

# Tecnologías para el Desarrollo del Proyecto

EVALUACIÓN DE  
DESEMPEÑO LABORAL POR  
OBJETIVOS

2024

## **INTEGRANTES:**

Keyla Muñoz - 7191

Diego Campos - 7161

## COMPARATIVA DE TECNOLOGÍAS

### Lenguajes de Programación

Criterio	JavaScript/TypeScript	Python	Java	PHP
Popularidad	Muy Alta	Muy Alta	Alta	Alta
Compatibilidad Web	Excelente (Frontend y Backend)	Regular (Backend principal)	Regular (Backend principal)	Excelente (Principalmente Backend)
Curva de Aprendizaje	Baja (JS) / Media (TS)	Baja	Media	Media
Soporte a Frameworks	Amplio (Nest, React, Next.js)	Amplio (Django, Flask)	Amplio (Spring Boot)	Amplio (Laravel, Symfony)
Ecosistema y Comunidad	Muy Amplio	Amplio	Amplio	Medio

**Resultado:** *JavaScript/TypeScript*, se elige debido a que es un lenguaje con alta popularidad, tiene una excelente compatibilidad web tanto para el frontend como para el backend, tiene un amplio soporte de frameworks así como de ecosistemas y comunidad. En conclusión, entre todos los lenguajes planteados es el ideal para el desarrollo de nuestro proyecto cumpliendo con la mayoría de las características. No obstante, también se hará uso de Python para la aplicación de Análisis de Sentimientos con Deep Learning, como se puede observar Python es una gran opción por su alta popularidad, una curva de aprendizaje baja y su amplio uso en aplicaciones de IA.

### IDE's

Criterio	Visual Studio Code	PyCharm	IntelliJ IDEA	NetBeans
Compatibilidad	Excelente	Excelente	Excelente	Regular
Curva de Aprendizaje	Baja	Media	Media	Media
Rendimiento	Muy Alto	Alto	Alto	Regular
Extensiones y Plugins	Amplia disponibilidad	Específico para Python	Amplia disponibilidad	Limitada

**Resultado:** *Visual Studio Code*, se selecciona debido a su flexibilidad, extensibilidad y compatibilidad con una gran cantidad de tecnologías.

## Frameworks para Backend

Criterio	NestJS	Express.js	Koa.js	Fastify
<b>Arquitectura</b>	Modular, basada en capas	Minimalista	Modular	Performance
<b>Características Principales</b>	Basado en decorators, soporte para GraphQL y WebSockets, inyección de dependencias, soporte para microservicios.	Framework ligero, flexibilidad para definir middleware y rutas, soporte extensivo de paquetes.	Mejor manejo de middlewares, basado en Promesas, sin middleware incluido por defecto.	Enfoque en velocidad, manejo eficiente de JSON, esquema de validación.
<b>Popularidad</b>	Muy Alta	Muy Alta	Alta	Alta
<b>Documentación y comunidad</b>	Excelente, activa y detallada	Muy activa, muchos recursos	Activa pero más pequeña que Express	Activa, creciendo rápidamente
<b>Casos de Uso Comunes</b>	API REST, Microservicios, Backend robusto	API REST, Aplicaciones pequeñas/medianas	Aplicaciones pequeñas, enfoque personalizado	APIs de alta velocidad y escalabilidad

**Resultado:** *NestJS*, se selecciona por sus destacadas características para proyectos grandes y escalables debido a su arquitectura modular y soporte integrado para múltiples tecnologías. Su soporte nativo para conceptos avanzados como inyección de dependencias, microservicios y GraphQL, junto con una documentación robusta y una comunidad activa, lo hacen ideal para proyectos empresariales y de largo plazo que requieren robustez y flexibilidad, que es el tipo de proyecto que se desea realizar

## Frameworks para Frontend

Criterio	React	Angular	Vue.js	Svelte
<b>Facilidad de Uso</b>	Alta	Media	Alta	Alta
<b>Ecosistema</b>	Amplio	Amplio	Amplio	Medio
<b>Escalabilidad</b>	Muy Alta	Muy Alta	Alta	Media
<b>Comunidad</b>	Muy Amplia	Amplia	Amplia	En crecimiento

**Resultado:** *React*, se selecciona por su flexibilidad, compatibilidad con librerías externas y gran comunidad.

### Sistemas de Gestión de Bases de Datos

Criterio	PostgreSQL	MySQL	MongoDB	SQLite
Modelo de Datos	Relacional	Relacional	NoSQL (Documentos)	Relacional
Escalabilidad	Muy Alta	Alta	Muy Alta	Baja
Seguridad	Muy Alta	Alta	Alta	Media
Comunidad y Soporte	Muy Amplia	Muy Amplia	Amplia	Amplia

**Resultado:** *PostgreSQL*, se selecciona por su capacidad de manejar grandes volúmenes de datos y funcionalidades avanzadas como soporte JSON.