

Introducción al Backend y Arquitectura Cliente-Servidor

Curso: Desarrollo Backend con Django

Universidad de los Andes | Vigilada Mineducación. Reconocimiento como Universidad: Decreto 1297 del 30 de mayo de 1964.
Reconocimiento personería jurídica: Resolución 28 del 23 de febrero de 1949 MinJusticia.



Introducción al Backend y Arquitectura Cliente- Servidor

Curso: Desarrollo Backend con Django

Educación Continua Uniandes

¿Qué es el Backend?

El **backend** es la capa de una aplicación que se ejecuta en el **servidor** y que los usuarios no ven directamente.

Es responsable de:

- Procesamiento de datos
- Lógica de negocio
- Seguridad y autenticación
- Comunicación con bases de datos
- Servicios e integración con APIs



Arquitectura Cliente-Servidor



Cliente

Interfaz de usuario que solicita servicios a través del navegador o aplicación móvil



Internet

Red que transmite las solicitudes HTTP y respuestas entre cliente y servidor



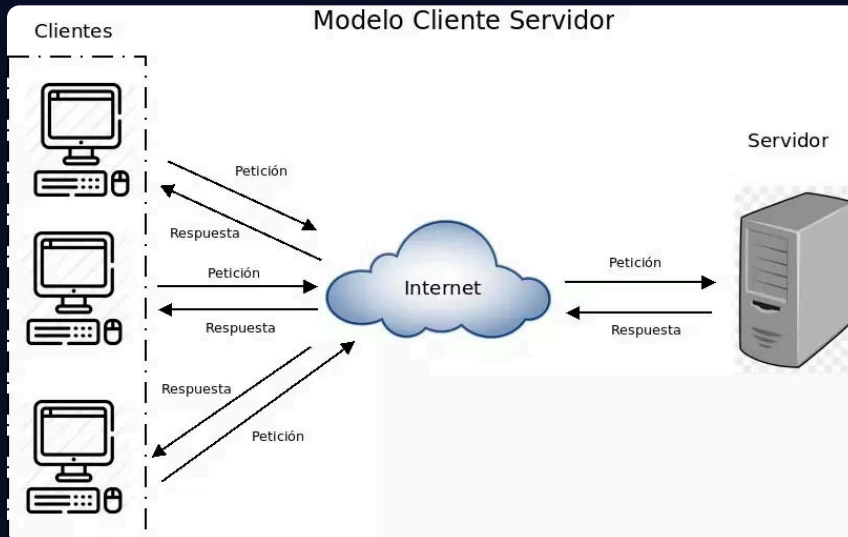
Servidor

Procesa solicitudes, ejecuta lógica de negocio y devuelve resultados al cliente

La arquitectura cliente-servidor establece una clara **separación de responsabilidades**, mejorando la escalabilidad, seguridad y mantenimiento de las aplicaciones.

Flujo de Datos Cliente-Servidor

1. El **cliente** genera una solicitud (ej. clic en un botón de compra)
2. La solicitud viaja por **internet** mediante protocolos como HTTP/HTTPS
3. El **servidor** recibe la solicitud y la procesa (verificación, autenticación)
4. Si es necesario, consulta la **base de datos** para obtener o almacenar información
5. El servidor genera una **respuesta** (generalmente en formato JSON)
6. La respuesta viaja de vuelta al **cliente** para actualizar la interfaz



Backend vs. Frontend

Frontend

¿Qué es? Interfaz visible para el usuario

Tecnologías: HTML, CSS, JavaScript, React, Angular, Vue

Enfoque: Experiencia de usuario, diseño, interactividad

Procesamiento: En el navegador del cliente

Backend

¿Qué es? Lógica e infraestructura detrás de la interfaz

Tecnologías: Node.js, Python, Java, PHP, SQL, MongoDB

Enfoque: Procesamiento de datos, seguridad, lógica de negocio

Procesamiento: En servidores remotos

Ambos son importantes para la creación de una aplicación completa y funcional.

La interacción entre frontend y backend ocurre principalmente a través de **APIs** que definen cómo pueden comunicarse.

Componentes Clave del Backend



Servidor

Computadora o programa que procesa solicitudes y envía respuestas. Ejemplos: Apache, Nginx, Express.



Base de Datos

Sistema para almacenar y recuperar datos estructurados. Ejemplos: MySQL, MongoDB, PostgreSQL.



API

Interfaz que permite la comunicación entre diferentes sistemas de software mediante reglas definidas.



Lógica de Negocio

Conjunto de reglas que determinan cómo los datos pueden ser creados, almacenados y modificados.

Casos Reales: Arquitectura Cliente-Servidor

Netflix



- **Ciente:** Aplicación web, móvil, Smart TV
- **Backend:** Microservicios, recomendaciones, gestión de contenido
- **Datos:** Perfiles, historial, preferencias

Instagram



- **Ciente:** App móvil, versión web
- **Backend:** Gestión de usuarios, fotos, feed
- **Datos:** Imágenes, comentarios, relaciones

Ventajas de la Arquitectura Cliente-Servidor

Escalabilidad

Permite aumentar la capacidad del sistema añadiendo más servidores o mejorando hardware sin afectar al cliente.

Seguridad

La lógica crítica y los datos sensibles residen en el servidor, protegidos de manipulación directa por usuarios.

Centralización

Administración centralizada de datos y procesos, facilitando actualizaciones y mantenimiento.

Especialización

Permite que desarrolladores se especialicen en frontend o backend, optimizando el desarrollo.



Conceptos Clave a Recordar

❗ El **backend** es fundamental para la seguridad, rendimiento y funcionalidad de cualquier aplicación web moderna.

1 Separación de Responsabilidades

Cliente (presentación) y Servidor (procesamiento) tienen roles claramente definidos.

2 Comunicación Estandarizada

APIs RESTful y GraphQL facilitan la interacción cliente-servidor mediante protocolos bien definidos.

3 Seguridad como Prioridad

La validación de datos, autenticación y autorización deben implementarse rigurosamente en el backend.



Preguntas para Reflexionar

1

Tecnologías

¿Qué tecnologías de backend son más demandadas actualmente en el mercado laboral?

2

Escalabilidad

¿Cómo afecta la arquitectura elegida a la capacidad de crecimiento de una aplicación?

3

Seguridad

¿Qué vulnerabilidades comunes existen en la arquitectura cliente-servidor?

4

Futuro

¿Cómo están cambiando los paradigmas de backend con tecnologías como serverless y edge computing?

#EDCOUNIANDES

<https://educacioncontinua.uniandes.edu.co/>

Contacto: educacion.continua@uniandes.edu.co

© - Derechos Reservados: La presente obra, y en general todos sus contenidos, se encuentran protegidos por las normas internacionales y nacionales vigentes sobre propiedad Intelectual, por lo tanto su utilización parcial o total, reproducción, comunicación pública, transformación, distribución, alquiler, préstamo público e importación, total o parcial, en todo o en parte, en formato impreso o digital y en cualquier formato conocido o por conocer, se encuentran prohibidos, y solo serán lícitos en la medida en que se cuente con la autorización previa y expresa por escrito de la Universidad de los Andes.