

1. Servidor web vs Servidor de aplicaciones

- Servidor web (Nginx, Apache): Maneja solicitudes HTTP y sirve contenido estático (HTML, CSS, imágenes). Puede actuar como proxy inverso.
- Servidor de aplicaciones (Gunicorn, uWSGI): Ejecuta la lógica de la aplicación (ej. código Python en Django), procesa peticiones dinámicas y se comunica con la base de datos.
 - ♦ *En producción, trabajan juntos: el servidor web gestiona el tráfico y redirige las peticiones dinámicas al servidor de aplicaciones.*

2. Protocolo DNS

- Traduce nombres de dominio (como reservasrestaurante.com) en direcciones IP (como 192.168.1.1).
- Funciona como una "guía telefónica de internet": cuando el usuario entra a un sitio, el navegador consulta un servidor DNS para obtener la IP del servidor.
 - ♦ *Permite acceder a sitios web usando nombres fáciles de recordar, no direcciones numéricas.*

3. Modelo y Vista en Django (MVT)

- Modelo: Define la estructura de los datos y su almacenamiento en la base de datos (ej. el modelo Reserva con campos como fecha y nombre).
- Vista: Procesa las solicitudes del usuario, interactúa con el modelo (lee o guarda datos) y devuelve una respuesta (por ejemplo, renderizando una plantilla).
 - ♦ *La vista usa el modelo para gestionar datos y los envía a la plantilla para mostrarlos al usuario.*

4. HTTPS vs HTTP

- HTTP: Protocolo sin cifrado; los datos viajan en texto plano → inseguro.
- HTTPS: Versión segura con cifrado SSL/TLS. Protege la información durante la transmisión.
 - ♦ *Ventajas en un sistema de reservas:*
 - Protege datos personales (nombre, hora, contacto).
 - Evita ataques de intermediarios (man-in-the-middle).
 - Aumenta la confianza del usuario (candado en el navegador).

- Es esencial para formularios y requisitos de seguridad modernos.