

“UNIDAD III. Microservicios y contenedores”

Materia: Administración Avanzada de Infraestructura Virtualizada.

Alumno: Garduño Velazquez Carlos Uriel

Docente: Maribel Morales Guzmán

Fecha: 13/06/2021

Cuatrimestre: Mayo-agosto 2021

Grupo: 7IRI1

Indicé

[Microservicio 1](#_Toc74498557)

[Componentes en una arquitectura de microservicios 1](#_Toc74498558)

[Identificar las ventajas y desventajas de la arquitectura de microservicios 2](#_Toc74498559)

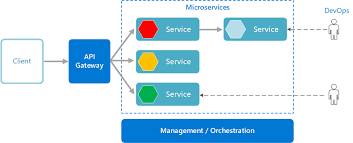
[Ventajas 2](#_Toc74498560)

[Identificar la diferencia entre contenedor y microservicio 3](#_Toc74498561)

[Contenedores 3](#_Toc74498562)

[Referencias 4](#_Toc74498563)

# Microservicio

Hace referencia a un estilo de arquitectura de sistemas, donde se desarrollan sistemas que sean flexibles y escalables y que conlleven un mantenimiento más sencillo. En otras palabras, es que una arquitectura se dedique a “hacer una cosa y hacerla bien”, donde los servicios se enfoquen en realizar una sola función.

# Componentes en una arquitectura de microservicios

La arquitectura de los microservicios, esta compuestas de por un conjunto de pequeños servicios, cada uno tiene un propio propósito, tienen un conjunto de reglas de negocio especifico, implementadas de forma independiente. Los servicios son los siguientes:

* **Mantenimiento y Evolución de los Servicios más Estables:** Los desarrolladores tratarán de códigos que ejecuta una sola función, y cada servicio individual no crece indefinidamente con el crecimiento del sistema.
* **Servicios con bajo nivel de Acoplamiento e Interdependencia:** De esta forma, el mantenimiento en un servicio no interfiere directamente en otras funcionalidades del sistema.
* **Escalabilidad del Sistema:** Se obtiene con el deploy y la replicación de micro servicios a través de la infraestructura de servidores, máquinas virtuales y contenedores de forma independiente. Esto hace que el crecimiento y la posibilidad de adaptación del sistema mucho más flexible.
* **Reducción de Costos:** Como cada aplicación sólo utiliza los servicios que necesita, los costos se asocian directamente a la funcionalidad ya la carga de uso del sistema, no siendo necesario cargar costos asociados a funcionalidades no utilizadas.
* **Flexibilidad de la Tecnología:** No es necesario atar a los desarrolladores a una tecnología específica, ya que hay un bajo acoplamiento entre los servicios. De esta forma, se puede utilizar la mejor tecnología para atender a cada caso, además de la posibilidad de evolucionar el sistema continuamente, disminuyendo el riesgo de la obsolescencia tecnológica.
* **Facilidad para Introducir Cambios en la Producción:** Los cambios en el sistema se realizan a través de los cambios y las evoluciones realizadas en los servicios. Por lo tanto, no hay un sistema que debe reiniciarse para continuar funcionando. El equipo de desarrollo que necesitará acompañar el cambio será el equipo responsable de los servicios que están siendo alterados.

# Identificar las ventajas y desventajas de la arquitectura de microservicios

Como cualquier sistema tiene sus ventajas y sus desventajas, todo depende del proyecto que se va a generar como también las necesidades de la empresa u organización, a continuación, se mencionaron algunas ventajas y desventajas:

## Ventajas

* Versátil los microservicios permiten el uso de diferentes tecnologías y lenguajes.
* Fácil de integrar y escalar con aplicaciones de terceros.
* Los microservicios pueden desplegarse según sea necesario, por lo que funcionan bien dentro de metodologías ágiles.
* Las soluciones desarrolladas con arquitectura de microservicio permiten la mejora rápida y continua de cada funcionalidad.
* El mantenimiento es más simple y barato: Con los microservicios se puede hacer mejoras de un módulo a la vez, dejando que el resto funcione normalmente.
* El desarrollador puede aprovechar las funcionalidades que ya han sido desarrolladas por terceros: no necesita reinventar la rueda, simplemente utilizar lo que ya existe y funciona.
* Un proyecto modular basado en microservicios evoluciona de forma más natural, es una forma fácil de gestionar diferentes desarrollos, utilizando los recursos disponibles, al mismo tiempo.

Desventajas

* Debido a que los componentes están distribuidos, las pruebas globales son más complicadas.
* Es necesario controlar el número de microservicios que se gestionan, ya que cuantos más microservicios existan en una solución, más difícil será gestionarlos e integrarlos.
* Los microservicios requieren desarrolladores experimentados con un nivel muy alto de experiencia.
* Se requiere un control exhaustivo de la versión.
* La arquitectura de microservicios puede ser costosa de implementar debido a los costos de licenciamiento de aplicaciones de terceros.

# Identificar la diferencia entre contenedor y microservicio

Para conocer la diferencia entre contenedor y microservicio, hay que definir que es contenedor ya que se definió microservicio anteriormente.

## Contenedores

Estos en lugar de transportar un sistema operativo completo, así como su software, simplemente encapsula su código y sus recursos a un contenedor, que se puede ejecutar en cualquier entorno, con esto se trajo la posibilidad de agrupar productos en masa, de forma organizada y mucho más práctica.



La diferencia radica en que el microservicio es un conjunto de servicios, que están conectados entre si para dar un funcionamiento, y se depende a que no falle ningún servicio, mientras en contenedor puede ser todos los servicios en uno solo y se tiene mas control de su funcionamiento ya que esta unificado todos los servicios.

# Referencias

Computing, R. C. (2019, 8 mayo). *Microservicios y Contenedores*. Revista Cloud Computing. <https://www.revistacloudcomputing.com/2019/05/microservicios-y-contenedores/>

Rebels, G. (2018, agosto 16). *Microservicios: ventajas y contras de la arquitectura descentralizada*. Medium. <https://medium.com/@goodrebels/microservicios-ventajas-y-contras-de-la-arquitectura-descentralizada-a3b7fc814422>

Cantó, J. C. (2019, 6 febrero). *Arquitectura de software: microservicios, contenedores y orquestación*. Bilib. <https://www.bilib.es/actualidad/blog/noticia/articulo/arquitectura-de-software-microservicios-contenedores-y-orquestacion/>