



## **Propuesta proyecto final**

**Tema: Creación de una API para analizar el  
rendimiento de aplicaciones.**

Jaime Andrés Muñoz

Estefania Goez Moreno

Facultad de Ingeniería

Universidad de Antioquia

Sistemas Operativos y

laboratorio

Henry Alberto Arcila Ramirez

Medellín, 2024

## Resumen

Hemos decidido desarrollar una API para analizar el rendimiento de aplicaciones con el fin de ofrecer a los desarrolladores una herramienta accesible, eficiente y fácil de usar. Esta API recopilará y mostrará métricas clave como el uso de recursos (CPU, memoria, almacenamiento), tiempo de respuesta, y posibles cuellos de botella en la ejecución. Nuestra solución estará dirigida tanto a desarrolladores expertos como a principiantes, con una interfaz amigable que evite el uso de comandos complejos y se pueda integrar fácilmente con cualquier entorno de desarrollo.

## Introducción

El rendimiento de las aplicaciones es un problema recurrente en el desarrollo de software moderno. Un mal rendimiento puede generar frustración entre los usuarios, pérdida de clientes, e incluso costos adicionales en infraestructura. Actualmente, la mayoría de las herramientas disponibles para monitorear el rendimiento son técnicas y requieren un conocimiento avanzado en la línea de comandos, lo que puede dificultar su uso para aquellos sin experiencia en sistemas operativos o monitoreo avanzado.

Con la creciente demanda en la industria de TI de aplicaciones que ofrezcan experiencias rápidas y fluidas, creemos que es fundamental desarrollar una API que simplifique la tarea de monitoreo. Nuestra API está pensada para desarrolladores de cualquier nivel de habilidad, proporcionando información detallada en tiempo real a través de una interfaz sencilla y accesible.

## **Antecedentes**

Para abordar este proyecto, nos basamos en conceptos teóricos clave de sistemas operativos, como la gestión de recursos, el monitoreo de procesos y la optimización del rendimiento. Algunos de los temas fundamentales que abordaremos incluyen:

Aprenderemos a utilizar llamadas al sistema que nos permitan acceder a estadísticas sobre el rendimiento de las aplicaciones, como el uso de la CPU y la memoria. Esto se relaciona con lo estudiado en la gestión de procesos en los sistemas operativos.

Al analizar el rendimiento, es esencial considerar cómo las aplicaciones interactúan con el sistema de almacenamiento. Esto nos permite identificar cuellos de botella relacionados con la lectura/escritura de datos.

Utilizaremos técnicas de optimización vistas en clase para mejorar la eficiencia en el uso de recursos del sistema. Estas teorías se conectan directamente con lo aprendido en el curso de Sistemas Operativos y su laboratorio, ya que el monitoreo eficiente de recursos es clave en el desarrollo y mantenimiento de aplicaciones robustas.

## Objetivos

### **Objetivo principal:**

Desarrollar una API que permita analizar el rendimiento de aplicaciones en tiempo real, mostrando información accesible y comprensible para usuarios de todos los niveles.

### **Objetivos específicos:**

1. Crear una interfaz amigable que no requiera el uso de comandos complejos.
2. Implementar funcionalidades que permitan el monitoreo de uso de CPU, memoria, y almacenamiento.
3. Incluir herramientas para identificar cuellos de botella en la ejecución de las aplicaciones.

## Metodología

Para desarrollar esta API, utilizaremos tecnologías y herramientas que nos permitan cumplir con nuestros objetivos de manera eficiente. Entre las herramientas principales, consideramos:

En cuanto al lenguaje de programación utilizaremos **JavaScript** junto con Express para el desarrollo de la API, **Express** nos permitirá estructurar y manejar las rutas y solicitudes HTTP de manera sencilla.

Utilizaremos **PostgreSQL**, que es una base de datos relacional robusta, ideal para almacenar los registros históricos de rendimiento y manejar grandes volúmenes de datos. Spring Boot, lo utilizaremos únicamente, si lo vemos necesario.

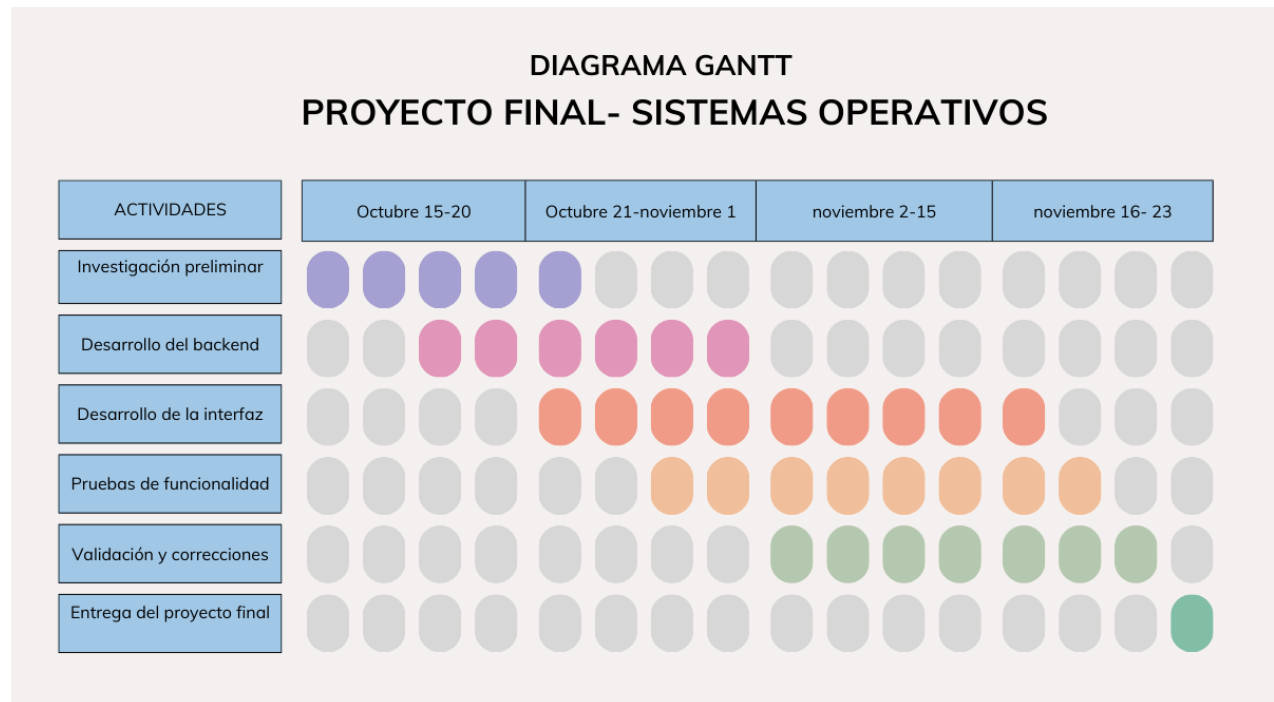
## **Diseño de experimentos**

Para validar la funcionalidad y el rendimiento de la API, realizaremos pruebas controladas que nos permitan medir su efectividad

Simularemos diferentes niveles de uso de recursos en varias aplicaciones para verificar que la API pueda manejar el monitoreo de manera efectiva sin sobrecargar el sistema.

Mediremos el tiempo que tarda la API en procesar y devolver los datos de rendimiento, para asegurarnos de que sea lo más eficiente posible.

Probaremos la API en diferentes sistemas operativos (Linux, Windows) para asegurarnos de que sea compatible.





## **Referencias**

Bullock, J. (2022). Mastering API Architecture: A Comprehensive Guide to Building Scalable APIs. TechPress. <https://doi.org/10.1002/api.architecture>

Roberts, S. (2020). Real-Time Data Processing: Best Practices and Techniques. TechJournal. <https://doi.org/10.1016/j.realtime.2020.01.003>

Singh, A. (2019). Optimizing Application Performance: Tools and Techniques. Journal of Software Development, 45(2), 67-89. <https://doi.org/10.1016/j.jsd.2019.02.007>