

### **Desarrollo WEB en Entorno Servidor:**

Comparativa de lenguajes y Frameworks

**Alumno:** Fara Santeyana, María Guillermina.

**NIA:** 13298631.

**Curso:** 2do DAW.

**Instituto:** I.E.S La Sènia.

**Año:** 2025-2026.

Realiza una tabla comparativa entre las tecnologías de programación web en entorno servidor: **PHP, Node.js, Python y Java**, analizando sus características, ventajas, frameworks asociados y casos de uso recomendados:

- Características del lenguaje
- Ventajas
- Inconvenientes
- Frameworks disponibles
- Situaciones o casos de uso recomendados
- Facilidad de aprendizaje

Escoge una de las opciones en la que preferirías desarrollar un proyecto y justifica tu elección.

**RA1.a, RA1.e, RA1.g**

Como elección de un lenguaje para desarrollar de un proyecto sería Java, porque tengo mayor experiencia y conocimiento, conozco mejor su alcance; cuenta con una muy buena documentación lo que facilita conocer sus bibliotecas y usos; su aplicación es sumamente versátil. En conjunto con SpringBoot que integra muchas funcionalidades haciendo fácil y potente su implementación.

	PHP	Node.js	Python	Java
Características	Lenguaje de scripting, código abierto, POO, puede embeberse con HTML Sintaxis tomada de C, Java y Perl.	Es un entorno de tiempo de ejecución de JavaScript. Impulsado por eventos, asíncrono, de código abierto.	Lenguaje de código abierto, interpretado, tipado dinámicamente, alto nivel, orientado a objetos, multiplataforma.	Lenguaje compilado, de código abierto, orientado a objetos, multiplataforma, APIs integradas. Tipado fuerte y estático.
Ventajas	*Fácil implementación. *Soporta gran cantidad de datos. *Permite automatizar tareas.	*Gran escalabilidad. *Muy rápido y eficiente. *Puede ser backend como frontend *Comunidad activa, actualizaciones, documentación. *Encaja en metodologías ágiles.	*Sintaxis clara. *Biblioteca amplia, ahorrando escribir código desde 0. *Gran soporte de su comunidad. *Fácil integración con otros lenguajes.	*Características de seguridad integradas. *Escalabilidad. *Fuerte soporte de la comunidad. *Portabilidad no necesita grandes modificaciones. *JVM, funciona como capa intermedia entre el código y hardware.
Inconvenientes	*Mala calidad de manejo de errores. *Baja seguridad. *Necesita un hosting para ejecutarse.	*Las tareas que exigen uso intensivo de CPU pueden bloquear eventos y afectar rendimiento. *Algunas librerías no están mantenidas o fallan en seguridad	*Velocidad de ejecución lenta al ser interpretado. *Presenta dificultades en operaciones asíncronas *Consumo excesivo de memoria. *No recomendado para aplicaciones móviles.	*Arranque lento. *Consumo excesivo de memoria que puede ser crítico en ciertos entornos. *Problemas al integrar bibliotecas no javas o lenguajes diferentes.
Frameworks	Laravel, Symfony, Zend, CodeIgniter, Phalcon.	Express.js, Hapi.js, Nest.js, Adonis.js	Django, Flask, Pyramid, Tornado, PyQt.	JavaServer Faces, Spring Framework, Hibernate.

Usos recomendados	<ul style="list-style-type: none"> <li>*Desarrollo web.</li> <li>*Manipulación de BBDD.</li> <li>*Proceso de formularios.</li> <li>*APIs web.</li> <li>*Automatización de tareas.</li> </ul>	<ul style="list-style-type: none"> <li>*APIS y microservicios</li> <li>*Plataformas streaming</li> <li>*juegos online.</li> <li>*Para envío de archivos de gran peso.</li> <li>*Aplicaciones en tiempo real.</li> <li>*Aplicaciones de comercio electrónico.</li> <li>* Internet of Things.</li> </ul>	<ul style="list-style-type: none"> <li>- Análisis de datos.</li> <li>- Inteligencia Artificial, aprendizaje automático permite a las máquinas aprender patrones y tendencias.</li> <li>*Aprendizaje profundo, procesamiento del lenguaje Natural.</li> <li>-Desarrollo Web.</li> <li>-Automatización de tareas y scripting.</li> </ul>	<ul style="list-style-type: none"> <li>*Aplicaciones de escritorio.</li> <li>*Apps Android.</li> <li>*Apps IDE Internet of Things.</li> <li>*Aplicaciones empresariales.</li> </ul>
Facilidad de aprendizaje	<p>Fácil implementación.</p> <p>Fácil aprendizaje.</p>	<p>Fácil de aprender. Con excepción de aprendizaje de asincronía.</p>	<p>Fácil configuración.</p> <p>Fácil Aprendizaje</p>	<p>No tan fácil aprendizaje.</p>

## **Bibliografía**

<https://contabo.com/blog/es/como-crear-un-servidor-web-local-con-python/>

<https://kinsta.com/es/blog/python-frameworks/>

<https://aws.amazon.com/es/es/what-is/python/>

<https://keepcoding.io/blog/ventajas-y-desventajas-de-python/>

<https://talently.tech/blog/python-ventajas-y-desventajas/>

<https://www.php.net/manual/es/faq.general.php>

<https://www.hostinger.com/es/tutoriales/mejores-frameworks-php>

<https://talently.tech/blog/ventajas-de-php/>

<https://www.arsys.es/blog/php#tree-2>

<https://kinsta.com/es/base-de-conocimiento/que-es-node-js/>

<https://openwebinars.net/blog/que-es-nodejs/>

<https://www.mytaskpanel.com/the-4-most-used-node-js-frameworks-read-our-list/>

<https://keepcoding.io/desarrollo-web/5-aplicaciones-de-nodejs/>

<https://www.solbyte.com/blog/nodejs/>

<https://www.cursosaula21.com/que-es-java/>

[https://es.wikipedia.org/wiki/Java\\_\(lenguaje\\_de\\_programaci%C3%B3n\)](https://es.wikipedia.org/wiki/Java_(lenguaje_de_programaci%C3%B3n))

<https://teclab.edu.ar/tecnologia-y-desarrollo/ventajas-y-desventajas-de-java/>

<https://imagineinformacion.com/tutoriales/los-3-mejores-frameworks-de-java>