

Gestión de Identidades

Cada servicio puede requerir diferentes niveles de acceso, lo que complica la implementación de autenticación y autorización centralizadas.

Amenazas Internas y Externas

Al ser sistemas distribuidos, los microservicios son susceptibles a ataques internos (como accesos no autorizados) y externos (como DDoS o suplantación).

Estrategias y Herramientas para Fortalecer la Seguridad

La comunidad tecnológica ha desarrollado una serie de estrategias y herramientas para abordar estos desafíos y garantizar la seguridad en entornos de microservicios:

Uso de API Gateway

Este componente actúa como un intermediario entre los clientes y los microservicios, centralizando la gestión de autenticación, autorización y enrutamiento. Herramientas como Kong y Apigee son ampliamente utilizadas en la industria.

Autenticación y Autorización con OAuth2

Este estándar permite gestionar el acceso de manera segura mediante tokens, asegurando que cada solicitud a un servicio sea validada previamente. OAuth2 facilita la integración con sistemas externos como redes sociales y servicios en la nube.

Protocolos de Cifrado (TLS)

El uso de TLS (Transport Layer Security) asegura que la comunicación entre servicios esté protegida contra interceptaciones y manipulaciones. Este protocolo es esencial para proteger datos sensibles en tránsito.

Monitoreo y Detección de Amenazas

Herramientas como Prometheus y Grafana permiten monitorear el comportamiento de los servicios en tiempo real, detectando anomalías que puedan indicar intentos de ataque. Adicionalmente, servicios como AWS GuardDuty o Azure Security Center ofrecen protección avanzada contra amenazas.

Patrones de Seguridad en Microservicios

Además de las herramientas, existen patrones arquitectónicos diseñados específicamente para abordar problemas de seguridad en sistemas distribuidos:

- **Circuit Breaker:** Previene fallos en cascada al desconectar temporalmente servicios que están fallando o sobrecargados.
- **Token-Based Authentication:** Garantiza que cada solicitud esté acompañada de un token válido emitido por un sistema de confianza.
- **Rate Limiting:** Limita el número de solicitudes que un cliente puede hacer a un servicio, mitigando ataques de fuerza bruta o DDoS.

Importancia de la Seguridad como Pilar Estratégico

En un mundo donde los ciberataques son cada vez más sofisticados, garantizar la seguridad de una arquitectura de microservicios no solo es un requisito técnico, sino también una necesidad estratégica. La implementación de estas estrategias y herramientas no solo protege los sistemas de posibles amenazas, sino que también fortalece la confianza de los usuarios y socios comerciales. .

...

Ventajas de los Microservicios: Modernización y Eficiencia

Los microservicios han transformado la manera en que las organizaciones diseñan y operan sus sistemas tecnológicos, ofreciendo ventajas significativas frente a las arquitecturas monolíticas tradicionales. Su diseño modular, capacidad de despliegue independiente y escalabilidad los convierten en una solución ideal para sistemas complejos que requieren flexibilidad y adaptabilidad.

1. Escalabilidad Dinámica

Una de las mayores ventajas de los microservicios es la capacidad de escalar servicios individuales según las necesidades del sistema. En lugar de escalar toda la aplicación, como sucede en los sistemas monolíticos, los microservicios permiten ajustar recursos únicamente en los componentes que enfrentan mayor demanda.

Ejemplo práctico: En una aplicación de comercio electrónico, durante promociones masivas, solo los servicios relacionados con el procesamiento de pagos o la gestión de inventario pueden ser escalados, optimizando los recursos y reduciendo costos.

2. Despliegue Independiente

Los microservicios permiten que cada componente sea desplegado de forma autónoma, sin necesidad de interrumpir el sistema completo. Esto acelera el proceso de desarrollo y facilita la incorporación de nuevas funcionalidades.

Beneficio clave: Actualizaciones, correcciones de errores o mejoras pueden ser implementadas rápidamente sin afectar la experiencia del usuario ni generar tiempo de inactividad.

3. Modularidad y Mantenimiento

El diseño modular de los microservicios facilita el mantenimiento y la actualización del sistema. Cada servicio tiene una función específica, lo que permite a los equipos de desarrollo concentrarse en problemas concretos sin afectar otras partes de la aplicación.

Impacto en la productividad: Equipos multidisciplinarios pueden trabajar en paralelo en diferentes servicios, reduciendo los tiempos de desarrollo y entrega.

4. Flexibilidad Tecnológica

Los microservicios permiten utilizar diferentes tecnologías y lenguajes de programación para cada servicio, según las necesidades específicas. Esto ofrece la libertad de adoptar soluciones tecnológicas innovadoras sin comprometer la funcionalidad general del sistema.

Ejemplo: Un sistema puede combinar servicios escritos en Java, Python y Node.js, utilizando bases de datos SQL y NoSQL, según los requerimientos de cada módulo.

5. Tolerancia a Fallos

En sistemas monolíticos, un fallo en un componente puede interrumpir toda la aplicación. En cambio, los microservicios están diseñados para ser independientes, lo que significa que un fallo en un servicio no afectará a los demás.

Caso práctico: Si un servicio de envío de correos electrónicos falla, el resto de la aplicación (como el procesamiento de pedidos) puede seguir funcionando normalmente.

6. Aceleración de la Innovación

Gracias a su arquitectura modular y despliegue independiente, los microservicios permiten a las organizaciones innovar más rápidamente. Nuevas funcionalidades pueden ser diseñadas, probadas y desplegadas sin interrumpir las operaciones existentes.

Beneficio estratégico: Las empresas pueden responder con mayor agilidad a las demandas del mercado y mantener una ventaja competitiva.

7. Compatibilidad con Metodologías Ágiles

Los microservicios son ideales para entornos de desarrollo que emplean metodologías ágiles. Su estructura modular facilita la implementación de ciclos cortos de desarrollo, pruebas y despliegue continuo.

Enfoque iterativo: Equipos pequeños pueden trabajar en iteraciones rápidas, mejorando constantemente los servicios y adaptándose a los cambios.

Consideraciones Finales sobre las Ventajas

Aunque los microservicios ofrecen numerosas ventajas, su implementación requiere una planificación cuidadosa para garantizar que las ganancias superen los costos iniciales. La inversión en herramientas de orquestación, monitoreo y seguridad es esencial para aprovechar al máximo esta arquitectura. En un mundo donde la velocidad, la flexibilidad y la eficiencia son esenciales, los microservicios se posicionan como una herramienta clave para la modernización tecnológica.

...

Resiliencia en Microservicios: Garantizando la Continuidad Operativa

La resiliencia es un aspecto crítico en la arquitectura de microservicios, ya que los sistemas distribuidos son inherentemente más complejos y susceptibles a fallos. Este concepto se refiere a la capacidad de un sistema para recuperarse rápidamente de errores y mantener la operatividad, incluso bajo condiciones adversas. Los microservicios, gracias a su diseño modular e independiente, ofrecen una base sólida para implementar patrones y estrategias que fortalecen la continuidad operativa.

1. Separación de Responsabilidades

En una arquitectura de microservicios, cada servicio tiene una responsabilidad bien definida y se diseña de manera independiente. Esto minimiza el impacto de un fallo en un servicio sobre el resto del sistema, ya que los problemas se aíslan rápidamente.

Ejemplo: Si un servicio de notificaciones deja de funcionar, el sistema puede seguir procesando pedidos, ya que los servicios restantes no dependen directamente del primero.

2. Patrones de Tolerancia a Fallos

Los microservicios permiten implementar patrones avanzados que aseguran la continuidad operativa ante errores temporales o fallos graves:

- **Circuit Breaker:** Este patrón desconecta temporalmente un servicio que está fallando o tardando demasiado en responder, protegiendo al resto del sistema de ser afectado por este problema.
- **Retries Automáticos:** Permiten reintentar una operación fallida después de un intervalo de tiempo predefinido, lo que puede ser útil en casos de fallos transitorios.
- **Fallbacks:** Se diseñan rutas alternativas o respuestas predeterminadas para garantizar que los usuarios no se enfrenten a errores críticos.

3. Escalabilidad Resiliente

Los microservicios pueden escalarse de forma horizontal, aumentando el número de instancias de un servicio específico para manejar una mayor carga. Esto no solo mejora

el rendimiento, sino que también refuerza la resiliencia al distribuir las solicitudes entre múltiples instancias, evitando puntos únicos de fallo.

Caso práctico: Durante un evento de alta demanda, como un Black Friday, un servicio de procesamiento de pagos puede escalar rápidamente para manejar millones de transacciones sin interrupciones.

4. Monitoreo y Diagnóstico en Tiempo Real

El monitoreo activo es esencial para identificar problemas antes de que afecten significativamente al sistema. Herramientas como Prometheus, Grafana y ELK Stack permiten rastrear métricas clave, analizar logs y generar alertas en tiempo real.

Ventaja clave: Con estos datos, los equipos pueden diagnosticar y resolver problemas de manera proactiva, reduciendo el tiempo de inactividad.

5. Gestión de Estados Distribuidos

Un desafío común en los sistemas distribuidos es gestionar estados compartidos entre servicios. Los microservicios resuelven este problema mediante estrategias como:

- **Bases de datos independientes por servicio:** Esto reduce los conflictos de acceso y asegura que cada servicio sea autónomo.
- **Eventos asíncronos:** Utilizando colas de mensajes como RabbitMQ o Kafka, los servicios pueden comunicarse de manera eficiente sin depender de respuestas inmediatas.

6. Recuperación ante Desastres

Los microservicios facilitan la implementación de estrategias de recuperación ante desastres, como la replicación geográfica y los despliegues en múltiples regiones. Esto asegura que, incluso en caso de fallos a gran escala, el sistema pueda recuperarse rápidamente.

Ejemplo: Un sistema basado en microservicios puede desplegar instancias en diferentes centros de datos, garantizando disponibilidad incluso ante un fallo catastrófico en una ubicación.

Conclusión sobre la Resiliencia

La resiliencia es un pilar fundamental para el éxito de los microservicios en entornos distribuidos. La combinación de patrones de tolerancia a fallos, herramientas de monitoreo y estrategias de recuperación robustas garantiza que los sistemas basados en esta arquitectura puedan operar de manera confiable, incluso bajo condiciones adversas. Aunque requiere una inversión inicial en herramientas y diseño, la resiliencia asegura la continuidad operativa y la satisfacción del usuario, lo que la convierte en un elemento clave para organizaciones que buscan modernizarse y competir en un entorno dinámico y exigente.

...

Conclusión

Los microservicios se han consolidado como una solución arquitectónica clave para la modernización de sistemas en entornos empresariales que demandan escalabilidad, flexibilidad y resiliencia. Su enfoque modular e independiente permite abordar las limitaciones de las arquitecturas monolíticas tradicionales, ofreciendo ventajas significativas en términos de despliegue, mantenimiento y adaptación tecnológica. Además, su capacidad para trabajar con tecnologías heterogéneas y escalar servicios de forma individual los convierte en una opción ideal para organizaciones que buscan optimizar sus operaciones y responder rápidamente a las demandas del mercado.

En cuanto a la seguridad, los microservicios plantean desafíos que exigen soluciones robustas como el uso de **API Gateway**, protocolos de cifrado como **TLS** y estándares de autenticación como **OAuth2**. Estas herramientas, combinadas con patrones de diseño como **Circuit Breaker** y estrategias de monitoreo en tiempo real, garantizan la protección de datos y la continuidad operativa en sistemas distribuidos. Aunque implementar estas medidas puede requerir una inversión significativa en tiempo y recursos, los beneficios a largo plazo superan ampliamente los costos iniciales.

Por otro lado, la resiliencia es otro aspecto fundamental que destaca en los microservicios. La capacidad de aislar fallos, reintentar operaciones fallidas y escalar servicios de forma dinámica asegura que el sistema pueda adaptarse rápidamente a cambios y fallos inesperados. Este enfoque no solo mejora la confiabilidad del sistema, sino que también refuerza la confianza de los usuarios y socios comerciales.

No obstante, adoptar microservicios implica enfrentarse a ciertos retos, como la gestión de la comunicación entre servicios, la necesidad de herramientas avanzadas de monitoreo y la capacitación del equipo técnico. Las organizaciones que buscan implementar esta arquitectura deben considerar cuidadosamente su infraestructura tecnológica, su capacidad organizativa y los objetivos específicos que desean alcanzar.

En conclusión, los microservicios no solo representan una arquitectura tecnológica, sino también un enfoque estratégico para construir sistemas modernos, seguros y resilientes. Al ofrecer una combinación única de flexibilidad, eficiencia y tolerancia a fallos, esta arquitectura permite a las organizaciones mantenerse competitivas en un entorno tecnológico en constante evolución. Sin embargo, su implementación exitosa requiere una planificación rigurosa, el uso de herramientas adecuadas y un compromiso continuo con la innovación y la mejora. Al superar estos desafíos, los microservicios se posicionan como una herramienta esencial para la transformación digital y el éxito empresarial a largo plazo.

...

Referencias

Bibliografía • Mejía, C., Noel, D. (2022). Arquitectura de micro servicios para mejorar la calidad de software en una entidad bancaria de Lima metropolitana. • Suárez, K. T. G., Anaya, R., Cano, A. F. (2017). Un acercamiento a los micro servicios. UNA-CIENCIA: Revista de Estudios e Investigaciones, 10(19), 116-126. • Toledo Azorza, M. A. J. (2017). Comparación del rendimiento de las arquitecturas monolíticas y microservicios en los sistemas web. • Moreno Belinchón, D. (2018). Despliegue de microservicios mediante técnicas de virtualización ligeras basadas en contenedores. • Colombero, V. T., Estelles, J. P., Gallegos, L., Lopez, P. (2024). Monolitos vs. Microservicios en

Arquitectura de Software: Perspectivas para un Desarrollo Eficiente. Memorias de las JAIIO, 10(5), 42-54. • Guerrero Martínez, D. S. (2023). Estrategia para la implementación de modelos de aprendizaje automático (machine learning) usando arquitecturas basadas en microservicios en el contexto de la seguridad en ciudades inteligentes. • Rodríguez, Z. M., Rodríguez, L. D. P., Suarez, J. C. G. (2020). Arquitectura basada en Microservicios y DevOps para una ingeniería de software continua. Industrial Data, 23(2), 141-149. • Jiménez Aliaga, C. (2019). Arquitecturas de microservicios para aplicaciones desplegadas en contenedores.