**UNIVERSIDAD LA SALLE**

**CARRERA DE INGENIERÍA DE SISTEMAS**

****

**Las arquitecturas de software más usadas hoy en día**

**Equipo de desarrollo:**  Carlos Antonio Tarqui Guillen

**Carrera:** Ing de Sistemas

La Paz - Bolivia

2023

La arquitectura de software es el conjunto de decisiones de diseño importantes para organizar el software, y promover los atributos de calidad deseados.

La arquitectura de software define la estructura del software, las piezas que lo componen y la forma en que se comunican entre sí. También define los principios y patrones de diseño que se utilizarán para desarrollar el software.

Una buena arquitectura de software es aquella que:

* Es flexible y adaptable a los cambios.
* Es escalable para soportar un mayor volumen de tráfico o carga de trabajo.
* Es fácil de mantener y actualizar.
* Es segura y protege los datos del usuario.

La arquitectura de software es un aspecto fundamental del desarrollo de software. Una buena arquitectura puede ayudar a garantizar que el software sea exitoso y cumpla con los requisitos del usuario.

**Las arquitecturas de software más usadas hoy en día son las siguientes:**

Arquitectura en capas: Este patrón divide el software en capas, cada una de las cuales tiene un propósito específico. Las capas más comunes son la capa de presentación, la capa de negocio y la capa de datos.

Arquitectura cliente-servidor: Este patrón divide el software en dos componentes: un cliente que solicita servicios a un servidor.

Arquitectura de microservicios: Este patrón divide el software en pequeños servicios independientes que se comunican entre sí a través de una red.

Arquitectura basada en eventos: Este patrón se basa en la comunicación entre componentes a través de eventos.

Arquitectura basada en el espacio: Este patrón se basa en la distribución del software en diferentes máquinas.

**Arquitectura en capas**

La arquitectura en capas es un patrón de diseño de software que divide el software en capas, cada una de las cuales tiene un propósito específico. Las capas más comunes son la capa de presentación, la capa de negocio y la capa de datos.

Capa de presentación: Es la capa que se comunica con el usuario. Se encarga de mostrar la interfaz de usuario y de procesar las entradas del usuario.

Capa de negocio: Es la capa que implementa la lógica de negocio. Se encarga de realizar los cálculos y las operaciones necesarias para el funcionamiento del software.

Capa de datos: Es la capa que almacena los datos. Se encarga de almacenar y recuperar los datos del software.

La arquitectura en capas es un patrón sencillo de entender y de implementar. Es un buen punto de partida para la mayoría de los proyectos de software.

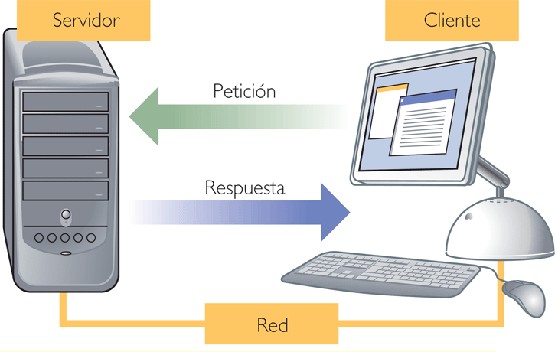
**Arquitectura cliente-servidor**

La arquitectura cliente-servidor es un patrón de diseño de software que divide el software en dos componentes: un cliente que solicita servicios a un servidor.

Cliente: Es el componente que solicita servicios al servidor.

Servidor: Es el componente que proporciona servicios al cliente.

La arquitectura cliente-servidor es un patrón muy popular para aplicaciones web y móviles. Permite distribuir el software en diferentes máquinas, lo que facilita el escalado y la gestión.



**Arquitectura de microservicios**

La arquitectura de microservicios es un patrón de diseño de software que divide el software en pequeños servicios independientes que se comunican entre sí a través de una red.

Microservicio: Es un servicio pequeño e independiente que implementa una función específica.

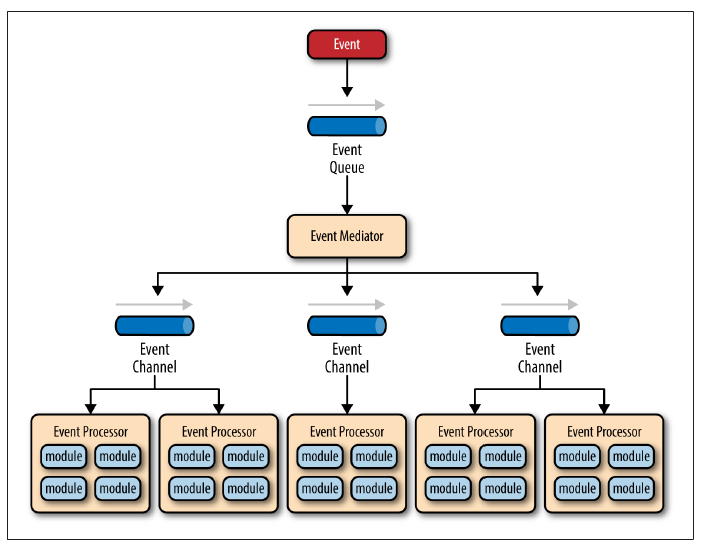
La arquitectura de microservicios es un patrón muy flexible y escalable. Permite desarrollar aplicaciones complejas de forma modular y facilita el mantenimiento y la actualización.

**Arquitectura basada en eventos**

La arquitectura basada en eventos es un patrón de diseño de software que se basa en la comunicación entre componentes a través de eventos.

Evento: Es un mensaje que se envía a un componente para notificarle de un cambio.

La arquitectura basada en eventos es un patrón muy eficiente para la comunicación entre componentes. Permite que los componentes se comuniquen sin necesidad de conocer la existencia del otro.



**Arquitectura basada en el espacio**

La arquitectura basada en el espacio es un patrón de diseño de software que se basa en la distribución del software en diferentes máquinas.

Arquitectura distribuida: Es una arquitectura en la que los componentes del software se encuentran en diferentes máquinas.

Arquitectura de nube: Es una arquitectura distribuida que se ejecuta en un entorno de computación en la nube.

La arquitectura basada en el espacio es un patrón muy escalable y permite aprovechar la potencia de las máquinas virtuales.



**Elección de la arquitectura de software**

La elección de la arquitectura de software adecuada depende de varios factores, como el tamaño y la complejidad del software, los requisitos de rendimiento y escalabilidad, y las limitaciones de tiempo y presupuesto.

En general, la arquitectura en capas es un buen punto de partida para la mayoría de los proyectos de software. La arquitectura cliente-servidor es una buena opción para aplicaciones web y móviles. La arquitectura de microservicios es una buena opción para aplicaciones complejas y escalables. La arquitectura basada en eventos es una buena opción para aplicaciones que requier en una comunicación eficiente entre componentes. La arquitectura basada en el espacio es una buena opción para aplicaciones que requieren un alto rendimiento y escalabilidad.