

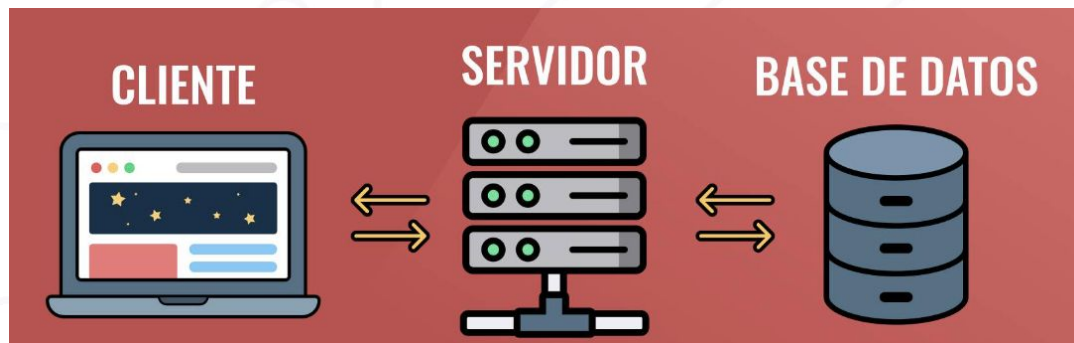
Arquitecturas de software

DEV.F
DESARROLLAMOS(PERSONAS);

dev

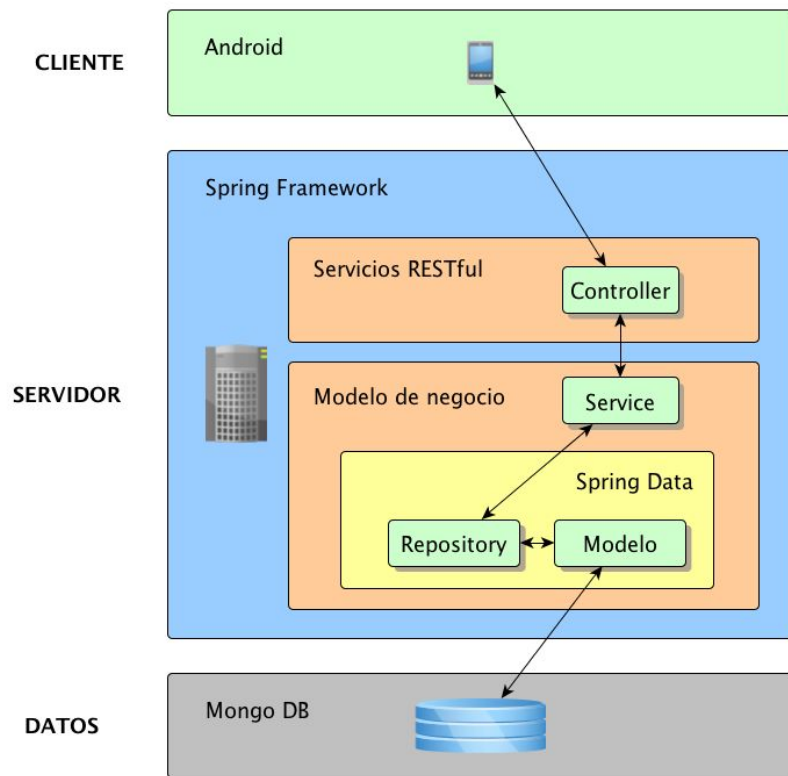
¿Qué es una arquitectura en software?

- Es la estructura y forma en que los componentes de software o hardware se distribuyen y relacionan dentro del stack tecnológico.
- La arquitectura de software es el diseño de más alto nivel de la estructura de un sistema.
- Definir las herramientas, patrones y lineamientos con los que una aplicación va a trabajar.



¿Por qué hablar de arquitectura?

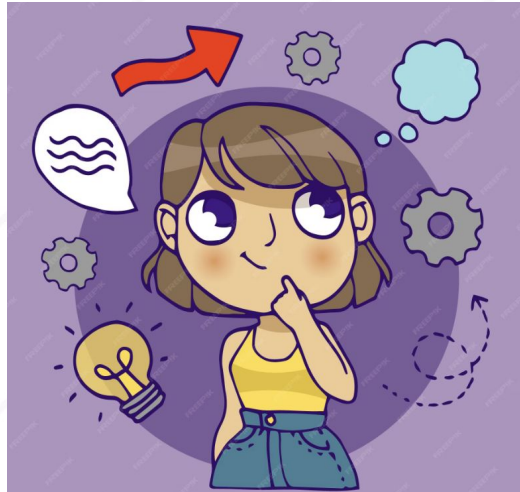
A semejanza de los planos de un edificio o construcción, estas indican la estructura, funcionamiento e interacción entre las partes del software.



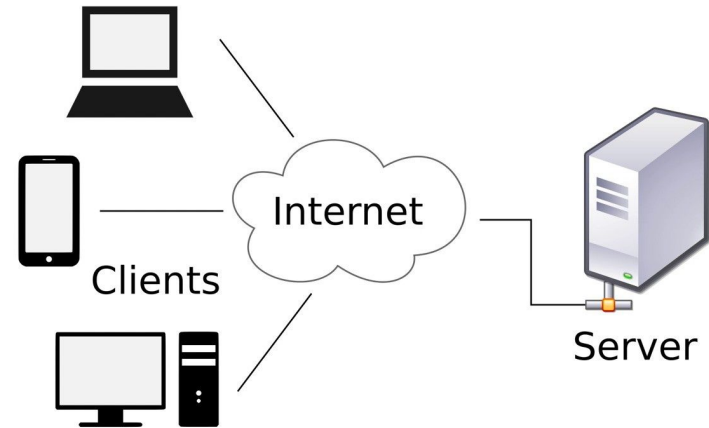
Arquitecturas más comunes

Generalmente, no es necesario inventar una nueva arquitectura de software para cada sistema de información.

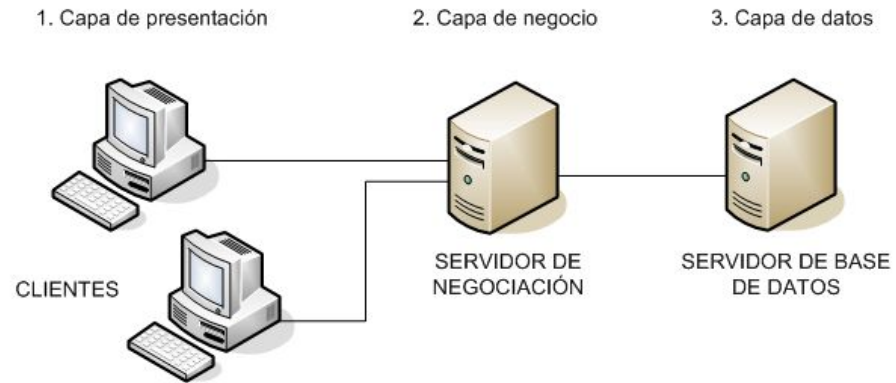
Lo habitual es adoptar una arquitectura conocida en función de sus ventajas e inconvenientes para cada caso en concreto.



Cliente Servidor



Arquitectura de Tres Niveles



Arquitectura MVC

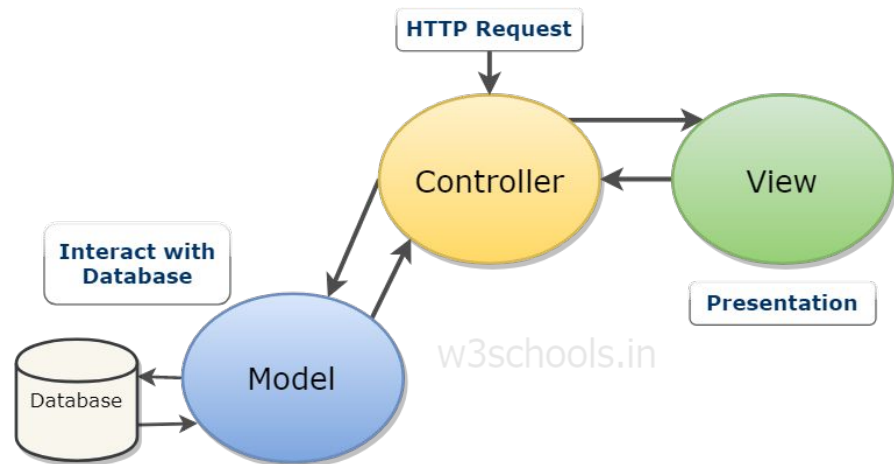


Fig: MVC Architecture

Arquitectura dirigida por eventos

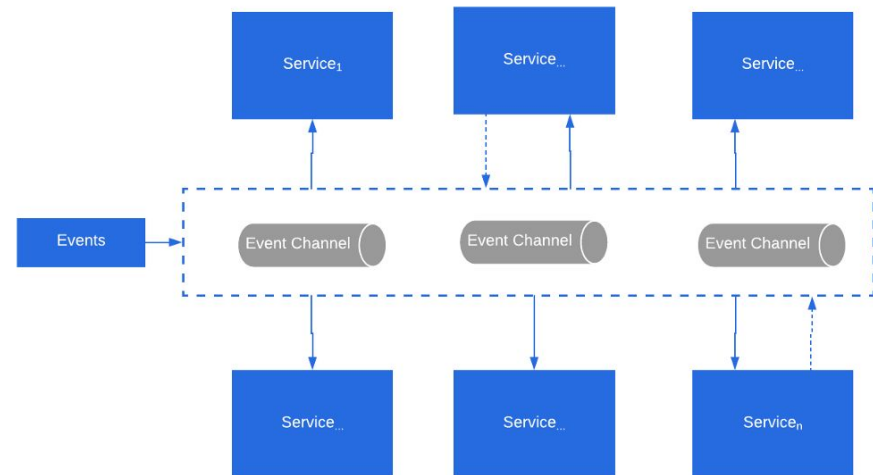
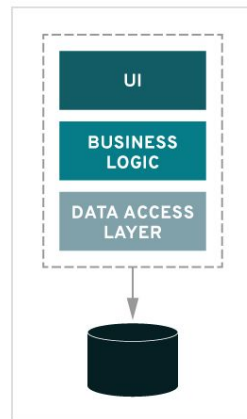


Figure 2 - Broker Topology

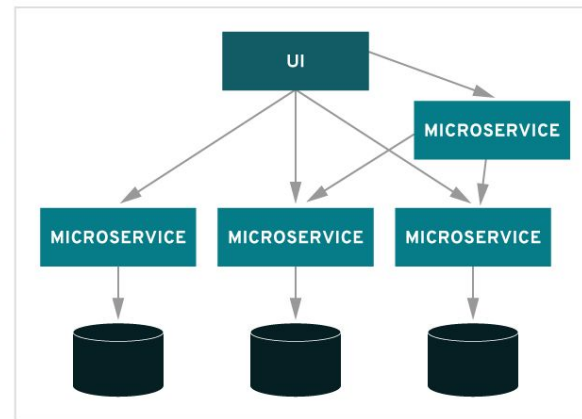
Arquitectura de Micro Servicios

MONOLITHIC



VS.

MICROSERVICES



Serverless

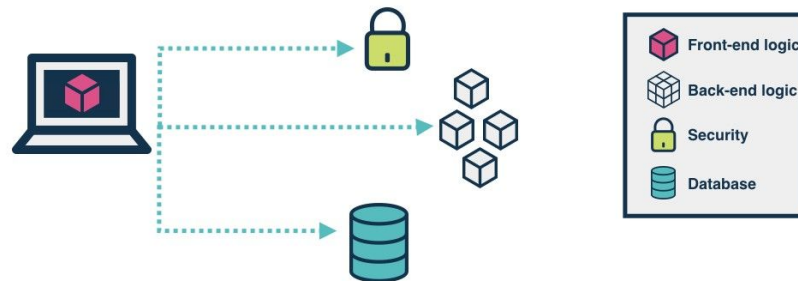
TRADITIONAL vs SERVERLESS

TRADITIONAL



SERVERLESS

(using client-side logic and third-party services)



Stacks de Desarrollo web

DEV.F
DESARROLLAMOS(PERSONAS);

dev

¿Qué es un stack tecnológico?

Es un **conjunto de servicios de software que se utilizan para el desarrollo de aplicaciones**. Normalmente, un Stack se conforma por lenguajes de programación, frameworks, bibliotecas, herramientas de desarrollo y enfoques de programación.

Hace referencia al método de apilamiento de los componentes de este conjunto de herramientas, uno encima del otro.

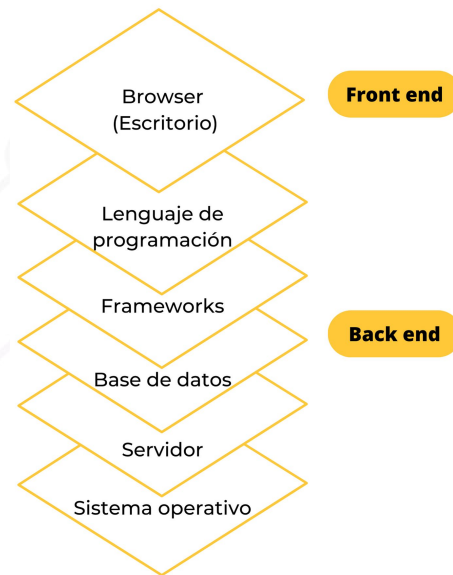
¿Qué es un stack tecnológico?

Un stack de **desarrollo** web se compone de:

- Las tecnologías utilizadas (y su distribución) para el desarrollo del front end, la API y el back end.

pero también de:

- Un sistema operativo.
- Un servidor web.
- Una base de datos.
- Un intérprete de lenguaje de programación.



LAMP



ASP.NET

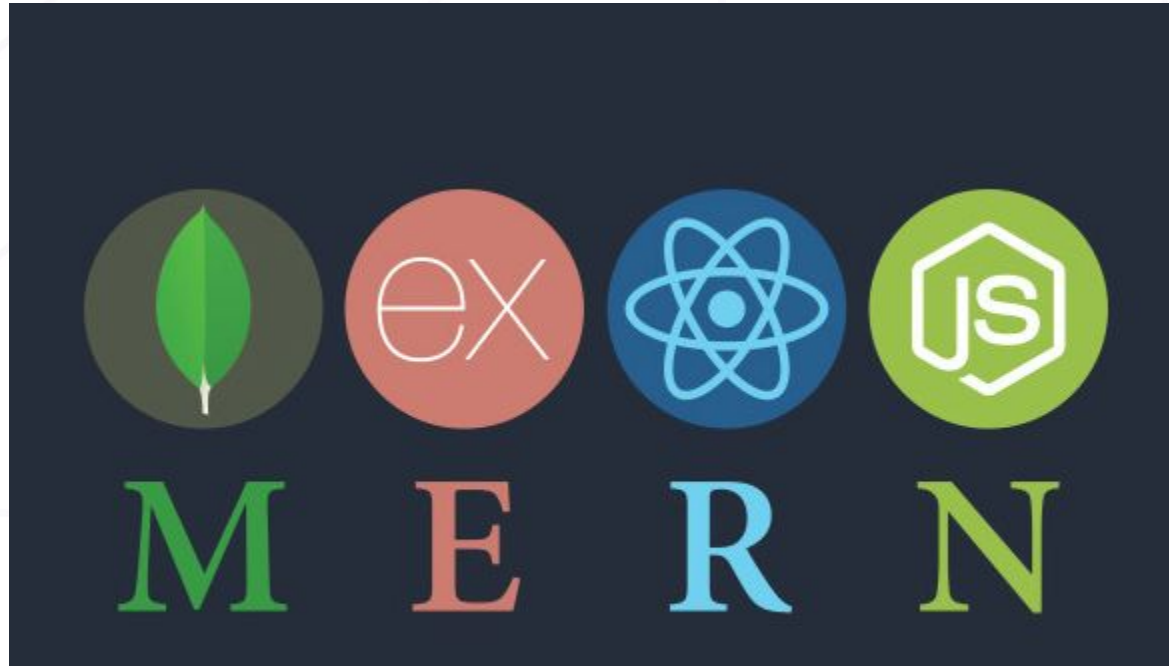


ASP.NET

MEAN



MERN



MEVN

MEVN

MONGODB - EXPRESS - VUEJS - NODEJS

Aptitudes de un desarrollador front end

- Dominio de Lenguajes de Programación y Marcado.
- Responsive Design y Cross-Browser Compatibility.
- Consumo de APIs.
- Diseño y Experiencia de Usuario (UX/UI).
- Capacidad de resolución de problemas.
- Conocimiento del negocio y sus reglas.
- Lenguajes:
 - Html.
 - CSS.
 - JS.



Aptitudes de un desarrollador back end

- Conocimiento en lenguajes de programación que funcionan del lado del servidor.
- Conocimiento sólido de bases de datos y técnicas de manejo de datos.
- Habilidad para trabajar con APIs y servicios externos.
- Comprensión de seguridad.
- Optimización y escalabilidad
- Capacidad de resolución de problemas.
- Conocimiento del negocio y sus reglas.
 - Lenguajes:
 - Java.
 - Node.
 - Php.

