ESCUELA POLITÉCNICA NACIONAL

Facultad de Ingeniería de Sistemas

Ingeniería de Software

APLICACIONES WEB AVANZADAS

Flores Bryan

Periodo: 2021 B

Fecha: 28-02-2022

**SOA VS MICROSERVICIOS**

**COURSERA**

Contenido

[Introducción 2](#_Toc96028665)

[Desarrollo 2](#_Toc96028666)

[SOA 2](#_Toc96028667)

[Microservicios 2](#_Toc96028668)

[SOA vs Microservicios 2](#_Toc96028669)

[Caso de estudio Coursera 3](#_Toc96028670)

[Conclusión 3](#_Toc96028671)

[Bibliografía 3](#_Toc96028672)

# Introducción

En el presente ensayo se discutirá del tema de Microservicios, SOA y el caso de estudio de Coursera.

Se definirá lo que es SOA, Microservicios respectivamente, además de hacer una comparación entre ambas opciones, y se presentará un caso de estudio de Microservicios de Coursera.

El Documento se encuentra estructurada en 3 partes principales que son la Introducción, el Desarrollo y Conclusiones

# Desarrollo

## SOA

De sus siglas del idioma inglés (Service Oriented Architecture), es un estilo de diseño de software en donde los componentes de este software brindan servicios a los otros componentes a través de un protocolo de comunicación a través de una red. En SOA una serie de servicios se comunican entre sí, en una de dos formas: a través del paso de datos o a través de dos o más servicios que coordinan una actividad[1]. Abarcan principios claves tales como:

* Valor para el negocio
* Metas estratégicas
* Interoperabilidad intrínseca
* Servicios compartidos
* Flexibilidad

Imagen que contiene Diagrama

Descripción generada automáticamente

Figura 1 Aspectos básicos de SOA[1]

## Microservicios

Al igual que SOA es un estilo de diseño de software, que estructura una aplicación como una colección de servicios que son:

* Autónomos: Cada servicio es una arquitectura que se puede desarrollar, implementar y operar sin afectar el funcionamiento de otros.[2]
* Especializados: Cada servicio se enfoca en resolver un único problema específico.[2]

Las características principales de los microservicios son[3]:

* Múltiples Componentes
* Creado para los negocios
* Enrutamiento sencillo
* Descentralizado
* Resistente a fallas
* Evolutivo

## SOA vs Microservicios

Empezaremos diciendo que una diferencia importante es el alcance, debido a que SOA tiene un alcance empresarial, mientras que la arquitectura de microservicios tiene un alcance de aplicación.

En la Tabla [1] se pueden observar las distintas diferencias entre SOA y Microservicios:

|  |  |  |
| --- | --- | --- |
|  | SOA | Microservicios |
| Arquitectura | Diseño para compartición de recursos entre servicios | Servicios que pueden funcionar independientemente |
| Granularidad | Servicios grandes y por módulos | Servicios detallados |
| Gobernanza | Protocolos comunes en todos los equipos | Requiere colaboración entre equipos |
| Alcance | Mejora una integración a gran escala | Mejor para aplicaciones mas pequeñas generalmente basadas en web |
| Comunicación | Se comunica a través de un Bus de servicio empresarial | Se comunica a través de una capa API |
| Acoplamiento y cohesión | Se basa en la compartición de recursos | Se basa en un contexto acotado para el acoplamiento |
| Servicios remotos | Protocolos SOAP Y AMQP | Utiliza REST y JMS |

Tabla 1 Diferencias entre SOA y Microservicios

## Caso de estudio Coursera[4]

Coursera es una empresa tecnológica educativa, la compañía se encuentra asociada con universidades y organizaciones de todo el mundo para ofrecer cursos en línea.

Coursera cuenta con mas de 130 millones de usuarios.

### El reto de Coursera:

* Coursera tenia una gran aplicación monolítica para procesar trabajos por lotes, y esto llevaba a grandes complicaciones para ejecutar, implementar y escalar.
* Para esto se creaba un nuevo subproceso cada vez que se necesitaba completar un nuevo trabajo, lo que obviamente llevaba un consumo de memoria y CPU produciendo ineficiencias.
* El tener centralización hicieron que este desgaste de recursos botó la aplicación.
* Intentaron solucionarlo con microservicios, pero Apache Mesos fue un problema antes que una solución.

### ¿Por qué Coursera eligió Amazon Web services?

* Los contenedores Docker permitieron a Coursera pasar fácilmente a una arquitectura basada en microservicios
* Cada trabajo es un Docker, que Amazon ECS programa.
* Amazon ECS brindan aislamiento de recursos necesarios.

### Los Beneficios que recibió Coursera

* El equipo de ingeniería de Coursera tuvo un prototipo en funcionamiento en menos de 2 meses debido a que la configuración de Amazon ECS es sencilla y administra todos los detalles del cluster.
* Ahora cada cambio que se realizaba en horas, se lo realiza en minutos, además que cada equipo puede desarrollar y actualizar sus respectivas aplicaciones independientemente.
* Los grupos de escalamiento automático permite que la capacidad de cómputo se amplie de acuerdo con un trabajo dinámico
* Se transfiere el tiempo gastado en la ingeniería de infraestructura debido a que Amazon ECS se encarga de todo.

# Conclusión

* Dependiendo del punto de vista que se evalúa SOA y Microservicios cada uno tendrá ventajas sobre el otro, por lo que se aconseja al lector, se evalúa el contexto en donde se implementará la arquitectura para tomar una decisión adecuada.
* Es una realidad que los microservicios estén en auge debido a su flexibilidad y bajo acoplamiento.
* Muchas empresas disfrutan de los microservicios, entre ellas Facebook, Netflix, Amazon.

# Bibliografía

[1] “What Is Service-Oriented Architecture? | by Software Development Community | Medium.” https://medium.com/@SoftwareDevelopmentCommunity/what-is-service-oriented-architecture-fa894d11a7ec (accessed Feb. 17, 2022).

[2] “¿Qué son los microservicios? | AWS.” https://aws.amazon.com/es/microservices/ (accessed Feb. 17, 2022).

[3] “What are Microservices? | API Basics | SmartBear.” https://smartbear.com/solutions/microservices/ (accessed Feb. 17, 2022).

[4] “Coursera Case Study.” https://aws.amazon.com/es/solutions/case-studies/coursera-ecs/ (accessed Feb. 17, 2022).