



PERÍODO ACADÉMICO: 2021-B

ASIGNATURA: Aplicaciones Web Avanzadas

NOMBRE: Mahatma Israel Quijano Zanipattini

TÍTULO: Ensayo de SOA vs Microservicios

FECHA: 28 de febrero de 2022

En la construcción de sistemas resulta muy importante tomar en cuenta la arquitectura que se aplique de mejor manera a los requerimientos del negocio. Esta arquitectura puede responder a la necesidad de una organización y a la demanda de operación de datos, para lo que cual se dispone del enfoque de la arquitectura orientada a servicios y los microservicios.

La arquitectura orientada a servicios (SOA) corresponde a un enfoque de desarrollo de software que hace énfasis en servicios reutilizables donde cada uno contiene una función del negocio. Estos se gestionan y comunican a través de un bus de servicios empresariales (ESB). Por otro lado, los microservicios constituyen componentes especializados con un nivel alto de cohesión que generalmente se comunican mediante el uso de las API. Estos servicios son autónomos y generalmente están completamente separados unos de otros [1].

Si bien a simple vista puede parecer que uno de estos enfoques debe ser mejor que el otro, se debe realmente tomar en consideración las diferentes ventajas y desventajas que cada uno de estos representa para diferentes entornos de software que pueden encontrarse en una organización.

Entre las ventajas de la arquitectura orientada a servicios se tiene que esta puede ser la mejor elección ante aplicaciones complejas puesto que permite una fácil actualización de servicios. Igualmente, es independiente de la plataforma puesto que los servicios se comunican con otras aplicaciones a través de un lenguaje en común. La SOA ofrece también aplicaciones confiables en las que se pueden probar los servicios en comparación con una gran cantidad de código [2].

Sin embargo, la SOA cuenta con ciertas desventajas. Por un lado, todas las entradas deben validarse antes de enviarse al servicio. Además, el costo de esta arquitectura en términos de recursos humanos, tecnología y desarrollo es alto. Igualmente, las interacciones entre distintos servicios pueden aumentar el tiempo de respuesta y representar una mayor sobrecarga, tomando en cuenta que algunos servicios web pueden requerir un amplio envío y recepción de información [2].

Por otro lado, los microservicios tienen la ventaja de permitir al equipo de trabajo desarrollar, desplegar y escalar el servicio de manera independiente con respecto a los demás equipos de trabajo. Asimismo, facilita la obtención de un producto mínimo viable por lo que se lanzaría al mercado significativamente más rápido [2] [3].

No obstante, los microservicios cuentan con una mayor dificultad al momento de realizar pruebas. De igual manera, son costosos y requieren que los desarrolladores implementen el mecanismo de comunicación entre servicios [2].

Para comprender de mejor manera el uso de los microservicios, se puede tomar como ejemplo el siguiente escenario: la compañía SCC de cadenas de suministros ofrece servicios de entrega puerta a puerta, sin embargo su sistema de procesamiento de órdenes era lento y anticuado. Por esta razón, la compañía requería implementar un sistema de procesamiento de órdenes que automatice las tareas y el flujo de trabajo, que habilite el acceso a tecnologías escalables de vanguardia que mejoren el proceso y que permita analizar los requerimientos de negocio para evaluar el impacto y aplicar la mejora continua. Con esto en mente, la compañía decidió aplicar el uso de microservicios para lo cual eligió Azure Microservices al igual que BizTalk Server como IaaS. Posteriormente a esta implementación se tuvieron diferentes beneficios. En lo que respecta al despliegue del servicio de manera ágil, se obtuvo una gran facilidad para gestionar las entregas de nuevas características, actualizaciones y correcciones de errores en componentes individuales sin volver a desplegar la aplicación entera. Asimismo, la compañía contó con la habilidad para escalar los servicios individuales y subsistemas basados en la necesidad de recursos sin escalar toda la aplicación. Por último, se aumentó la agilidad de los equipos de trabajo dado que cada uno podía disponer de la aproximación de despliegue, lenguaje, plataforma de microservicios y modelo de programación que prefiera para cada servicio [4].

Como se pudo observar en el caso de estudio, la aplicación de una arquitectura de servicios puede ofrecer diferentes ventajas. Sin embargo, no hay que olvidar que al momento de elegir entre SOA o microservicios (o incluso una aplicación monolítica) se debe tomar en cuenta las necesidades del negocio.

Bibliografía:

- [1] "SOA vs. Microservices: What's the Difference?", IBM.com, 2021. [Online]. Available: <https://www.ibm.com/cloud/blog/soa-vs-microservices>. [Accessed: 26- Feb- 2022].
- [2] A. Walker, "SOA vs Microservices: Key Differences explained with Examples", Guru99, 2022. [Online]. Available: <https://www.guru99.com/microservices-vs-soa.html>. [Accessed: 26- Feb- 2022].
- [3] "SOA vs. Microservices: A Complete Comparison", Newizze LLC - Custom Web Development Company, 2019. [Online]. Available: <https://newizze.com/soa-vs-microservices-comparison-for-2019/>. [Accessed: 26- Feb- 2022].
- [4] "Azure Microservices - A Sample Case Study | TwoConnect", TwoConnect, 2021. [Online]. Available: <https://www.twoconnect.com/azure-microservices-sample-case-study/>. [Accessed: 27- Feb- 2022].