API Gateway

In this research I researched what Api gateways do and why they are useful.

What is an API Gateway?

An API Gateway is a tool that sits between the user and the backend services. (What does an API gateway do?, 2019)

Why are API Gateways used?

An API gateway has multiple tasks. Such as **user authentication, load balancing, rate limiting, statistics, routing, billing, monitoring, policies, alerts and security**. An API gateway has a lot of benefits such as:

Providing Low Latency

- **Providing low latency** by optimizing the traffic, routing and balancing the load across services,
- **rate limiting** with x number of requests from the same user,
- **controlling concurrency** by specifying a max number of concurrent connections, monitoring the load for

- **dynamic load balancing** allowing the services to adjust to the current traffic especially useful for traffic spikes.

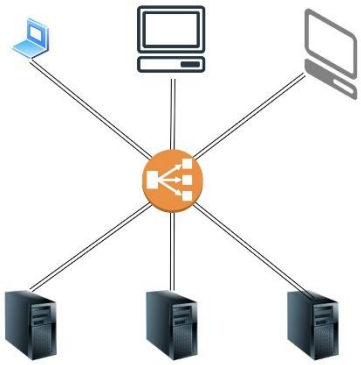
Cost Effectiveness

API gateways provide a more efficient way of working which saves money. By only scaling horizontally if the traffic goes up, we make sure we only use extra resources when necessary. And when the traffic spike is over, we scale back down. **You might want to make sure that the environment you are hosting your service in provides auto scaling** (What does an API gateway do?, 2019)

How does an API Gateway differ from load balancer?

Load balancer

A load balancer sits between the client and the backend services. It receives a request and decides what service to send it to. By doing this it prevents a service from overloading. A load balancer provides several benefits such as:



- preventing a direct connection between the server and the client providing a layer of security.
- The load balancer monitors the performance of services and can assign users to them according to their performance.
- Besides these core functionalities, there are some load balancers that do more such as caching or content aware routing.

Load balance use case

Whenever there are multiple servers or services it is useful to use a load balancer since it can detect which ones are overloaded or performing worse and it can react to that.

API Gateway

An API Gateway can also balance load, but on top of that it has more functionalities. Think of, traffic control, authentication, authorization, metrics and logging,

Using them combined?

This is a common practise.

Reverse Proxy?

Not suited for microservices.

Why use a Load balancer if an Api gateway can balance the load?

hm