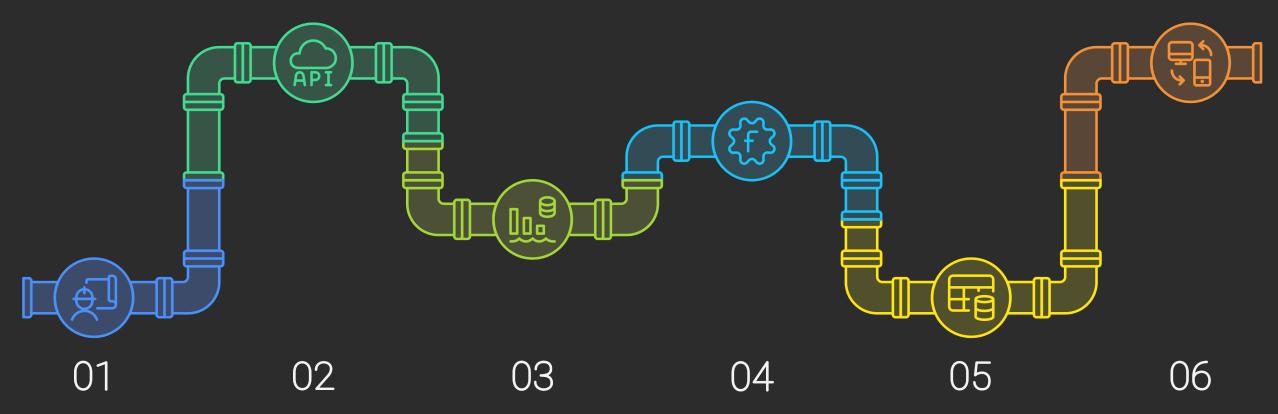
## Proyecto Final

# Análisis de Rendimiento de Aplicaciones Web través de una API en Python

David Camilo García Echavarría Cristian David Tamayo Espinosa Silvio José Otero Guzmán Gaia Ramirez Hincapié

### Resumen de API para Monitorizar el Rendimiento de Aplicaciones Web



# Proponer el Proyecto

El proyecto se propone para desarrollar una API.

### Desarrollar la API

La API se desarrolla usando Python.

### **Medir Métricas**

Las métricas de CPU, memoria y E/S se miden.

#### **Crear Perfiles**

Se crean perfiles detallados de funciones específicas.

### **Exponer Datos**

Los datos se exponen a través de endpoints REST.

### Garantizar Compatibilidad

Se garantiza la compatibilidad multiplataforma.

# Necesidad: Optimización del rendimiento de aplicaciones web

# Aplicación web no monitoreada

Cuellos de botella de recursos e inestabilidad

### Implementar métricas en tiempo real

Rastrear indicadores clave de rendimiento

# Perfilar funciones críticas

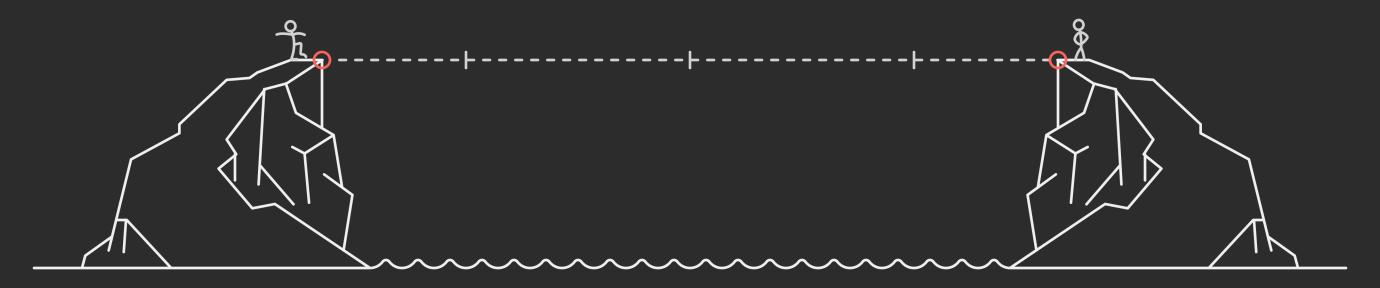
Identificar código intensivo en recursos

# Optimizar cuellos de botella

Mejorar la eficiencia y escalabilidad

# Aplicación web optimizada

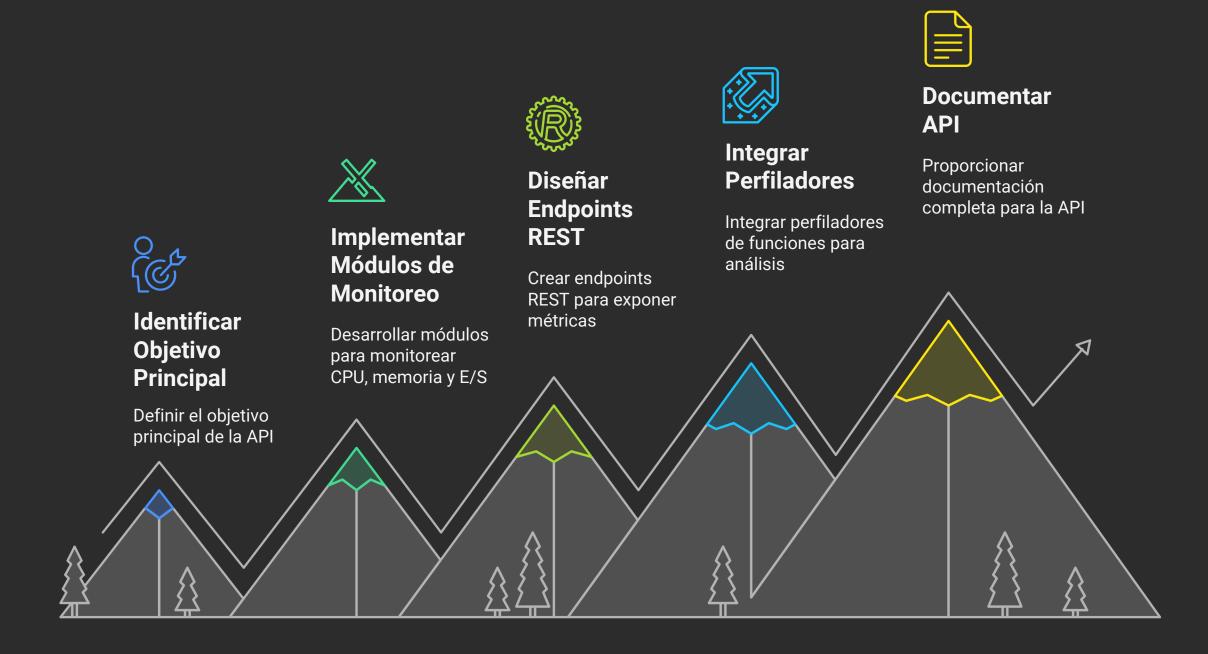
Estable, escalable y eficiente



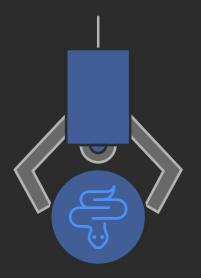
### Relación con el curso de Sistemas Operativos



### **Objetivos API de Monitoreo de Rendimiento**

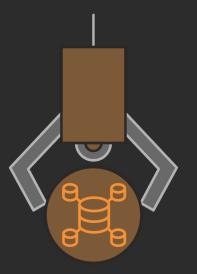


### Herramientas de Desarrollo



## Lenguaje Principal

Versión de Python 3.8 o superior



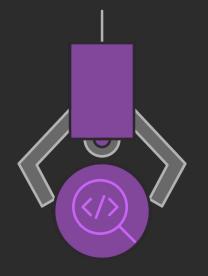
Capa REST

FastAPI o Flask para API REST



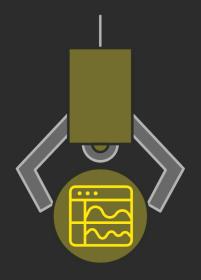
### Métricas del Sistema

Biblioteca psutil para métricas del sistema



### Herramientas de Perfilado

cProfile, memory\_profiler, py-spy



### Herramienta de Visualización

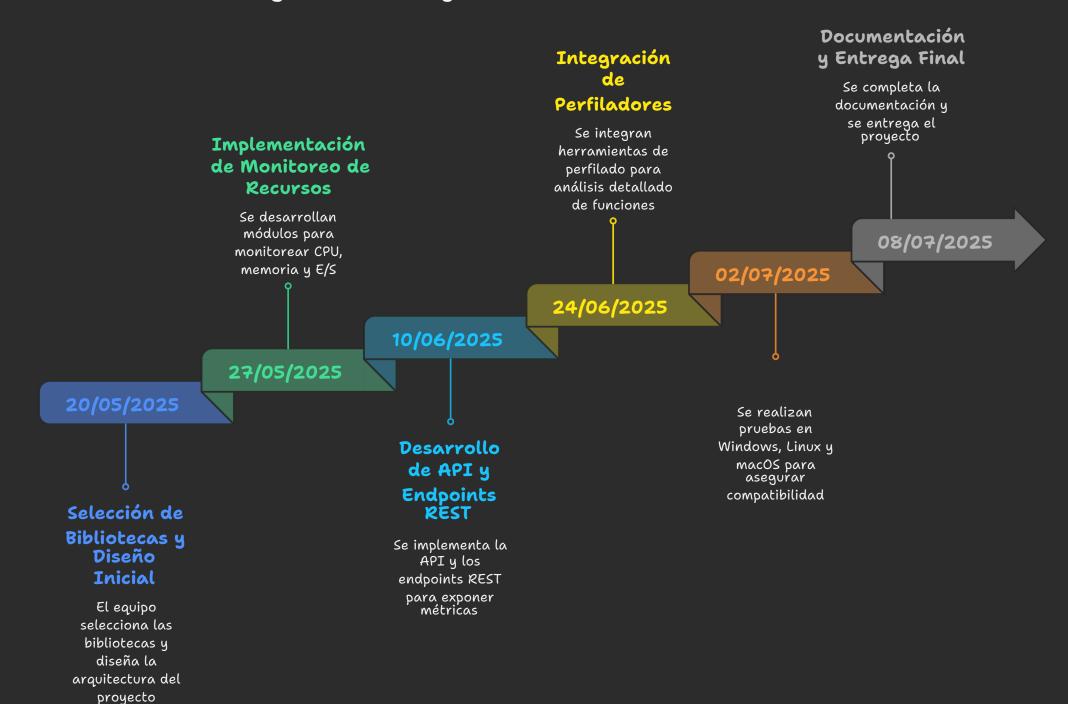
Cliente de Prometheus para visualización de datos

#### Actividades principales



Regulizar y leványes tigación exhaustiva y seleccionar

#### Cronograma del Proyecto de Análisis de Rendimiento Web



## Referencias

- Repositorio oficial de psutil: https://github.com/giampaolo/psutil
- Documentación de FastAPI: https://fastapi.tiangolo.com/
- Python memory-profiler. (n.d.). memory\_profiler: Monitor Memory Usage of Python Code. Recuperado el 22 de mayo de 2025, de https://pypi.org/project/memory-profiler/
- Hettinger, R. (n.d.). Profiling and Timing Code. Python Software Foundation. Recuperado el 22 de mayo de 2025, de https://docs.python.org/3/library/profile.html

# Gracias!