



**ESCUELA POLITÉCNICA NACIONAL  
FACULTAD DE INGENIERÍA DE SISTEMAS  
INGENIERÍA DE SOFTWARE**



**Tarea  
Aplicaciones web avanzadas**

**Estudiante:**  
Jonathan Hugo Puglla

**Ensayo N°1 – IIB**  
Prof. Adrián Eguez, MSc.

**Tema:**  
Comparación SOA, microservicios y monolitos



## **Una comparación profunda de monolitos, SOA, y microservicios: Ventajas, desventajas**

En el mundo del desarrollo de software, las arquitecturas de aplicaciones han evolucionado para adaptarse a las cambiantes necesidades y desafíos tecnológicos. Entre las arquitecturas más destacadas se encuentran los monolitos, la arquitectura orientada a servicios (SOA) y los microservicios. Cada uno de estos enfoques tiene su propia filosofía y enfoque para construir aplicaciones sólidas y escalables. En este ensayo, se planea la interpretación de las ventajas, desventajas, similitudes y diferencias de estas arquitecturas, permitiendo una comprensión más clara de cuál es la más adecuada para diferentes contextos en el desarrollo de aplicaciones web.

Los monolitos, considerados por muchos como la "forma tradicional" de arquitectura de software, son aplicaciones en las que todos los componentes y funciones se encuentran en un solo código base. Esta simplicidad de desarrollo es su principal ventaja, permitiendo una rápida implementación y un proceso de depuración relativamente sencillo. Sin embargo, esta ventaja se ve contrarrestada por una escalabilidad limitada y un acoplamiento fuerte entre los componentes. Los monolitos son ideales para aplicaciones pequeñas o equipos de desarrollo reducidos, pero pueden volverse complejos y difíciles de gestionar a medida que crecen en tamaño y complejidad. Además, es necesario tener en consideración que no se asegura la funcionalidad mínima del sistema cuando existe una ruptura en su arquitectura.

La arquitectura orientada a servicios (SOA) se basa en el concepto de crear servicios independientes y reutilizables que pueden ser compartidos entre diferentes aplicaciones. Esto fomenta la reutilización de código y la interoperabilidad, lo que a su vez conduce a una mayor eficiencia en el desarrollo. Sin embargo, SOA puede presentar complejidad inicial en términos de configuración y gestión de servicios, y la comunicación entre servicios puede introducir latencia en las operaciones. A pesar de estas desventajas, SOA sigue



siendo una opción sólida para aplicaciones empresariales que requieren una integración fluida entre sistemas heterogéneos.

En contraste, los microservicios son una respuesta a la necesidad de agilidad y escalabilidad en el desarrollo de software. Los microservicios descomponen una aplicación en componentes independientes y especializados, lo que permite la escalabilidad independiente de cada servicio. Esta arquitectura fomenta un desarrollo ágil, ya que los equipos pueden trabajar de manera independiente en servicios específicos. Aunque los microservicios abordan muchas de las limitaciones de los monolitos y SOA, también presentan su propio conjunto de desafíos. La gestión de múltiples microservicios puede volverse compleja y requiere una infraestructura robusta para el despliegue y la administración.

Una vez analizados todos los puntos en contra de estas arquitecturas, es necesario considerar que la elección entre monolitos, SOA y microservicios no es una decisión unidimensional. Cada enfoque tiene su lugar y sus beneficios en el desarrollo de software, y la elección dependerá de las necesidades específicas de la aplicación y de la organización. Los monolitos ofrecen simplicidad en el desarrollo, pero pueden enfrentar limitaciones de escalabilidad y acoplamiento. SOA brinda reutilización e interoperabilidad, pero puede ser más complejo de implementar y gestionar. Los microservicios ofrecen agilidad y escalabilidad, pero también presentan desafíos en la gestión y la comunicación.

Ahora bien, la elección de la arquitectura dependerá de factores como el tamaño y la complejidad del proyecto, los recursos disponibles y la visión a largo plazo de la organización. Cada enfoque tiene sus ventajas y desventajas, y es crucial evaluar cuidadosamente cuál es el más adecuado para lograr los objetivos de desarrollo y brindar una experiencia excepcional a los usuarios finales. El mundo del desarrollo de software seguirá evolucionando – y cada tecnología que más se usa en 2023 se volverá obsoleta en 2028 y así cada 5 años – y la elección de la arquitectura adecuada jugará un papel fundamental en el éxito continuo de las aplicaciones modernas, teniendo en consideración el equipo de desarrollo y los recursos de la empresa. (Harris, N/A)



## Bibliografía

Harris, C. (N/A). *Atlassian*. Retrieved from Atlassian:  
<https://www.atlassian.com/es/microservices/microservices-architecture/microservices-vs-monolith>