



PERÍODO ACADÉMICO: 2021_B

ASIGNATURA: Aplicaciones Web Avanzadas

CURSO: GR1ISW

NOMBRE: Diana Stefanía López Prihodko

FECHA: 28/02/2022

Ensayo SOA vs Microservicios

En los últimos años se escucha mucho el concepto de servicios en la nube, servicios web, arquitectura de servicios, entre otros. Sin embargo, no se investiga mucho realmente el significado o la utilidad que tienen los servicios en los diferentes sistemas que se utilizan todos los días y queda como un concepto usado en la publicidad para atraer a más usuarios. Por ello, en el presente texto se profundizará sobre esta temática en dos tipos de diseños arquitectónicos usados y se analizará sobre un caso de estudio planteado.

Los servicios se los relaciona con una unidad o un conjunto de funciones que debe hacer un sistema para ejecutar una tarea específica, en donde una persona analizó el impacto y la fragmentación del mismo de manera adecuada [1]. Por otro lado, esta operación debe entregar información en base a las entradas que se dio o bien realizar un cambio interno a la infraestructura que maneja.

En la industria existen una gran cantidad de arquitecturas orientadas a los servicios, las cuales tienen estándares bien definidos para la comunicación de los diferentes servidores que alojan los servicios utilizados, así bien se verá dos de ellos: la arquitectura orientada a los servicios (SOA) y la arquitectura de microservicios.

Ambos enfoques manejan una estructura bastante similar, pero con diferencias en varios aspectos importantes, como por ejemplo el proceso de la integración de estos servicios, en donde SOA tiene un bus de servicio y los microservicios pueden ser integrados usando un servicio manejado por una API, otra clase de tecnologías o simplemente este es independiente [2]. Lo primero que se analizarán son las ventajas que tienen cada una de estas arquitecturas.

SOA se puede mantener y escalar de manera sencilla, ya que todos los servicios manejan un mismo lenguaje y/o estructura haciéndolo así fácil de modificar la parte que es necesaria [1]. Por otro lado, al tener una estructura estable es fácil encontrar en el mercado servicios que te ayuden a generar esta infraestructura, volviéndola eficiente y versátil para las empresas de desarrollo.

Adicionalmente, el SOA es bastante versátil para sistemas que van a ser usados en territorios pequeños, ya que se puede permitir parar los servidores cada vez que exista un cambio o modificación en el código que se desarrolló [1]. No obstante, si es el caso contrario los microservicios permiten asegurar que no suceda un paro abrupto de todo el sistema y tener a los usuarios complacidos en la gran mayoría hasta que uno de los microservicios se ha vuelto a levantar.

Ahora si se ve a los microservicios, estos tienen una comunicación más simple de mensajería entre servidores, volviéndola más transparente que SOA y asimismo, los microservicios permiten seguir la ideología de “Divides y vencerás” de una manera más explícita [3].

Una gran desventaja que tienen los microservicios con respecto al SOA es que no existe un único camino o un conjunto de buenas prácticas para implementarlos, por ello SOA tiene una gran ventaja. Además, los microservicios no tienen coordinación, haciéndolos independientes (en su mayoría de casos) y esto produce que SOA sea utilizado para sistemas que debe manejar sus servicios de manera interdependiente o con un gran nivel de coordinación [2].

El caso de estudio que se va a analizar es de Rainyday Grocer, el cual es una tienda de productos alimenticios en línea de entrega a domicilio. Su política es llevar los productos puerta en puerta y de manera personalizada, ya que suponen que todos los días el clima es lluvioso y las personas no salen de la comodidad de su hogar o se trasladan a puntos fijos [4].

Su servicio se maneja por 4 microservicios por el momento, los cuales son: inicio de sesión, recibir orden, proporcionar estimación y confirmar orden. Estos fueron definidos de esta manera, debido al flujo de procesos de la empresa, los cuales fueron diagramados por medio de un modelo BPMN.

Se puede observar que al tener un equipo pequeño de desarrolladores y un entorno de negocio bastante definido, pudieron trabajar en equipo rápidamente en cada uno de los microservicios necesarios de manera ágil, usando la ideología DevOps. Por otro lado, al utilizar esta arquitectura lograron sacar provecho de la agilidad y flexibilidad que esta les entrega. No obstante, si el negocio crece de manera exponencial y se debe añadir una gran cantidad de procesos, el equipo actual de TI no lograría abastecer la necesidad.

Como se pudo ver, se puede optar la una o la otra opción para implementar los servicios en un sistema dado. Esto dependerá únicamente de la necesidad de la empresa y el análisis que realice un arquitecto al momento de ver el proceso o problema que debe solucionar. Por otro lado, cada empresa aplica algo diferente en base a prueba y error. Siendo así que se puede entender que las prácticas y las personas son los factores que alteran más el desarrollo de un sistema.

Bibliografía:

- [1] "¿Qué es la arquitectura orientada a los servicios (SOA)?", Red Hat, 2020. [Online]. Disponible: [https://www.redhat.com/es/topics/cloud-native-apps/what-is-service-oriented-architecture#:~:text=La%20arquitectura%20orientada%20a%20los%20servicios%20\(SOA\)%20es%20un%20tipo,red%20con%20un%20lenguaje%20com%C3%BAn](https://www.redhat.com/es/topics/cloud-native-apps/what-is-service-oriented-architecture#:~:text=La%20arquitectura%20orientada%20a%20los%20servicios%20(SOA)%20es%20un%20tipo,red%20con%20un%20lenguaje%20com%C3%BAn) .
- [2] "¿Qué es SOA y cuál es su diferencia con los Microservicios?", Blog.powerdata.es, 2022. [Online]. Disponible: <https://blog.powerdata.es/el-valor-de-la-gestion-de-datos/que-es-soa-y-cual-es-su-diferencia-con-los-microservicios>.
- [3] "Arquitectura web: SOA vs. Microservices, aplicaciones y diferencias", bilib, 2022. [Online]. Disponible: <https://www.bilib.es/actualidad/blog/noticia/articulo/arquitectura-web-soa-vs-microservices-aplicaciones-y-diferencias/>.
- [4] "Microservices Architecture – CASE STUDY: Rainyday Grocer", Opengroup.org, 2022. [Online]. Disponible: <https://www.opengroup.org/soa/source-book/msawp/p5.htm>.