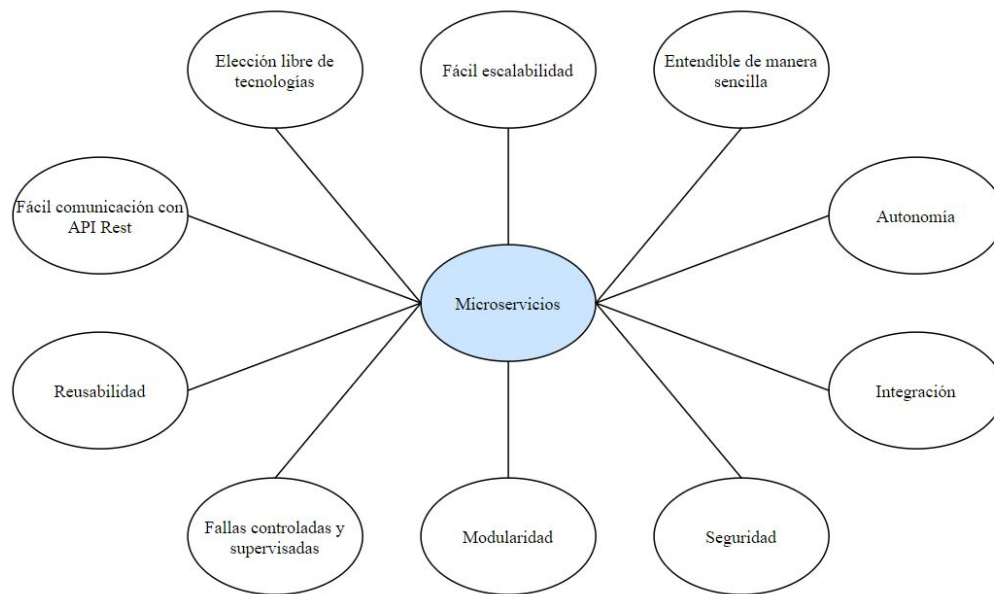


Nombre: Hernán Javier Aguilar Cruz

CC: 1006503110

Parcial Final - Arquitectura de Software (Teoría)

1. Realice un mapa mental muy bien elaborado donde se muestre claramente las ventajas de los microservicios, frente a las arquitecturas monolíticas



2. Explique con sus palabras los Middleware para Servicio Web y de un ejemplo.

El middleware para servicio web cumple la funcionalidad de ofrecer un conjunto de servicios que incluyen funciones. Esto quiere decir que son los encargados de desarrollar distintas actividades como lo son tareas, permitiendo así el desarrollo de webs de manera más eficaz. Entre sus tareas, cabe resaltar: optimización de páginas, servicios de mensajería, almacenamiento en memoria, automatización.

Ejemplo: Tenemos una página web desplegada, y un usuario requiere diligenciar un formulario, al momento de ser llenado este se guarda en la base de datos de la página. El middleware hace presencia ya que se encarga de comunicar la información diligenciada en el formulario desde un navegador con la base de datos para que sean almacenados.

3. Defina con sus palabras qué es un Microservicio.

Para entender qué es un microservicio vale la pena verlo como una parte de una aplicación en la que se está trabajando. Esta parte es la que tiene una sola tarea (de ahí su nombre micro de pequeño y servicio ya que presta como se dijo antes, funcionalidades en el trabajo). Estos microservicios entonces, son independientes de la aplicación, deben ser concretos y sencillos y aparte deben resolver un problema específico. Como se trata de microservicios independientes, se cuenta con libertad de elegir tecnologías. Para realizar la comunicación con otros se hace mediante APIS.

4. Para que un desarrollo sea clasificado como microservicio qué características se deben de cumplir

Se plantean las siguientes características para determinar si es un microservicio:

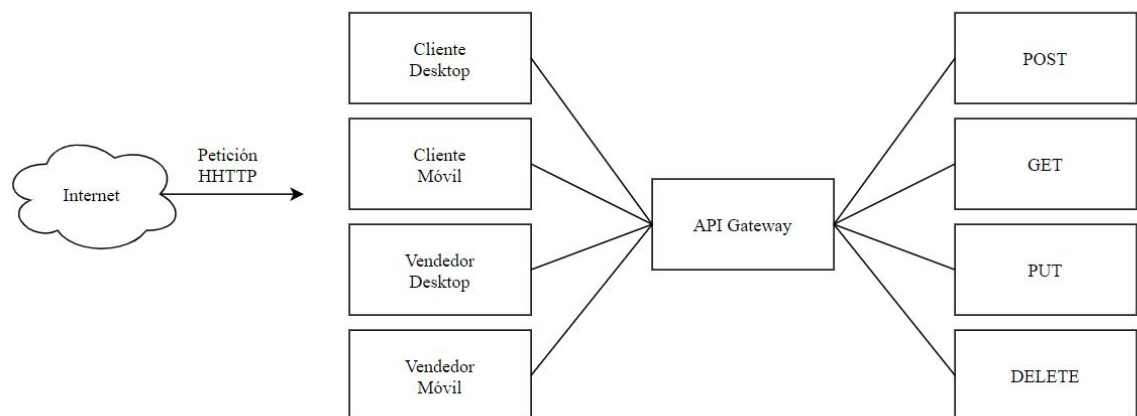
- Encargado de resolver problemas específicos, por lo que éste cuenta con una funcionalidad clara y precisa para encontrar una solución a un único problema.
- Son independientes, por lo que al escalarse no deben afectar a otros.
- La comunicación con externos debe hacerse mediante APIS.

5.Cuál es el objetivo del Api Gateway en los microservicios. De un ejemplo práctico.

El objetivo del Api Gateway en los microservicios es desarrollar aplicaciones de manera más eficiente, por lo que se presta como intermediario entre el cliente y los servicios a prestar. Si se piensa de esta forma, también se plantea el uso de realizar peticiones de manera simultánea a los servicios. Haciendo así, un puente para permitir el acceso a funcionalidades o servicios de manera más sencilla.

Ejemplo: Se cuenta con una página web de compra y venta de carros, en esta se opta por utilizar Rest para solucionar las solicitudes en línea. Como se plantea esto, entonces lo siguiente es darnos cuenta que estaremos utilizando 4 servicios diferentes, tales como POST (cuya funcionalidad consta de de agregar nuevos recursos, en este caso carros), GET (que sirve para retornar u obtener un recurso en concreto, para la página esto sería un carro), PUT (modificar el recurso) y DELETE (por si se requiere

eliminar un carro). Cuando se desarrolla con estas funcionalidades, la página debe asignar a cada desarrollador un servicio, por lo que estos tendrán 4 servicios cada uno con diferentes tecnologías, para que al final se una una API Gateway que pueda ayudar a seleccionar el servicio dependiendo del movimiento en la página (ya sea administrador que desea modificar la información de venta de su carro, o cliente buscando para comprar uno).



6. Explique con sus palabras el patrón Circuit Breaker que se aplica a los microservicios.

El Circuit Breaker se trata de un patrón de diseño cuya función es ayudar a saber si un microservicio se puede utilizar o no (si está disponible). Así mismo, el Circuit Breaker ayuda a saber si existe una falla, la detecta y bloquea el acceso al microservicio (espera hasta que el microservicio se esté prestando de nuevo). Por lo que, al momento de hacer una petición, este patrón de diseño es el encargado de decir si el microservicio que se requiere está funcionando. Se utiliza cuando se desea evitar fallos, y así el sistema es más tolerante a los mismos, y hace que no existan fallos en cadena. Se bloquea en caso de que un servicio no esté disponible en un preciso momento.

7. En el modelo de referencia en la arquitectura de los microservicios, explique para qué se utiliza Discovery Service.

Discovery Service se utiliza para dirigir y así mismo mostrar al cliente los microservicios que este requiere teniendo en cuenta los que estén disponibles y lo que el mismo usuario pueda necesitar. Como lo dice su nombre, este servicio tiene la capacidad de encontrar y clasificar servicios para poder utilizarlos próximamente. Estos servicios se guardan en el tiempo de arranque.

8. Explique las capas verticales en los microservicios.

- Capa 1: Servicios core (Servicios de negocio): Conocen la lógica del negocio y son los que aplican las reglas del mismo. Estos son los encargados entonces de la persistencia de los datos.
- Capa 2: Servicios compuestos: Servicios a los que son redireccionados.
- Capa 3: Servicios API: Como sabemos de antes, sirven para exponer y compartir los servicios a otros (terceros).

9. En el modelo de referencia en la arquitectura de los microservicios, explique para qué se utiliza Edge Service.

Edge Service se plantea como una API Gateway, entonces tiene como función dirigir los microservicios dependiendo de la persona que lo requiera. Esto entonces, nos permite filtrar peticiones de manera sencilla.

10. Cómo se clasifican los Middleware y cuál es su uso en la industria del software.

La clasificación del Middleware está planteada en 2 subgrupos:

- Middleware de alto nivel: Hacen énfasis en los servicios relacionados con la web, servidores de aplicaciones y así mismo transacciones. Lo que vendría siendo la infraestructura de una aplicación.

Uso: Publicar o desplegar una aplicación.

- Middleware de bajo nivel: Estos, a diferencia de los middleware de alto nivel, se enfocan en los servicios como lo son las comunicaciones entre una aplicación y el acceso a la base de datos; como puede ser los servicios que hace uso el cliente.

Uso: Integrar diversos servicios hacia el cliente, comunicación en la web por medio de protocolo http.