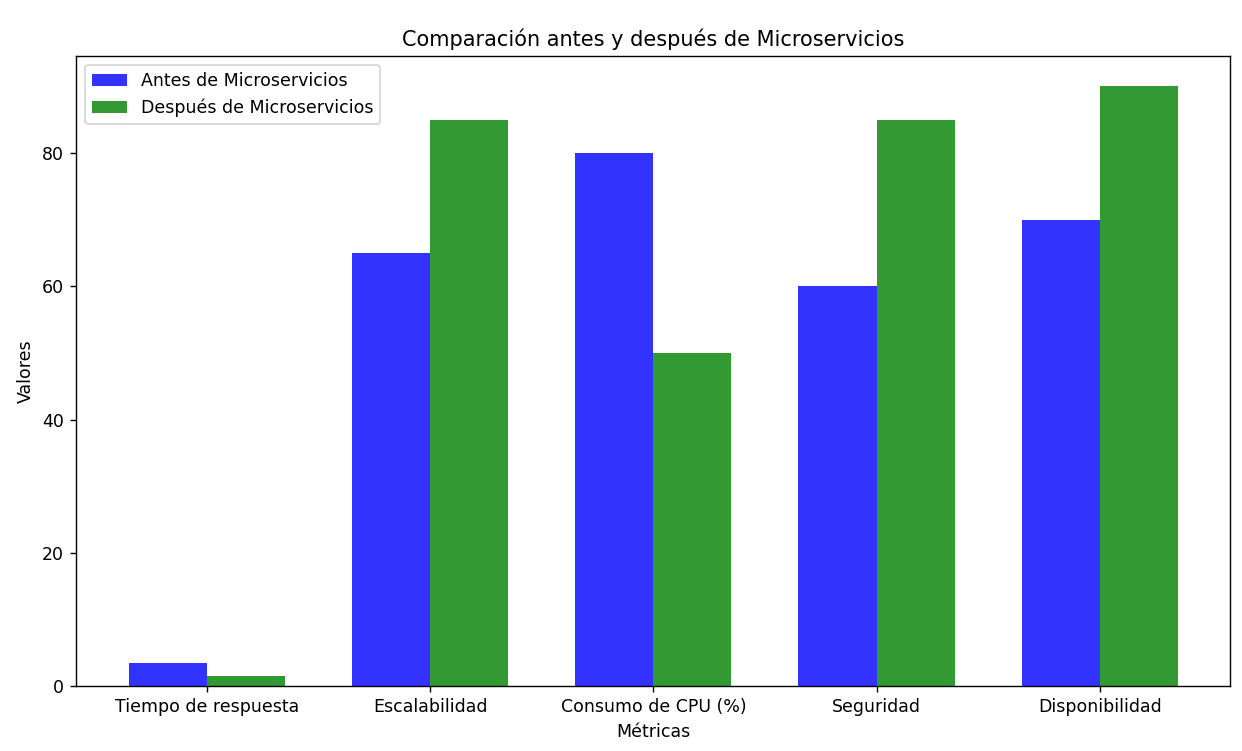
1. **Arquitectura de microservicios para mejorar la calidad de software en una entidad bancaria de lima**: Este documento aborda la implementación de una arquitectura de microservicios en una entidad bancaria en Lima con el objetivo de mejorar la calidad del software. La investigación destaca cómo los microservicios permiten la escalabilidad y flexibilidad en el desarrollo de software, lo que resulta en una mayor eficiencia y un menor tiempo de respuesta en comparación con las arquitecturas monolíticas tradicionales. Se examinan los desafíos que enfrentan las organizaciones al adoptar microservicios, como la gestión de la complejidad y la seguridad, y se presentan soluciones prácticas para superarlos. Este trabajo es valioso para cualquier organización que busque mejorar la calidad de su software mediante la adopción de arquitecturas modernas y más eficientes.

**Bibliografía:** Mejía, C., & Noel, D. Arquitectura de micro servicios para mejorar la calidad de software en una entidad bancaria de Lima metropolitana, 2022**.**

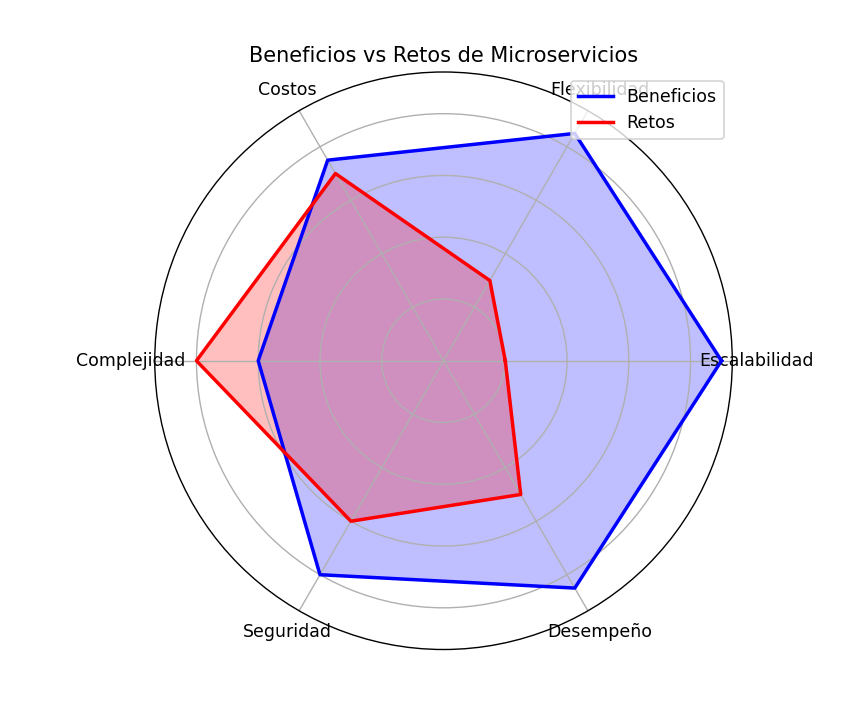


**Reflexión**

La implementación de una arquitectura de microservicios en una entidad bancaria en Lima resalta la importancia de la escalabilidad y la flexibilidad en el desarrollo de software. Estos beneficios son cruciales para mejorar la eficiencia operativa y reducir los tiempos de respuesta, algo esencial en un entorno financiero donde cada segundo cuenta. Sin embargo, esta transición no está exenta de desafíos. La gestión de la complejidad introducida por múltiples servicios independientes puede ser abrumadora, y la seguridad se convierte en una prioridad crítica. Las organizaciones deben adoptar un enfoque metódico y práctico para abordar estos retos, asegurando que la migración a microservicios no solo cumpla con las expectativas de rendimiento, sino que también proteja los datos sensibles de los clientes.

02-**UNACIENCIA: Un acercamiento a los microservicios:** Este artículo ofrece una visión general de la arquitectura de microservicios, destacando su emergencia como una tendencia clave en el desarrollo de aplicaciones web. Los autores explican cómo los microservicios permiten descomponer una aplicación en servicios pequeños y autónomos que pueden ser desplegados, evolucionados y escalados de manera independiente, lo que mejora la flexibilidad y la escalabilidad del sistema. Además, se comparan los microservicios con otras arquitecturas, como SOA y las arquitecturas monolíticas, señalando las ventajas y desafíos de cada enfoque. Este documento es útil para desarrolladores y arquitectos de software interesados en entender los beneficios y retos de adoptar micro servicios en sus proyectos

**Bibliografía: Suárez, K. T. G., Anaya, R., & Cano, A. F. (2017). Un acercamiento a los micro servicios. UNACIENCIA: Revista de Estudios e Investigaciones, 10(19), 116-126.**

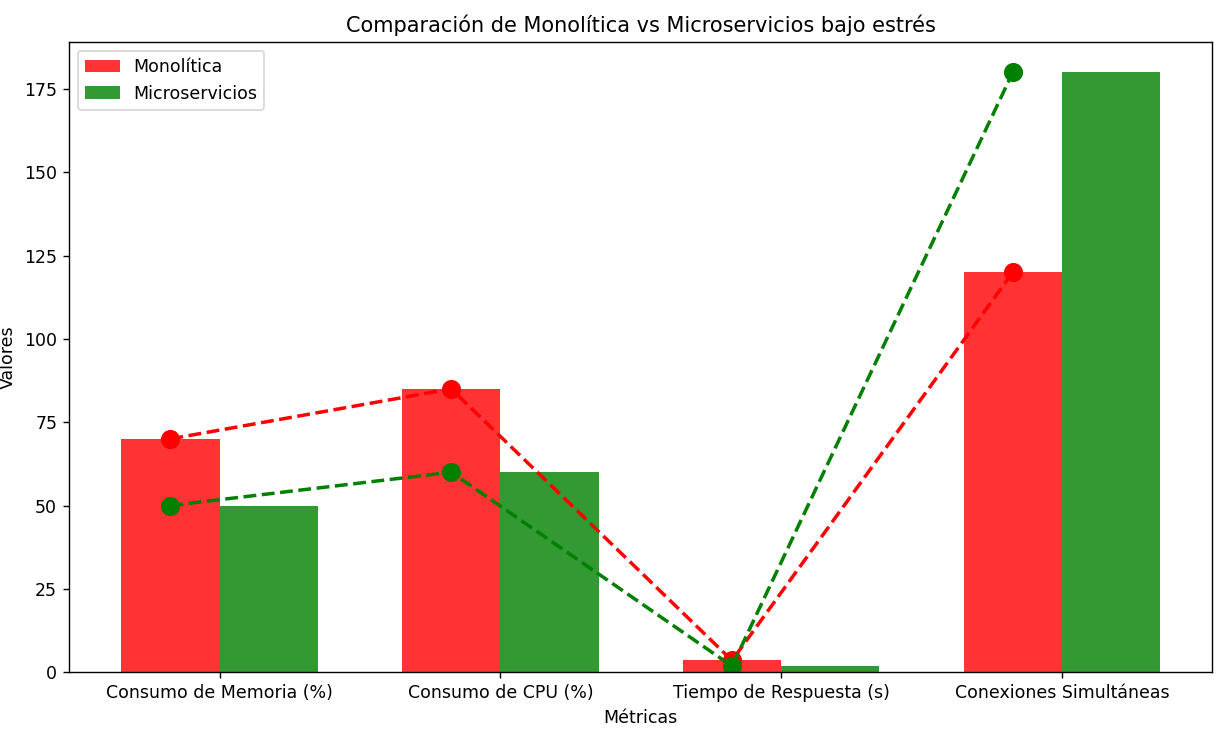


**Reflexión.**

El artículo sobre micro servicios subraya cómo esta arquitectura está transformando el desarrollo de aplicaciones web. La capacidad de descomponer aplicaciones en servicios pequeños y autónomos facilita la escalabilidad y mejora la flexibilidad, permitiendo que las organizaciones respondan rápidamente a las demandas del mercado. Este modularidad también permite a los equipos de desarrollo trabajar en diferentes componentes de forma independiente, lo que fomenta la innovación. Sin embargo, también plantea el reto de gestionar la comunicación entre servicios, lo que puede aumentar la complejidad del sistema. Por lo tanto, es vital que los equipos adopten herramientas y prácticas adecuadas para monitorear y gestionar estas interacciones.

**03-Comparación del rendimiento de las arquitecturas monolíticas y microservicios en los sistemas web:** Esta tesis compara el rendimiento de las arquitecturas monolíticas y de microservicios en sistemas web, analizando cómo cada una maneja aspectos clave como el consumo de memoria, uso de CPU y tiempo de respuesta bajo condiciones de estrés. Los resultados muestran que la arquitectura de microservicios supera a la monolítica en eficiencia y fiabilidad, mejorando en un 10.63% y 27.87% respectivamente. La investigación sugiere que los microservicios ofrecen una solución superior en términos de escalabilidad y manejo de grandes volúmenes de tráfico, aunque también recomienda evaluar los costos y la viabilidad de su implementación antes de proceder. Este estudio es esencial para quienes buscan justificar o evaluar la transición a microservicios en sistemas web

**Bibliografía: Toledo Azorza, M. A. J. (2017). Comparación del rendimiento de las arquitecturas monolíticas y microservicios en los sistemas web.**

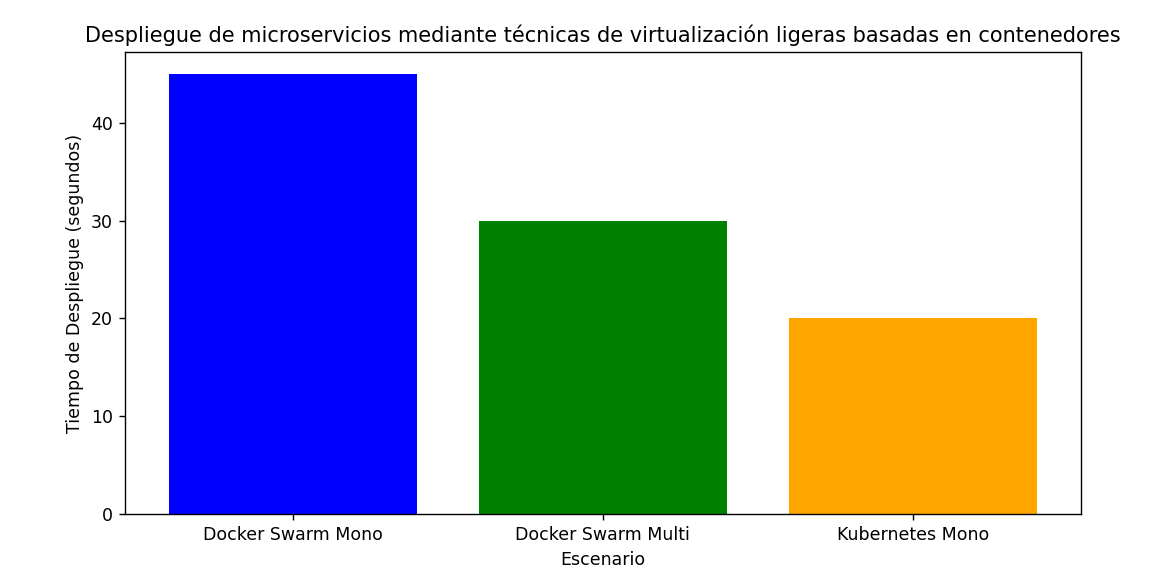


**Reflexión.**

El estudio que compara el rendimiento de arquitecturas monolíticas y de microservicios pone de relieve cómo esta última puede superar a la primera en eficiencia y fiabilidad. Con mejoras significativas en el consumo de recursos y el tiempo de respuesta, los microservicios se presentan como una solución viable para manejar grandes volúmenes de tráfico, lo cual es crucial en el contexto actual de demanda digital. Sin embargo, la investigación también advierte sobre la necesidad de evaluar los costos asociados con la implementación de microservicios. Esto sugiere que, aunque los beneficios son claros, las organizaciones deben realizar un análisis de costo-beneficio exhaustivo antes de embarcarse en esta transición.

**04- Despliegue de microservicios mediante técnicas de virtualización ligeras basadas en contenedores:** Este trabajo analiza el despliegue de aplicaciones y funciones de red virtualizadas (VNF) utilizando técnicas de contenedores ligeros. El enfoque principal es la arquitectura de microservicios y la virtualización basada en contenedores como Docker y Kubernetes. Proporciona casos de uso y escenarios prácticos donde se aplican estas tecnologías, enfocándose en la escalabilidad, agilidad y eficiencia en el uso de recursos.

**Bibliografía: Moreno Belinchón, D. (2018). Despliegue de microservicios mediante técnicas de virtualización ligeras basadas en contenedores.**

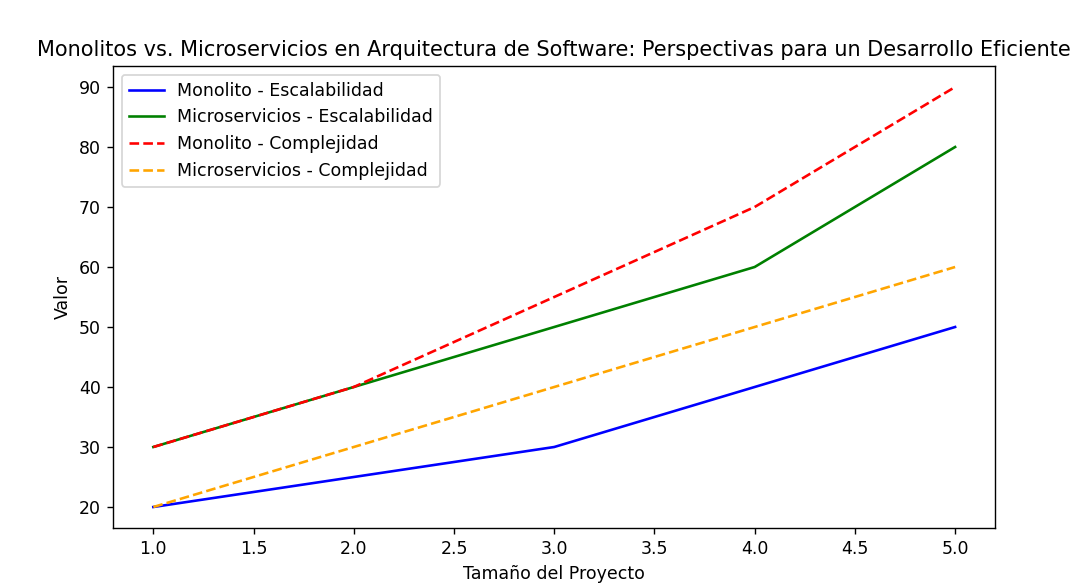


**Reflexión**

El trabajo sobre el despliegue de microservicios mediante contenedores destaca la importancia de la virtualización ligera, especialmente en el contexto de herramientas como Docker y Kubernetes. Estas tecnologías permiten una mayor agilidad y eficiencia en el uso de recursos, lo que es esencial para entornos que requieren escalabilidad. La capacidad de desplegar y gestionar aplicaciones en contenedores mejora la resiliencia y la portabilidad del software. No obstante, esto también introduce nuevos desafíos relacionados con la orquestación y la gestión de redes, que las organizaciones deben abordar para aprovechar al máximo estas herramientas.

**05-Monolitos vs. Microservicios en Arquitectura de Software:** Perspectivas para un Desarrollo Eficiente Este estudio compara las arquitecturas monolíticas y de microservicios, analizando sus ventajas, desventajas y los escenarios en los que cada una es más eficiente. Se examinan casos de empresas como Amazon y eBay que han migrado de una arquitectura a otra, explorando las dificultades y beneficios en términos de escalabilidad, mantenimiento y desarrollo.

**Bibliografía: 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.**



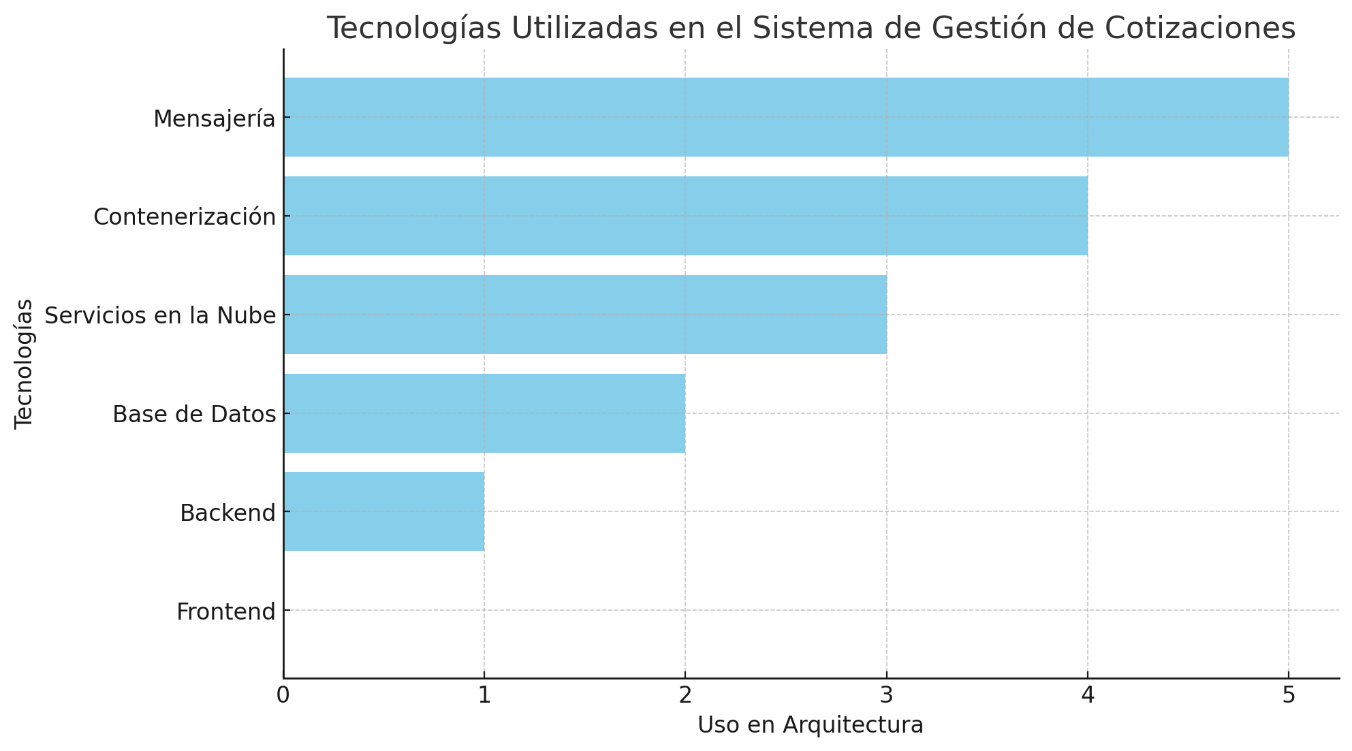
**Reflexión**

Perspectivas para un Desarrollo Eficiente Este estudio compara las arquitecturas monolíticas y de microservicios, analizando sus ventajas, desventajas y los escenarios en los que cada una es más eficiente. Se examinan casos de empresas como Amazon y eBay que han migrado de una arquitectura a otra, explorando las dificultades y beneficios en términos de escalabilidad, mantenimiento y desarrollo.

**06-Sistema de gestión de cotizaciones de servicios para empresas de telecomunicaciones mediante microservicios**

El artículo trata sobre el desarrollo de un sistema de gestión de cotizaciones de servicios para empresas de telecomunicaciones utilizando una arquitectura basada en microservicios. Este sistema busca optimizar el proceso de cotización, que implica la gestión de recursos compartidos como redes, torres de telefonía, y servicios de fibra óptica y roaming. La solución se desarrolló utilizando tecnologías como Kafka para la mensajería, Angular 15 para el frontend, Java 17 para el backend, Oracle para la base de datos, Azure para los servicios en la nube y Docker para la contenedorización. La arquitectura de microservicios permitió mejoras en la escalabilidad, disponibilidad, y experiencia del usuario.

**Bibliografía: Sanchez Matos, E., & Ore Quintana, M. A. Sistema de gestión de cotizaciones de servicios para empresas de telecomunicaciones mediante microservicios**.

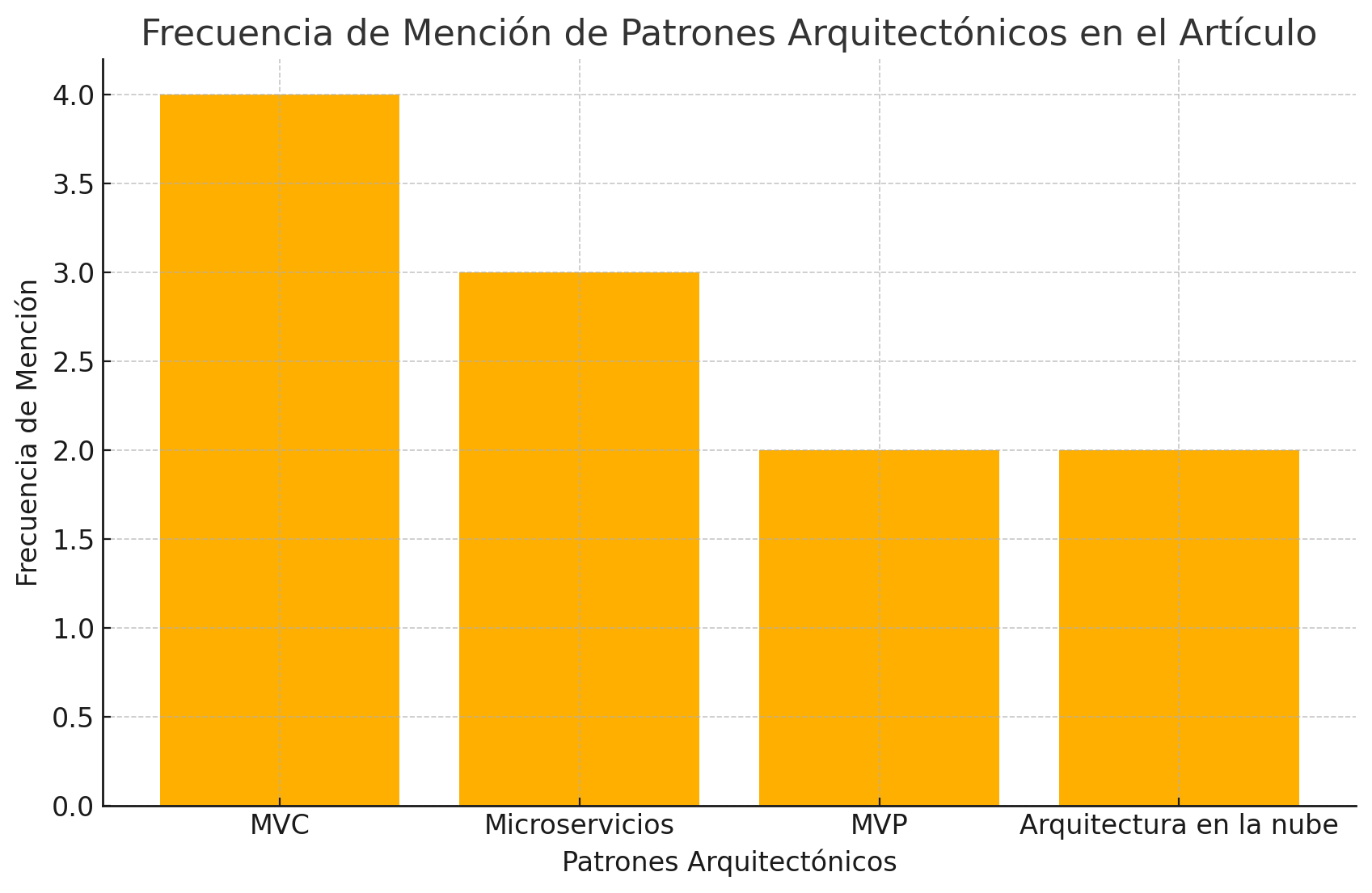


**Reflexión**

El uso de microservicios representa una evolución significativa en el diseño y desarrollo de sistemas de software, especialmente para sectores tan dinámicos y colaborativos como el de las telecomunicaciones. La fragmentación en componentes autónomos proporciona una flexibilidad que es esencial para enfrentar la complejidad de la colaboración entre empresas y gestionar de manera efectiva las cotizaciones de servicios. Además, al utilizar tecnologías modernas como Kafka, Docker y Azure, se garantiza que el sistema sea escalable y tenga una alta disponibilidad, lo que es crucial para mantener la competitividad en un mercado en crecimiento.

**07-Marco de Trabajo para Seleccionar un Patrón Arquitectónico en el Desarrollo de Software:** Este documento presenta un framework diseñado para seleccionar el patrón arquitectónico más adecuado en el desarrollo de software. El enfoque principal es resolver problemas de desacoplamiento en la lógica empresarial, la interfaz de usuario, la navegación y la arquitectura de la información, todo lo cual afecta la calidad del producto final. El marco propuesto se basa en la caracterización de patrones de arquitectura de software, soportado por una revisión de literatura y consultas a expertos. Se identifican patrones con un alto impacto en el sector productivo y se proporciona una guía para seleccionar el patrón adecuado según el tipo de proyecto de desarrollo y sus necesidades específicas

**Bibliografía: Mejía, J. C. G., Agudelo, F. A. V., & Gil, K. G. (2021). Marco de Trabajo para Seleccionar un Patrón Arquitectónico en el Desarrollo de Software. *Revista Ibérica de Sistemas e Tecnologias de Informação*, (E43), 568-581.**

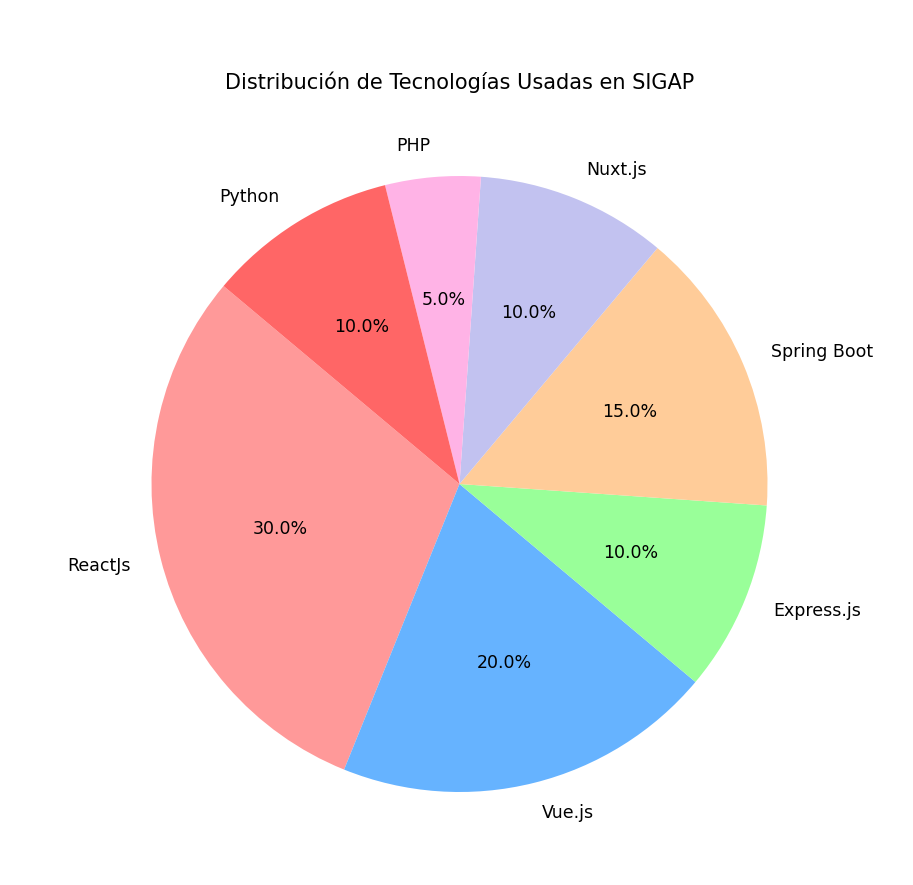
****

**Reflexión**

El marco presentado para seleccionar patrones arquitectónicos en el desarrollo de software es un recurso valioso para las organizaciones que buscan mejorar la calidad de sus productos. Al abordar problemas de desacoplamiento y diseño, este marco puede ayudar a los equipos a tomar decisiones informadas que impacten positivamente en la eficiencia y efectividad del software. La caracterización de patrones y la consulta a expertos brindan una base sólida para esta selección, pero es fundamental que las organizaciones no solo sigan un marco, sino que también consideren sus necesidades específicas y contextos únicos.

**08-Arquitectura Basada en Microservicios y DevOps para una Ingeniería de Software Continua:** Este documento propone una arquitectura basada en microservicios y DevOps para lograr una ingeniería de software continua. Los microservicios permiten desarrollar aplicaciones a partir de servicios independientes, escalables y colaborativos, mientras que DevOps facilita la entrega e integración continua en un entorno ágil y colaborativo. El documento incluye un caso de estudio donde se aplica esta arquitectura en un proyecto académico, resaltando la implementación de tecnologías de vanguardia y la cultura colaborativa impulsada por DevOps. El enfoque está en la automatización y escalabilidad, cruciales para enfrentar los desafíos actuales en la industria del software

**Bibliografía: 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.**

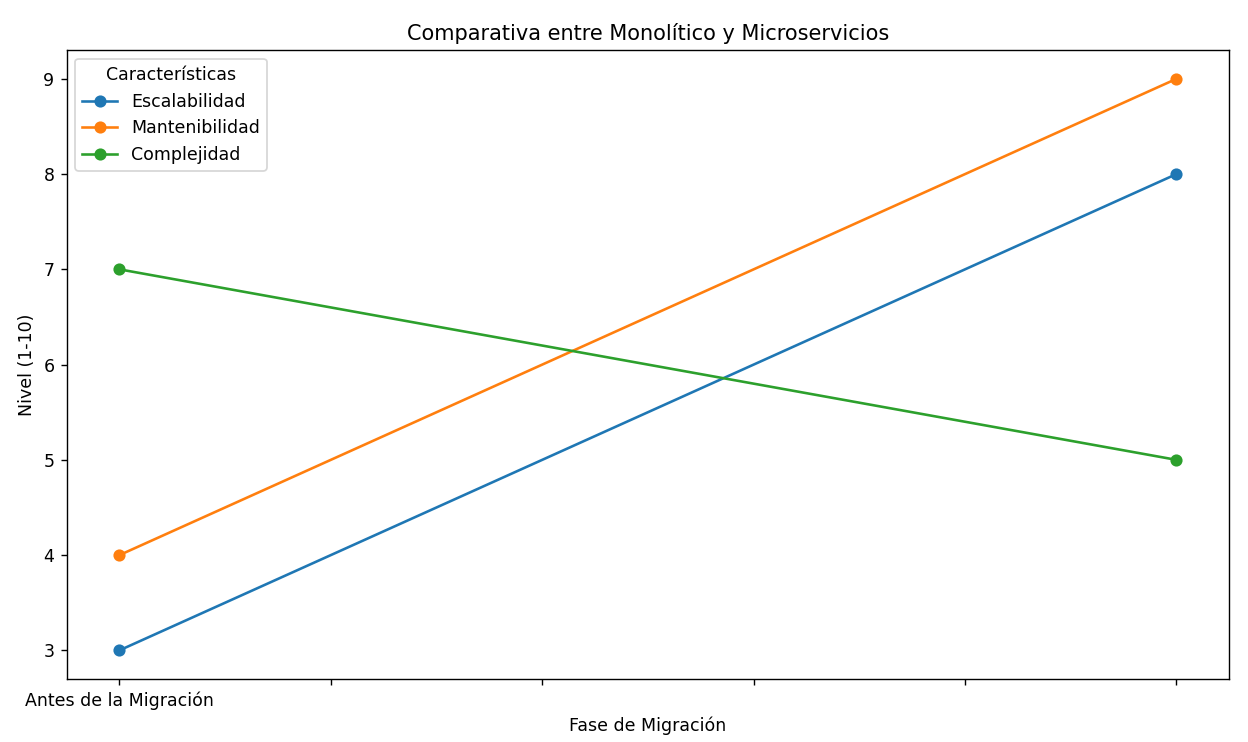


**Reflexión**

La sinergia entre microservicios y DevOps es crucial para lograr una ingeniería de software continua y efectiva. La capacidad de desarrollar aplicaciones a partir de servicios independientes, combinada con prácticas de integración y entrega continua, promueve una cultura de colaboración y rapidez en la entrega. Este enfoque no solo mejora la calidad del software, sino que también permite a las organizaciones adaptarse rápidamente a los cambios del mercado. Sin embargo, la implementación exitosa de esta sinergia requiere un cambio cultural dentro de la organización, donde la colaboración entre equipos de desarrollo y operaciones sea la norma.

**09- Metodología para la Migración de Aplicaciones Monolíticas a Sistemas Basados en Microservicios:** Este documento detalla una metodología para migrar aplicaciones monolíticas a arquitecturas basadas en microservicios. La propuesta aborda las limitaciones de los sistemas monolíticos, como la dificultad para escalar y mantener, y describe un proceso de migración estructurado para transformar una aplicación existente en un conjunto de microservicios. La metodología incluye la identificación de componentes, la creación de servicios independientes y la implementación de prácticas de DevOps para asegurar la integración y entrega continua, facilitando la adaptación a infraestructuras modernas como la nube

**Bibliografía: Peña Huérfano, L. D. Metodología para la migración de aplicaciones monolíticas a sistemas basados en microservicios.**

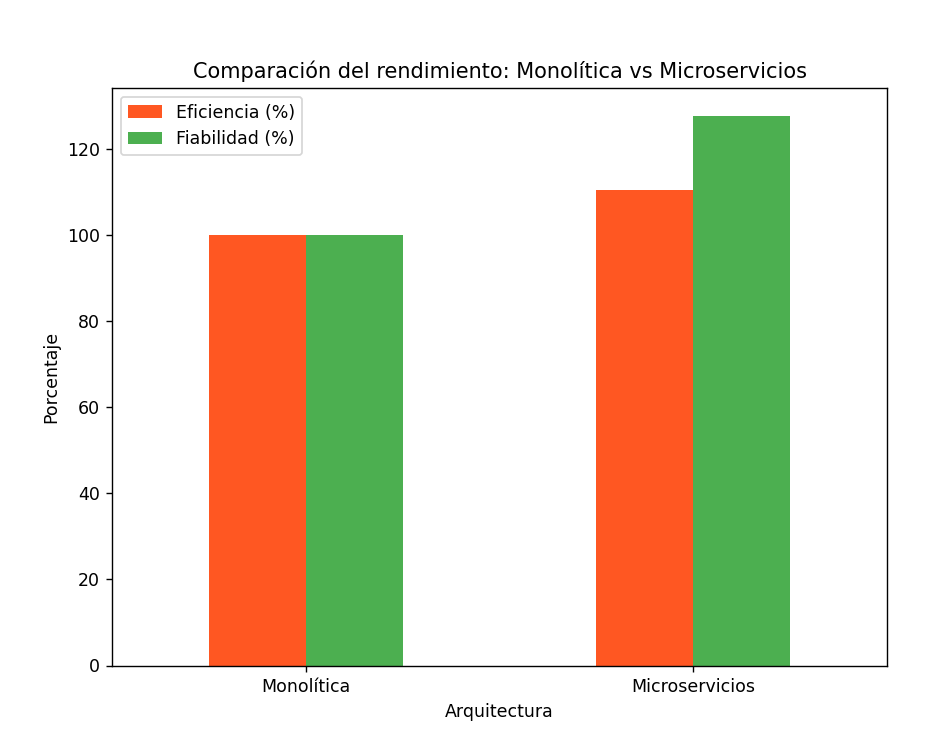


**Reflexión**

La exploración de arquitecturas de microservicios revela su potencial para transformar la forma en que las organizaciones desarrollan y gestionan software. Sin embargo, esta transición conlleva desafíos significativos que deben ser abordados de manera proactiva. La planificación cuidadosa, la capacitación del personal y la implementación de prácticas adecuadas son esenciales para maximizar los beneficios de los micro servicios. A medida que más organizaciones se aventuran en esta transformación, se hace evidente que el éxito radica no solo en la adopción de nuevas tecnologías, sino en la capacidad de las organizaciones para adaptarse y evolucionar en un entorno tecnológico en constante cambio.

**10- Comparación del rendimiento de las arquitecturas monolíticas y microservicios en los sistemas web:** Este documento es una tesis que busca comparar el rendimiento de las arquitecturas monolíticas y de microservicios en sistemas web. A través de un estudio cuantitativo y preexperimental, se evaluaron 30 procesos similares en sistemas web para determinar cómo cada arquitectura afecta la eficiencia y la fiabilidad del sistema. Los resultados mostraron que la arquitectura de microservicios mejora la eficiencia en un 10.63% y la fiabilidad en un 27.87% en comparación con la arquitectura monolítica. Se concluye que los microservicios son más efectivos para sistemas web modernos y se recomienda investigar más sobre los costos y la viabilidad de implementar esta arquitectura. Nos puede ser útil para comprender cómo las arquitecturas de software influyen en el rendimiento de los sistemas web y para evaluar la transición de sistemas monolíticos a microservicios, especialmente en proyectos que requieren alta escalabilidad y fiabilidad

**Bibliografía: Toledo Azorza, M. A. J. (2017). Comparación del rendimiento de las arquitecturas monolíticas y microservicios en los sistemas web.**

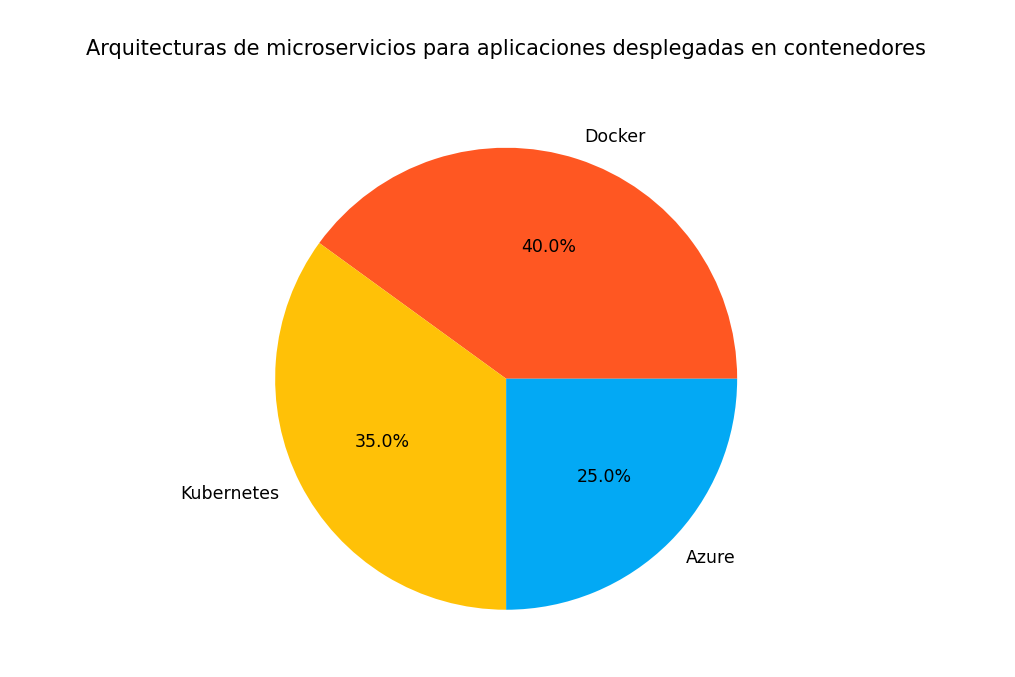


**Reflexión**

Este artículo presenta una guía estructurada para transformar sistemas monolíticos en microservicios. Destaca la importancia de descomponer aplicaciones en componentes independientes, lo que facilita el escalado y mantenimiento. La integración de prácticas de DevOps es fundamental, asegurando una entrega continua y adaptándose a entornos modernos como la nube. La claridad en este proceso es esencial para que las organizaciones puedan realizar transiciones efectivas.

**11- Arquitecturas de microservicios para aplicaciones desplegadas en contenedores**: Este trabajo de fin de grado se centra en la definición y despliegue de arquitecturas de microservicios utilizando contenedores, con un enfoque en la nube, específicamente en Microsoft Azure. El documento explora las buenas prácticas y patrones arquitectónicos para el desarrollo de aplicaciones escalables y de alta disponibilidad. Se utilizan tecnologías como .NET Core para el desarrollo y Docker como motor de contenedores, con Kubernetes como orquestador. El proyecto aborda cómo gestionar la escalabilidad, la replicación rápida y la implementación continua en entornos cloud, destacando la importancia de los contenedores y orquestadores para maximizar el potencial de los microservicios. Sirve mucho al momento de diseñar, implementar y escalar aplicaciones modernas basadas en microservicios, aprovechando tecnologías cloud y contenedores. Es particularmente útil si estás considerando un despliegue en la nube y quieres entender cómo orquestadores como Kubernetes pueden facilitar la gestión de aplicaciones distribuidas.

**Bibliografía: Jiménez Aliaga, C. (2019). Arquitecturas de microservicios para aplicaciones desplegadas en contenedores.**



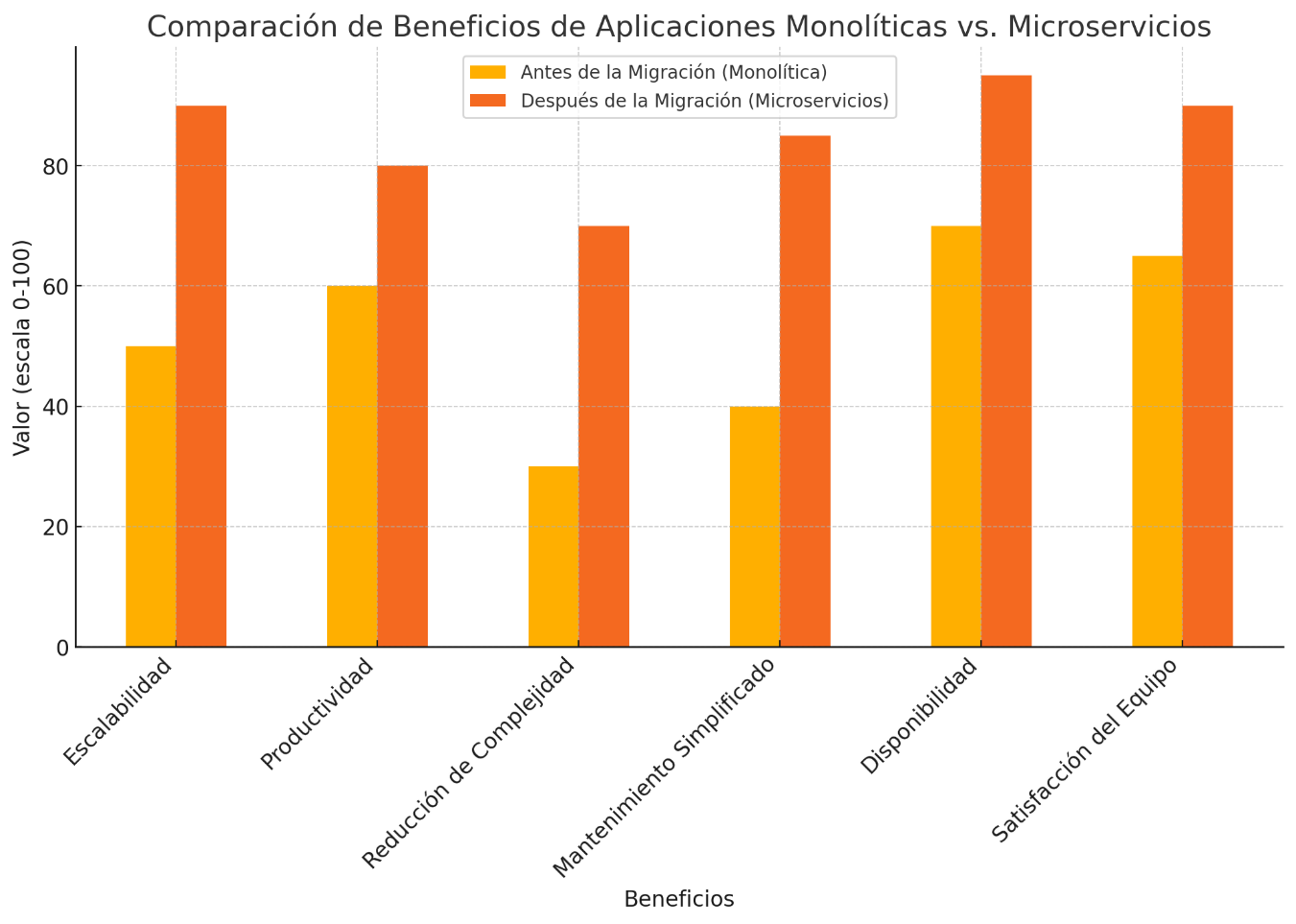
**Reflexión**

La tesis muestra que los microservicios superan a las arquitecturas monolíticas en eficiencia y fiabilidad, con mejoras significativas en rendimiento. Este hallazgo es clave para las empresas que consideran una migración, enfatizando que los microservicios son más adecuados para sistemas web que requieren escalabilidad. La evaluación de costos y viabilidad también se vuelve crucial, ya que el rendimiento mejorado debe ser balanceado con inversiones necesarias.

**Artículo 12: Metodología de conversión de aplicaciones monolíticas a microservicios desplegable en la nube para pequeñas empresas**

El documento describe una metodología para la conversión de aplicaciones monolíticas a una arquitectura de microservicios, específicamente para pequeñas empresas que buscan aprovechar las ventajas de la computación en la nube. Se presentan los pasos y actividades necesarias para llevar a cabo esta transformación, que incluyen el análisis de factibilidad, planeamiento, diseño, desarrollo y despliegue del software. La arquitectura de microservicios se plantea como una solución para problemas como la falta de escalabilidad y el difícil mantenimiento de las aplicaciones monolíticas.

**Bibliografía: German, M. L. J. (2022). Metodología de conversión de aplicaciones monolíticas a microservicios desplegable en la nube para pequeñas empresas**.

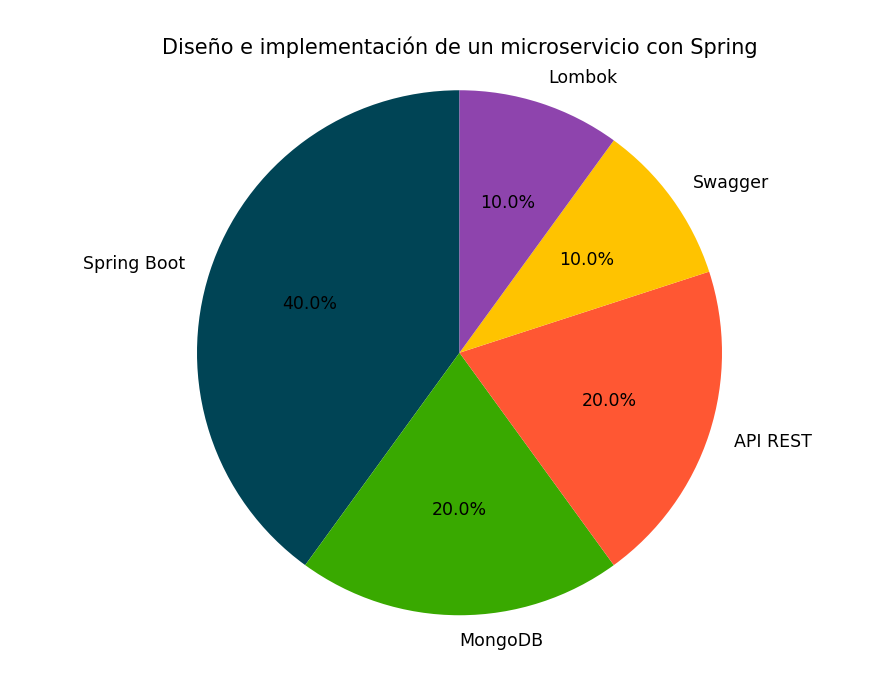


**Reflexión**

La transformación de aplicaciones monolíticas a microservicios representa una tendencia creciente entre pequeñas empresas, ya que permite lograr una mayor flexibilidad, escalabilidad y reducción de la complejidad en la gestión del software. Esta metodología es fundamental para lograr que las empresas se adapten a los cambios tecnológicos y mantengan la competitividad, aprovechando la nube para desplegar aplicaciones con alta disponibilidad y bajo costo. La migración a microservicios, aunque compleja, aporta beneficios clave en términos de productividad y sostenibilidad del desarrollo

**13- Diseño e implementación de un microservicio con Spring**: Este trabajo explora el desarrollo de aplicaciones web mediante microservicios usando tecnologías modernas como Spring Boot, Spring Data y MongoDB. Se implementa un microservicio que gestiona listas de reproducción de Spotify, filtrando las canciones escuchadas por el usuario. El proyecto analiza arquitecturas, patrones de diseño y frameworks, proporcionando una implementación eficiente de microservicios. También destaca las ventajas de usar Spring para simplificar el desarrollo y optimizar la funcionalidad de la aplicación

**Bibliografía: Rienda Iáñez, J. (2019). Diseño e implementación de un microservicio con Spring.**

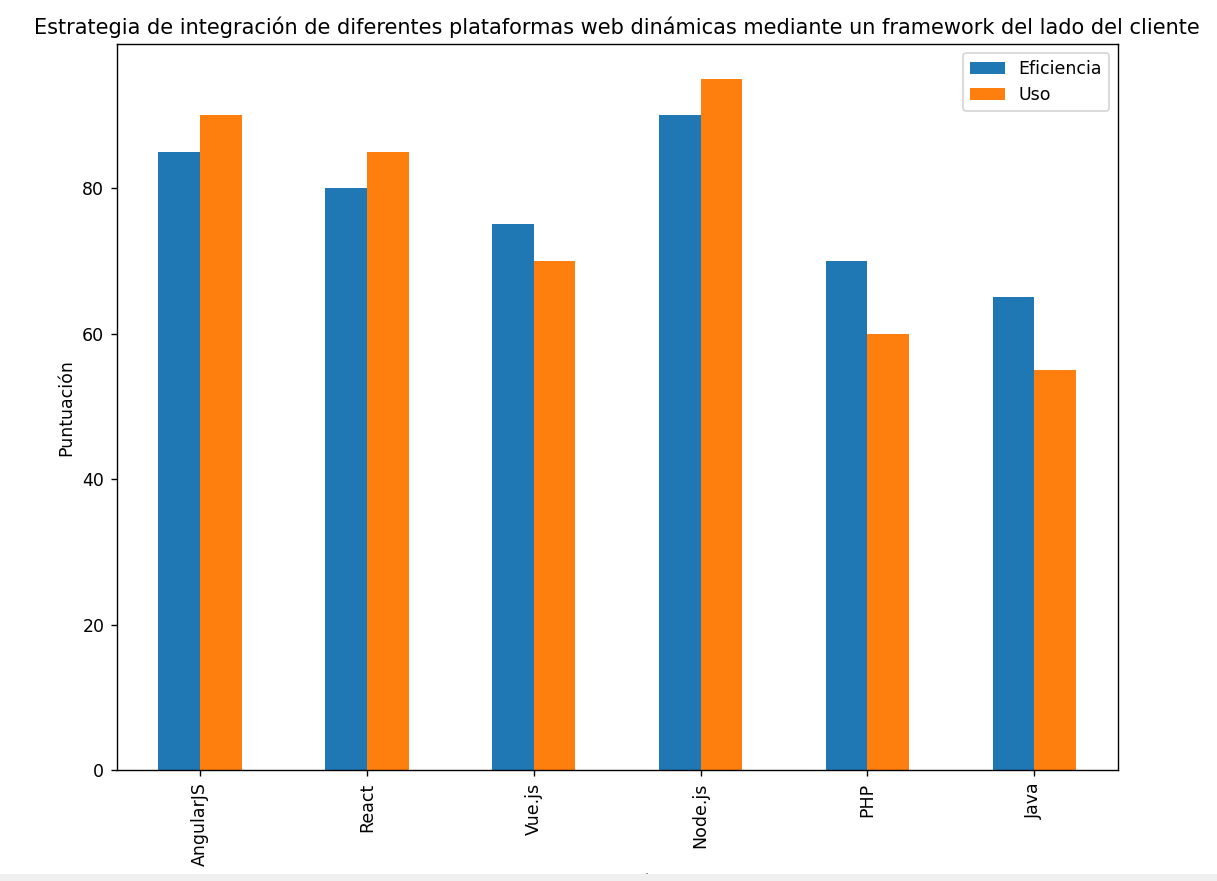


**Reflexión**

El enfoque en la seguridad en entornos de IoT es crítico, y la integración de TLS y OAuth2 en arquitecturas de microservicios ofrece soluciones a problemas de autenticación y confidencialidad. Este artículo resalta la flexibilidad de los microservicios para adaptarse a diferentes capacidades de dispositivos, lo que es esencial en un mundo donde la conectividad sigue expandiéndose.

**14- Estrategia de integración de diferentes plataformas web dinámicas mediante un framework del lado del cliente:** Este documento propone una estrategia para integrar diversas plataformas web dinámicas en una sola aplicación utilizando un framework del lado del cliente, específicamente AngularJS. Se realiza un análisis detallado de diferentes tecnologías frontend y backend para seleccionar las herramientas con mejor rendimiento. El estudio incluye la implementación de un prototipo basado en microservicios con arquitectura RESTful para validar la eficiencia y comportamiento de la estrategia propuesta

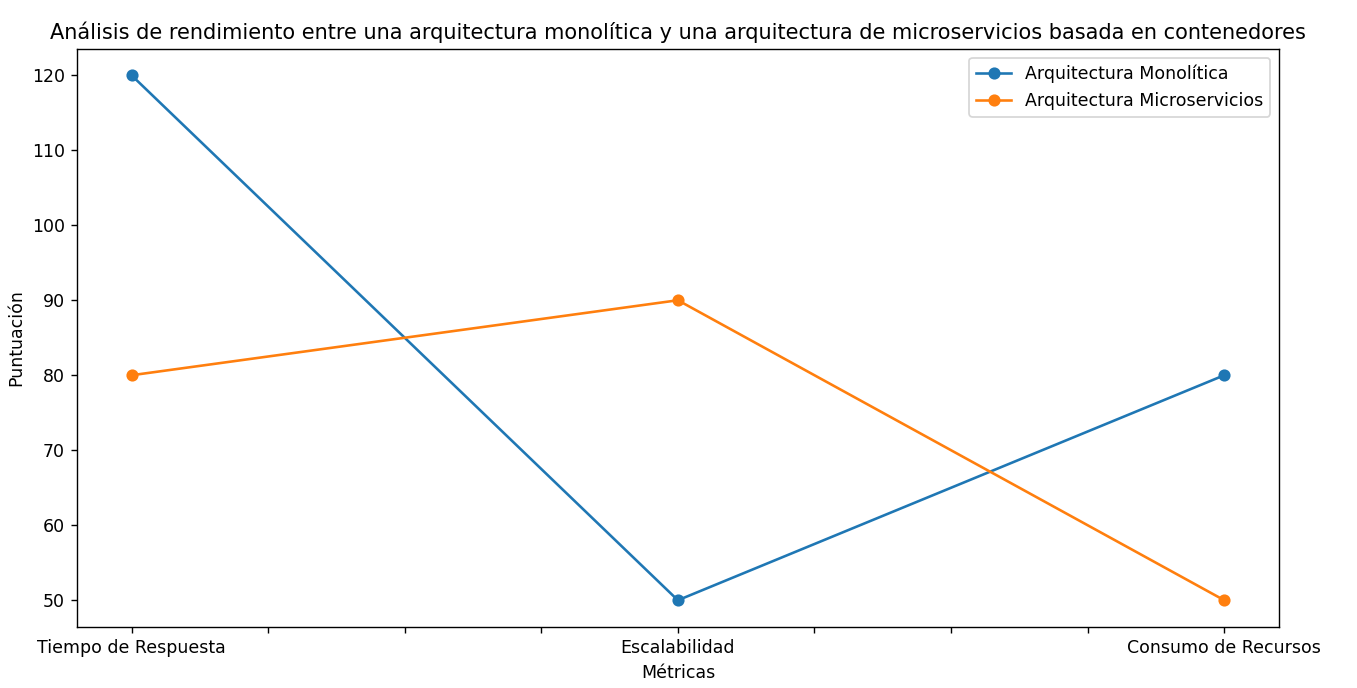
**Bibliografía: Mendoza Bautista, F. F. (2019). Estrategia de integración de diferentes plataformas web dinámicas mediante un framework del lado del cliente.**



**Reflexión**  
El uso de tecnologías modernas como Spring Boot para desarrollar microservicios resalta la eficiencia y facilidad que estas herramientas proporcionan. La implementación de un microservicio para gestionar listas de reproducción muestra cómo se pueden aplicar patrones de diseño efectivos para optimizar funcionalidad y rendimiento, ilustrando el potencial de microservicios en aplicaciones específicas.

**15- Análisis de rendimiento entre una arquitectura monolítica y una arquitectura de microservicios basada en contenedores:** Este estudio compara el rendimiento de arquitecturas monolíticas y de microservicios, específicamente en entornos de contenedores. A través de pruebas de rendimiento y análisis, se demuestra cómo la arquitectura de microservicios puede superar a la monolítica en términos de escalabilidad y gestión de recursos, aunque también se identifican desafíos asociados a la complejidad y administración de microservicios

**Bibliografía: Chiza, I. A. F. S., & Fernando, A. (2018). *Análisis de rendimiento entre una arquitectura monolítica y una arquitectura de microservicios–tecnología basada en contenedores* (Doctoral dissertation, Tesis Postgrado) Universidad Técnica del Norte. Ibarra. Ecuador).**

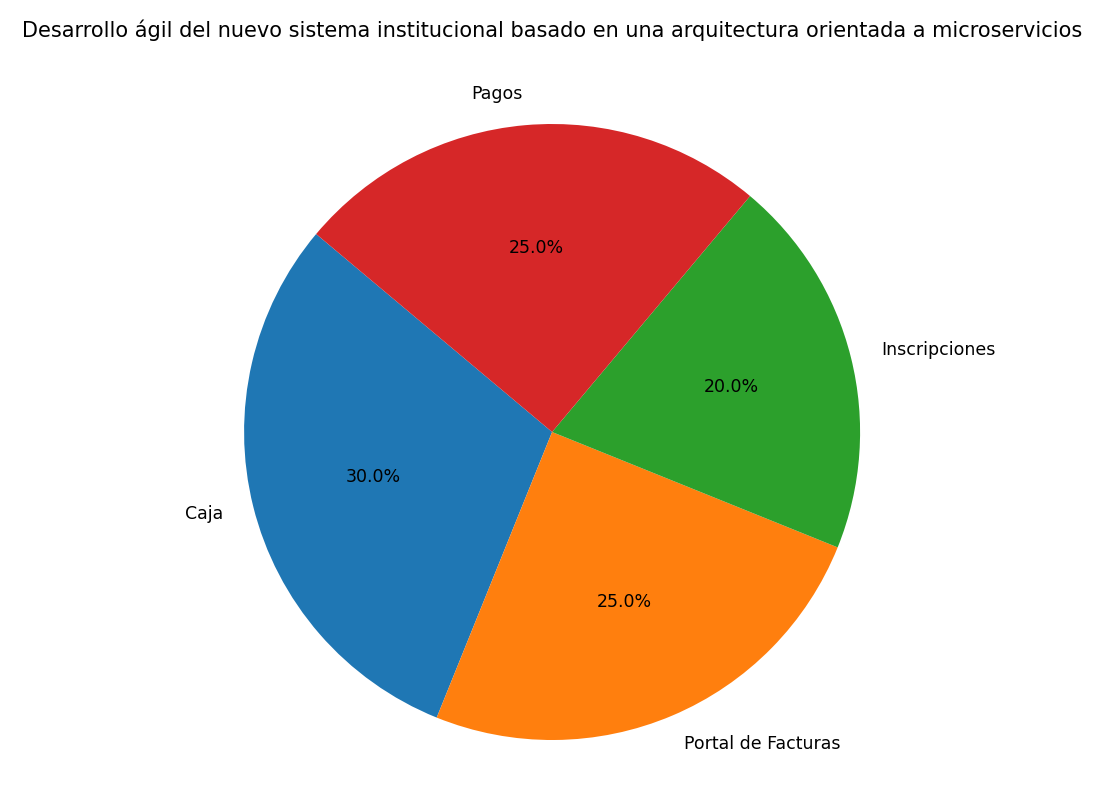


**Reflexión**

Este estudio propone una estrategia para integrar diferentes plataformas web dinámicas usando un framework del lado del cliente. La implementación de un prototipo basado en microservicios valida la eficiencia del enfoque. La capacidad de unir diferentes tecnologías en una sola aplicación refleja la flexibilidad que ofrecen los microservicios.

**16- Desarrollo ágil del nuevo sistema institucional basado en una arquitectura orientada a microservicios:** Este documento aborda el desarrollo de un sistema institucional en una universidad utilizando una arquitectura de microservicios, destacando la agilidad obtenida en comparación con una arquitectura monolítica. Se implementa la metodología de puntos de casos de uso para medir el rendimiento y se concluye que la nueva arquitectura facilita un desarrollo más rápido y flexible, validado a través de su implementación en módulos clave del sistema

­­­**Bibliografía: Rivadeneyra, A. H. (2019). *Desarrollo ágil del nuevo sistema institucional basado en una arquitectura orientada a microservicios* (Master's thesis, Universidad de Montemorelos (Mexico)).**

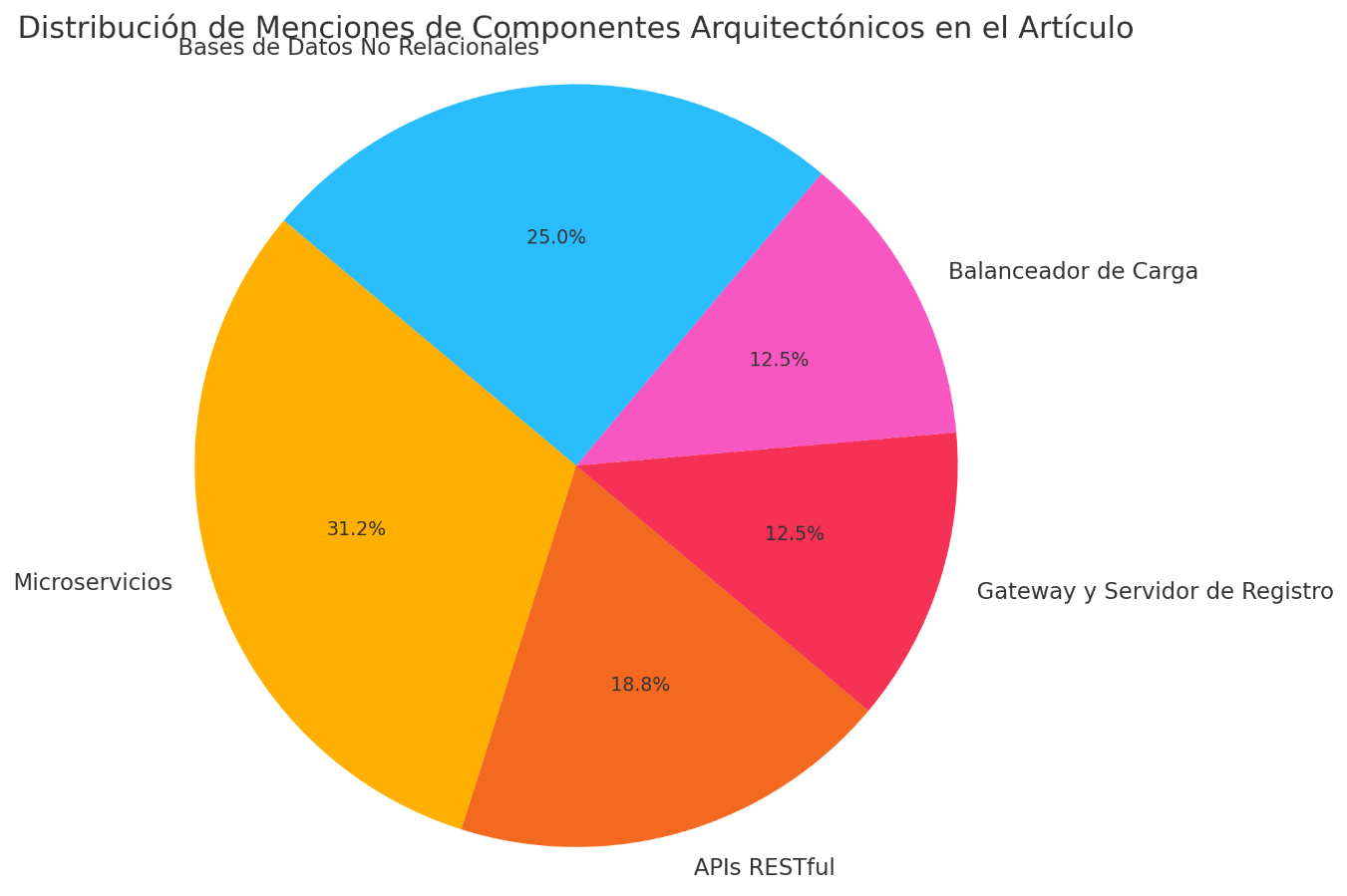


**Reflexión**

El análisis comparativo de rendimiento entre arquitecturas monolíticas y de microservicios en entornos de contenedores refuerza la idea de que los microservicios pueden gestionar recursos de manera más efectiva. Sin embargo, también se identifican desafíos de complejidad, lo que sugiere que las organizaciones deben estar preparadas para enfrentar estos retos al adoptar esta arquitectura.

17- **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:** Este documento propone una estrategia para implementar modelos de aprendizaje automático (ML) con arquitecturas de microservicios en sistemas de seguridad de ciudades inteligentes. Dado el crecimiento de la dependencia tecnológica, surgen desafíos como la protección ante ciberataques y la gestión de grandes volúmenes de datos. La solución implica el uso de microservicios para mejorar la escalabilidad y resiliencia de estos modelos. Se desarrollan dos microservicios: uno para el preprocesamiento de datos y otro para la predicción, utilizando herramientas como Kafka para la comunicación asíncrona y un balanceador de carga para optimizar el tráfico de solicitudes. Se validan estos enfoques con bases de datos de tráfico de red. brinda una visión clara sobre cómo implementar sistemas escalables y versátiles para manejar modelos de ML en ciudades inteligentes, con un enfoque en la seguridad y el manejo eficiente de datos en tiempo real.

**Bibliografía: Guerrero Martinez, 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.**

****

**Reflexión**

El enfoque en el desarrollo de sistemas institucionales con microservicios muestra cómo esta arquitectura permite una mayor agilidad. La implementación de metodologías como Scrum facilita un desarrollo más rápido y flexible, lo que es crucial para instituciones que buscan adaptarse rápidamente a las necesidades cambiantes.

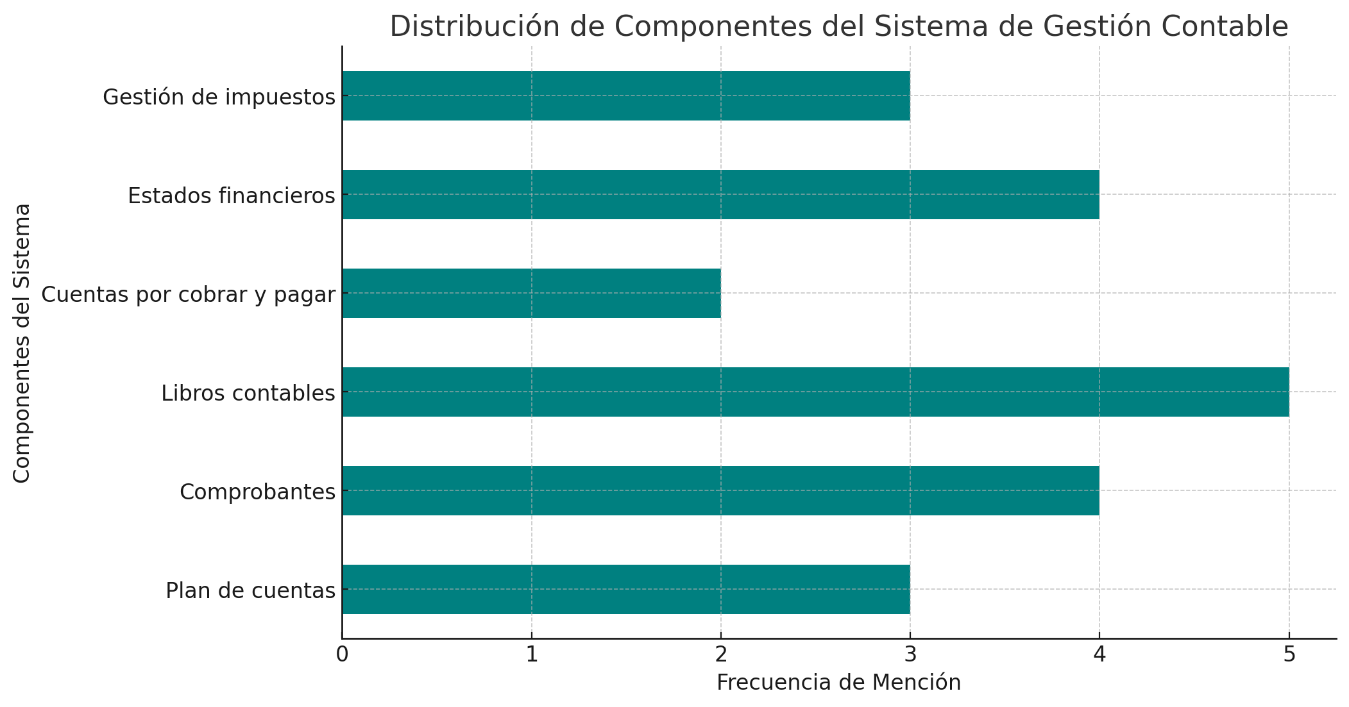
**18- Desarrollo de un módulo contable para fortalecer la gestión financiera utilizando una arquitectura de software basada en microservicios en la empresa Inveservice FGL S.A.S.** Este estudio presenta el desarrollo de un módulo contable basado en microservicios para optimizar la gestión financiera en Inveservice FGL S.A.S. La empresa enfrentaba ineficiencias debido a procesos manuales y un sistema contable monolítico. Se propone una solución automatizada, flexible y escalable que permita mejorar la precisión en los registros financieros y agilizar la generación de informes. Se utilizó la metodología Scrum para el desarrollo del software y se evaluó su impacto con el modelo de éxito de DeLone y McLean, obteniendo resultados positivos en cuanto a calidad del sistema, satisfacción del usuario y eficiencia. proporciona un enfoque práctico para automatizar y mejorar la eficiencia en la gestión financiera mediante microservicios, con una validación detallada sobre cómo esta arquitectura impacta positivamente en los procesos contables de la empresa.

**Bibliografía: Pinanjota Coyago, E. E. (2024). *Desarrollo de un módulo contable para fortalecer la gestión financiera utilizando una arquitectura de software basada en microservicios en la empresa Inveservice FGL SAS* (Bachelor's thesis).**

**Reflexión**El uso de microservicios para implementar modelos de aprendizaje automático en sistemas de seguridad de ciudades inteligentes destaca cómo esta arquitectura puede abordar desafíos modernos, como la gestión de grandes volúmenes de datos y la protección ante ciberataques. Esta capacidad para manejar datos en tiempo real es esencial en un entorno urbano cada vez más tecnológico.

**19 - Análisis de patrones de resiliencia en una arquitectura basada en microservicios** Este documento analiza patrones de resiliencia utilizados en arquitecturas de microservicios, enfocándose en cómo estos patrones ayudan a mantener el funcionamiento del sistema ante fallos. Se estudia un caso práctico en la empresa PedidosYa, que procesa 4 millones de órdenes semanales en América Latina, aplicando los patrones Timeout, Retry, Circuit Breaker y Bulkhead. Estos patrones mejoran la tolerancia a fallos al gestionar de forma eficiente errores de red, picos de tráfico y problemas de priorización. También se prueba la combinación de estos patrones para ofrecer una mayor robustez y evitar fallos en cascada

**Bibliografía: Suarez, S. L. (2023). *Análisis de patrones de resiliencia en una arquitectura basada en microservicios* (Doctoral dissertation, Universidad Nacional de La Plata).**

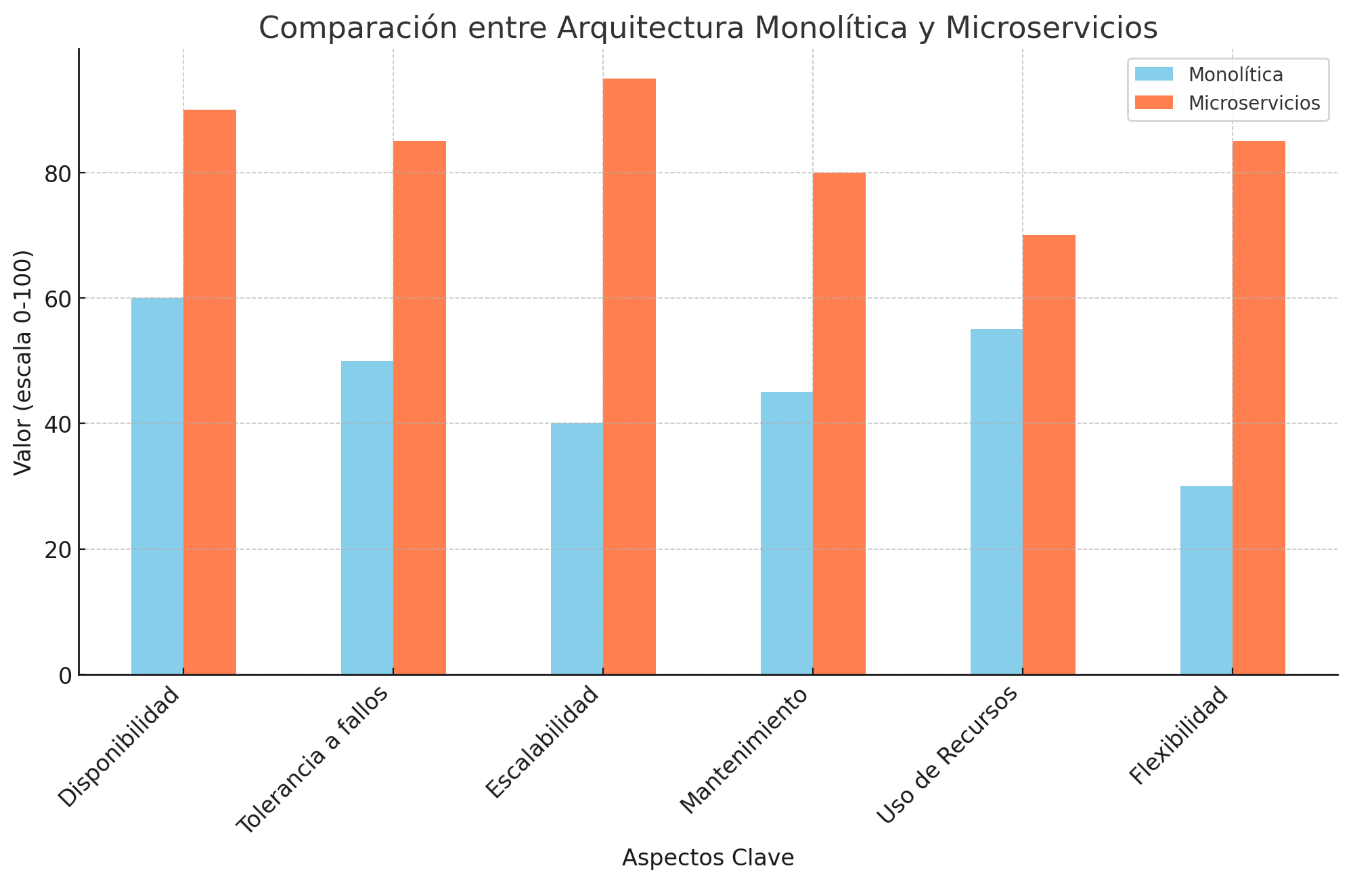
****

**Reflexión**El estudio de un módulo contable basado en microservicios muestra cómo la automatización puede optimizar procesos en la gestión financiera. La metodología Scrum utilizada para el desarrollo resalta la importancia de una estructura ágil que permite una rápida adaptación a las necesidades del negocio, mejorando la eficiencia y precisión en los registros financieros.

**Articulo 20-Desarrollo de software basado en microservicios un caso de estudio para evaluar sus ventajas e inconvenientes**

El documento analiza un caso de estudio sobre el desarrollo de software basado en microservicios, comparándolo con la arquitectura monolítica. Se realiza una implementación doble para una aplicación de comercio electrónico: una con una arquitectura monolítica y otra con microservicios. Se evaluaron aspectos como disponibilidad, tolerancia a fallos, y uso de recursos, utilizando tecnologías como Docker, Kubernetes y Azure. El objetivo es evaluar las ventajas e inconvenientes de cada enfoque, destacando cómo los microservicios aportan flexibilidad, escalabilidad, y facilitan el mantenimiento

**Bibliografía: Iranzo Jiménez, V. A. (2018). Desarrollo de software basado en microservicios: un caso de estudio para evaluar sus ventajas e inconvenientes (Doctoral dissertation, Universitat Politècnica de València).**

****

**Reflexión**

El enfoque de microservicios se presenta como una solución eficiente para superar las limitaciones de las arquitecturas monolíticas, especialmente en términos de escalabilidad y mantenimiento. Sin embargo, este tipo de arquitectura requiere una inversión inicial significativa y una infraestructura adecuada, como la orquestación con Kubernetes. Este caso de estudio muestra que aunque la transición a microservicios implica complejidad técnica, los beneficios obtenidos en términos de resiliencia y facilidad de mantenimiento justifican el esfuerzo, especialmente para sistemas en constante evolución.