

PROGRAMACIÓN CLIENTE-SERVIDOR

ISAAC VIEYRA SANTOS

ING. EN SOFTWARE

ANDRES LUNA JAIMES

PROGRAMACIÓN CLIENTE-SERVIDOR

6TO CUATRIMESTRE

UNIVERSIDAD POLITÉCNICA DE LÁZARO CÁRDENAS



1. ¿Qué es una arquitectura de software?

La arquitectura de software es el diseño estructural fundamental de un sistema, que define:

- **Componentes principales** (módulos, bases de datos, interfaces).
- **Cómo interactúan entre sí** (comunicación, flujo de datos).
- **Principios y patrones que guían su organización** (ej: escalabilidad, seguridad).

tienen como objetivo garantizar que el sistema sea mantenible, escalable y eficiente.

2. ¿Qué es una arquitectura cliente-servidor?

Se define como un modelo donde el Cliente solicita recursos o servicios (ej: navegador web, app móvil), y el Servidor procesa las solicitudes y devuelve respuestas (ej: servidor web, API REST).

Componentes:

Cliente

- Interfaz de usuario (UI)
- Solicitudes HTTP/API
- Navegador o app

Servidor

- Lógica de negocio
- Base de datos
- Autenticación/autorización

Funcionamiento

Cómo funciona:

1. el Cliente envía una petición (ej: "GET /users").
2. el Servidor procesa la petición (consulta DB, aplica reglas).
3. el Servidor responde (ej: JSON con datos de usuarios).
4. el Cliente muestra los datos al usuario.

3. Tipos de arquitecturas por capas

a) Monolítica (1 capa)

- Descripción: Todo el código (UI, lógica, DB) está en una sola aplicación.
- Ventaja: Simple de desarrollar.
- Desventaja: Difícil escalar o actualizar.
- Ejemplo: Aplicaciones desktop tradicionales.

b) 2 capas (Cliente-Servidor básico)

Capas:

- Capa de presentación (cliente).
- Capa de datos (servidor + DB).
- Ejemplo: App de escritorio que se conecta a una base de datos local.

c) 3 capas (MVC común)

Capas:

- Presentación: Interfaz de usuario (HTML, app móvil).
- Lógica de negocio: Reglas y procesos (API, servicios).
- Datos: Almacenamiento (MySQL, MongoDB).
- Ventajas:
 - Escalabilidad independiente por capa.
 - Mantenimiento más fácil.
- Ejemplo:
 - Cliente (React.js).
 - Servidor Node.js (API REST).
 - DB: PostgreSQL.

d) N-capas (Microservicios)

- Descripción: Divide la lógica en servicios independientes (ej: auth, pagos).
- Ejemplo: Netflix usa microservicios para recomendar películas, gestionar usuarios, etc.

Diagrama de la arquitectura de una app web

