

Investigación sobre entornos tecnológicos
(Frameworks) y lenguajes de programación para
servidores web



JAVA.-

Java es un lenguaje de programación orientado a objetos, robusto y multiplataforma, usado ampliamente en desarrollo empresarial, aplicaciones móviles (Android), backend y más.

Características:

- Compilado y ejecutado mediante la **Java Virtual Machine (JVM)**.
- Muy usado en sistemas bancarios, ERP, y grandes corporativos.
- Fuerte ecosistema de frameworks.

Frameworks populares:

- **Spring Boot**: Para crear APIs y servicios web rápidamente.
- **Hibernate**: ORM para conexión con bases de datos.
- **JavaFX**: Para interfaces gráficas (GUI).

Ventajas:

- Portabilidad (“escribe una vez, corre en cualquier parte”).
- Alto rendimiento en entornos de gran escala.
- Amplio soporte empresarial.

NET Core.-

.NET Core es un framework de desarrollo de código abierto de Microsoft, multiplataforma (Windows, Linux, macOS), usado para crear APIs, aplicaciones web, de escritorio y servicios.

Características:

- Soporta múltiples lenguajes: **C#**, F#, VB.NET.
- Arquitectura modular y rápida.
- Excelente integración con Azure.

Frameworks principales:

- **ASP.NET Core**: Para desarrollar aplicaciones web modernas y APIs REST.
- **Entity Framework Core**: ORM para bases de datos.

Ventajas:

- Multiplataforma.

- Alto rendimiento.
- Soporte oficial a largo plazo (LTS).

PHP.-

PHP es un lenguaje de programación interpretado del lado del servidor, ampliamente utilizado en desarrollo web. Fue uno de los pilares de la web dinámica.

Características:

- Rápido de aprender y fácil de implementar.
- Muy usado para CMS como WordPress, Joomla, Drupal.
- Compatible con la mayoría de servidores web y bases de datos.

Frameworks populares:

- **Laravel:** El más moderno, con enfoque en sintaxis elegante.
- **Symfony:** Escalable y robusto para grandes sistemas.
- **CodeIgniter:** Liviano y simple para proyectos pequeños.

Ventajas:

- Amplio soporte de hosting.
- Gran comunidad.
- Buen rendimiento para aplicaciones web tradicionales.

Python.-

Python es un lenguaje interpretado, de alto nivel, muy popular por su simplicidad y legibilidad. Se usa en web, ciencia de datos, automatización, IA, y más.

Características:

- Sintaxis clara y fácil de aprender.
- Soporte para múltiples paradigmas (OOP, funcional, imperativo).
- Bibliotecas potentes para ciencia, IA, y web.

Frameworks populares:

- **Django:** Framework completo para desarrollo web rápido.
- **Flask:** Microframework liviano y flexible.
- **FastAPI:** Para APIs modernas con validación automática.

Ventajas:

- Código limpio y conciso.
- Muy usado en universidades, startups y proyectos científicos.
- Gran comunidad y ecosistema.

Node.js.-

Node.js es un entorno de ejecución para JavaScript en el servidor, construido sobre el motor V8 de Chrome. Ideal para aplicaciones web en tiempo real y APIs.

Características:

- Asincronía no bloqueante.
- Usa JavaScript tanto en frontend como en backend.
- Perfecto para microservicios y APIs RESTful.

Frameworks populares:

- **Express.js:** El más usado para APIs REST.
- **NestJS:** Inspirado en Angular, estructura modular y escalable.
- **Socket.io:** Comunicación en tiempo real.

Ventajas:

- Un solo lenguaje (JavaScript) para todo el stack.
- Rápido y ligero para I/O intensivo.
- Comunidad enorme y en crecimiento.