

# FUNDAMENTOS DE UN PROYECTO WEB

1

## PROYECTO WEB

*Es el proceso de diseñar, desarrollar y mantener un sitio o aplicación web que satisfaga necesidades específicas (información, venta, educación, interacción, etc.).*

### Características principales:

- Tiene un objetivo definido.
- Es medible en tiempos, costos y resultados.
- Requiere planificación, fases y control de calidad.
- Puede crecer (escalabilidad) y adaptarse a nuevas necesidades.

### Tipos más comunes:

- Informativos: blogs, periódicos digitales, sitios institucionales.
- Comerciales: tiendas online (e-commerce), reservas, catálogos de productos.
- Educativos: plataformas e-learning, aulas virtuales.
- Corporativos o sociales: intranets, redes sociales, foros.

2

## ESTRUCTURA DE UN PROYECTO WEB

*organización frontend, backend y modelo cliente-servidor*

### Frontend:

- HTML: estructura y contenido.
- CSS: estilos, diseño visual y adaptabilidad (responsive design).
- JavaScript: interactividad, animaciones, validaciones.

### Backend:

- Lenguajes: PHP, Python, Ruby, Java, Node.js.
- Base de datos: MySQL, PostgreSQL, MongoDB.
- Servidores: Apache, Nginx.

### Arquitectura Cliente-Servidor:

- Cliente (navegador) → envía petición.
- Servidor → procesa, consulta la base de datos y responde.

3

## COMPONENTES ESENCIALES

*UI, UX, seguridad y accesibilidad garantizada.*

- **UI (Interfaz de Usuario):** diseño gráfico, botones, formularios, menús, tipografía.
- **UX (Experiencia de Usuario):** navegación clara, velocidad de carga, accesibilidad.
- **Seguridad:** encriptación (HTTPS, SSL), gestión de usuarios, protección contra ataques (XSS, SQL Injection).
- **Accesibilidad:** compatibilidad con lectores de pantalla, colores adecuados, estándares internacionales (WCAG).

4

## HERRAMIENTAS Y TECNOLOGÍAS

*editores, frameworks, CMS y control versiones.*

- **Editores de código:** Visual Studio Code, Atom, Sublime Text.
- **Frameworks y Librerías:**
  - **Frontend:** React, Angular, Vue.js.
  - **Backend:** Laravel (PHP), Django (Python), Express (Node.js).
- **CMS (Gestores de Contenido):** WordPress, Joomla, Drupal (útiles para proyectos rápidos).
- **Control de versiones:** Git (seguimiento de cambios, trabajo en equipo, ramas de desarrollo).
- **Entornos de trabajo colaborativos:** GitHub, GitLab, Bitbucket.

5

## DOCUMENTACIÓN TÉCNICA

*requisitos, diagramas y cronograma del proyecto.*

- **Requisitos funcionales:** funciones que debe cumplir (ejemplo: "el usuario puede registrarse con correo y contraseña").
- **Requisitos no funcionales:** criterios de calidad (ejemplo: "el sistema debe soportar 1000 usuarios simultáneos").
- **Diagramas:**
  - Flujo de procesos.
  - Casos de uso.
  - Arquitectura general.
- **Cronograma:** fases de análisis, diseño, codificación, pruebas, implementación y mantenimiento.