

# Clase integradora: microservicios

# Una arquitectura de **microservicios** está compuesta por:

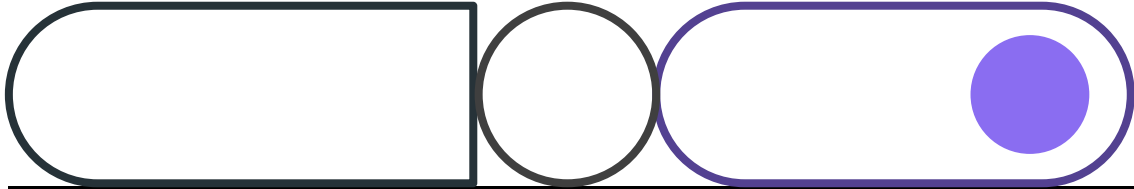
- **Interfaces de usuario (user interface):**

Son los componentes de presentación. Estos componentes son los responsables del control de la interfaz de usuario y el consumo de servicios remotos.

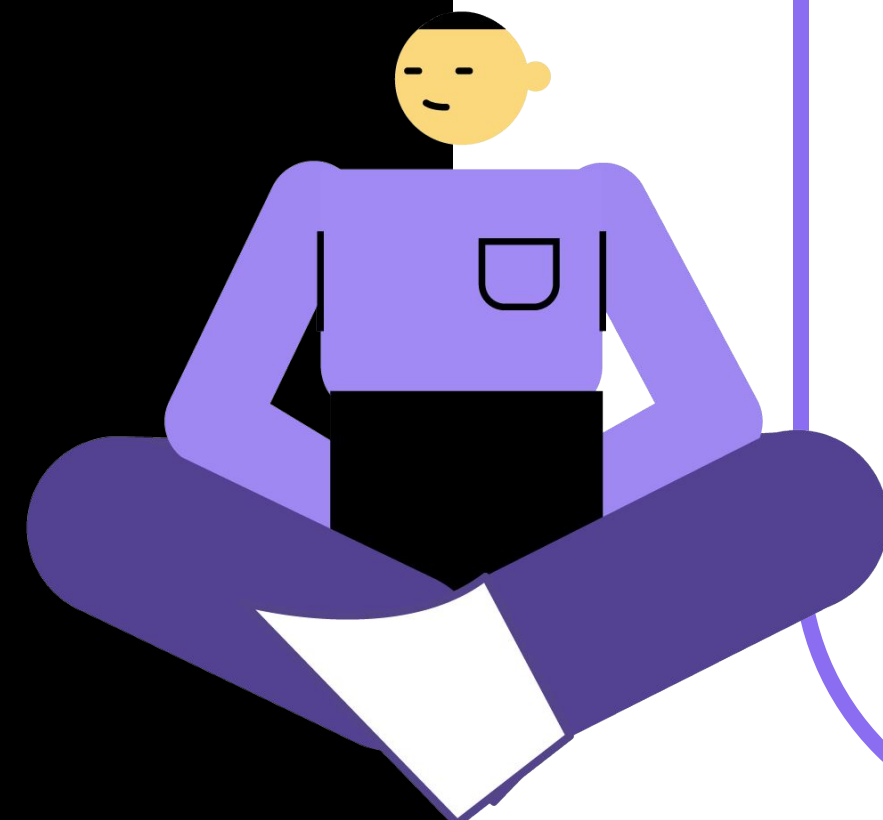
- **API Gateway:**

Es el sistema que se encarga de exponer los servicios a consumir mediante la implementación de API. En pocas palabras, la API Gateway es la cara que damos a los clientes, y es la forma en que los clientes externos se comunicarán con nosotros, por lo que es común que la API Gateway ofrezca servicios simples y de alto nivel que oculten la complejidad de nuestra arquitectura.

- **Sistema de seguridad (security):**  
Implementa la capa de seguridad (recomendable en la capa de servicios API).
- **Balanceador de carga (load balancer):**  
Permite el balanceo entre distintas instancias de forma transparente a la hora de consumir un servicio.
- **Servicio de registro (service registry):**  
Este servicio centralizado será el encargado de proveer los endpoints de los servicios para su consumo. Todo microservicio se registrará automáticamente cuando estén disponibles en el ecosistema de la plataforma.
- **Microservicios:**  
Son los componentes que gestionan la lógica de dominio o de negocios. Es quien ejecuta los procesos de dominio de la aplicación.



Una **API** es un conjunto de **definiciones y protocolos** que se utilizan para diseñar e integrar el software de las aplicaciones. Se lo considera como un contrato entre el proveedor de información y quien la necesita, donde se define lo que se necesita por parte del consumidor (la llamada) y lo que se requiere por el lado del productor (la respuesta).



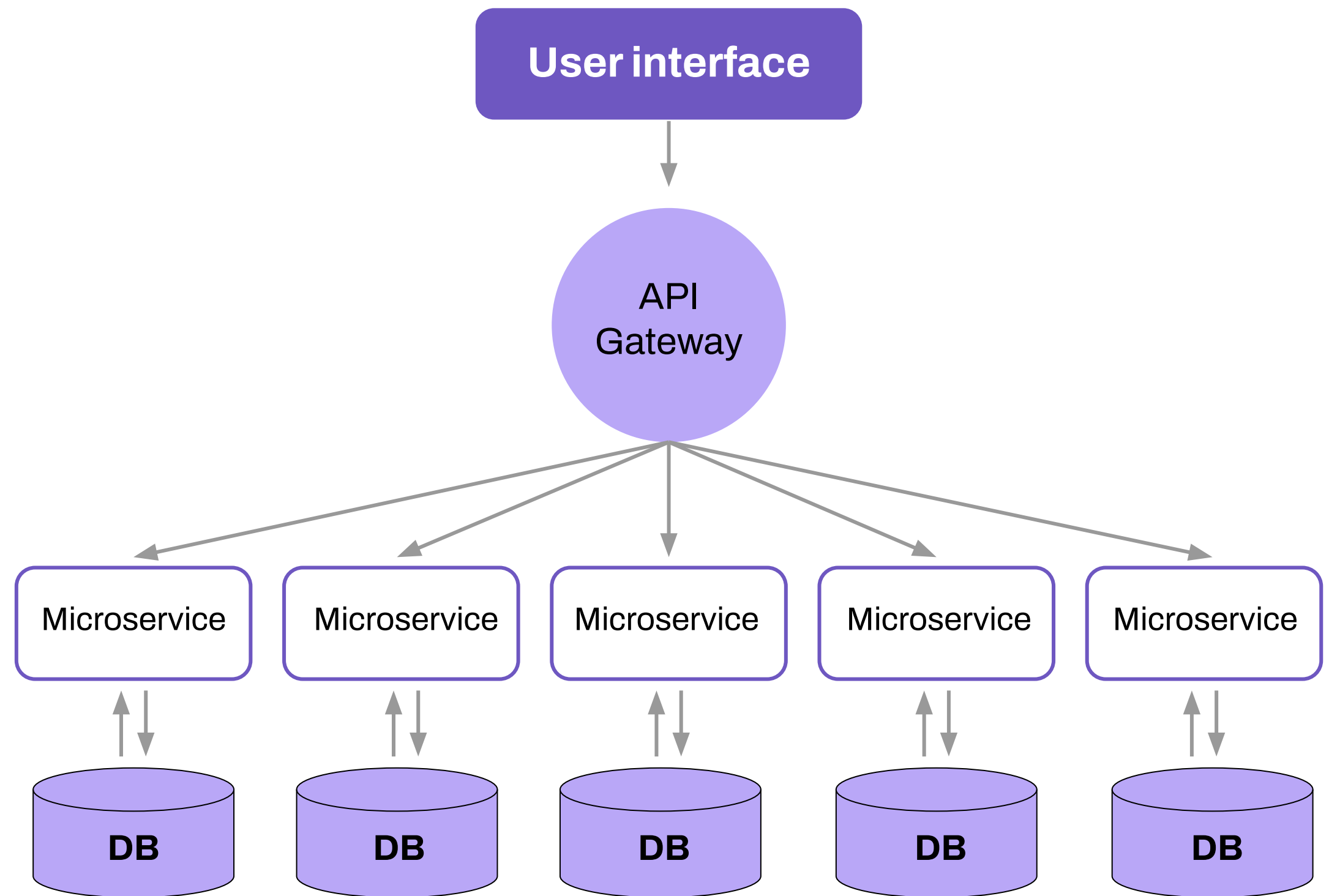
**Por ejemplo:** el diseño de una API de servicio meteorológico podría requerir que el usuario escribiera un código postal y que el productor diera una respuesta en dos partes: la primera sería la temperatura máxima y la segunda, la mínima.

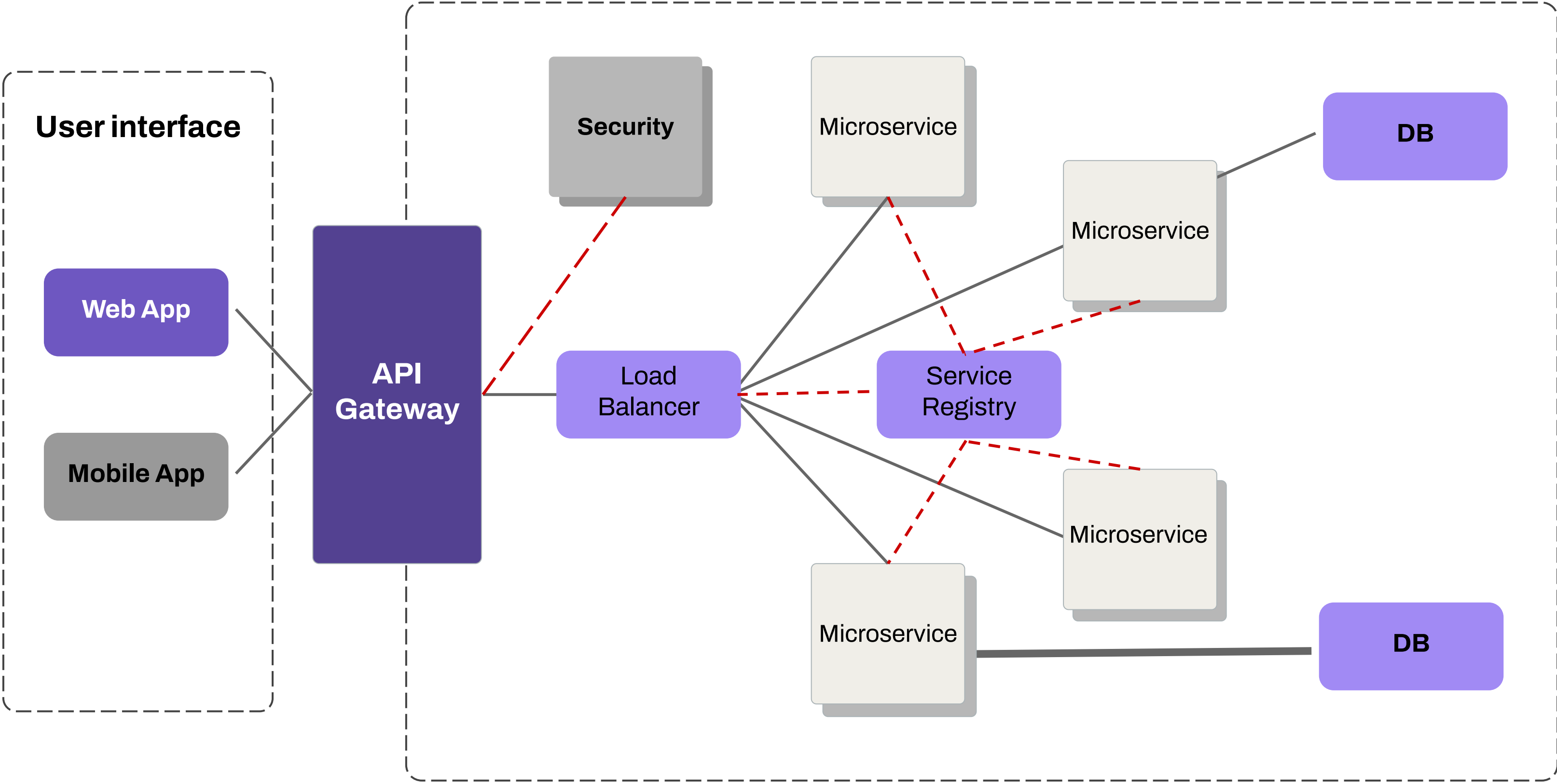


# En otras palabras, las **API** permiten interactuar con una computadora

- En otras palabras, las API permiten interactuar con una computadora o un sistema para obtener datos o ejecutar una función, de manera que el sistema comprenda la solicitud y la cumpla.
- Imaginemos que fueran los mediadores entre los usuarios o clientes y los recursos o servicios web que se quieren obtener.
- Con las API, las empresas pueden compartir recursos e información mientras conservan la seguridad, el control y la autenticación, lo cual les permite determinar el contenido al que puede acceder cada usuario.
- Otra ventaja de las API es que no necesitamos saber cómo se recibe el recurso ni de dónde proviene.

# Elementos de una API con microservicios







# La comunicación con la **API**:

- Sistemas legados dada la dificultad de actualizarlos por sus intrincadas dependencias.
- Acoplamiento excesivo de funcionalidades, código y dependencias.
- Deuda técnica por la complejidad de agregar cambios en el código.
- Errores frecuentes de lógica en las nuevas releases desplegadas.
- Complejidad al momento de escalar los sistemas.
- Poca tolerancia a fallos por la dificultad de implementar esquemas de alta disponibilidad.
- Es más complicado escalar los equipos y disminuye la flexibilidad.
- Al contener servicios acoplados entre sí, si ocurre un fallo en un servicio o componente, se podría propagar a todo el sistema fácilmente.

¡Muchas gracias!