

Proyecto Final

Curso de Sistemas Operativos y Laboratorio

Implementación de un monitor de sistema desde cero.

Jonathan David Fernandez Vargas

Valeria Alvarez Fernandez

Santiago Arenas Gomez

Resumen

El presente proyecto tiene como objetivo diseñar e implementar un monitor de sistema desde cero, capaz de mostrar en tiempo real información clave del funcionamiento del sistema operativo, como el uso de CPU, memoria RAM, procesos activos y almacenamiento. El desarrollo se realizará en un entorno tipo consola, utilizando lenguaje C y acceso directo a los archivos del sistema `/proc` en sistemas Linux. Este proyecto busca reforzar el entendimiento de los conceptos de Sistemas Operativos mediante una aplicación práctica y funcional.

Introducción

En los sistemas operativos modernos, el monitoreo del rendimiento es esencial para detectar cuellos de botella, gestionar recursos y garantizar el buen funcionamiento del sistema. Este proyecto propone construir una herramienta de monitoreo desde cero, sin el uso de bibliotecas avanzadas, permitiendo así explorar en profundidad las estructuras internas del sistema operativo.

El principal problema abordado es la falta de comprensión práctica del funcionamiento interno del sistema operativo, especialmente en la gestión de recursos como CPU, memoria y procesos. Al implementar un monitor desde cero, se soluciona esta necesidad mediante el diseño de una herramienta que permite observar directamente la actividad del sistema.

Hoy en día, el rendimiento de sistemas informáticos es crítico en múltiples áreas: desde servidores hasta dispositivos móviles. Entender cómo monitorear y gestionar los recursos del sistema es clave para optimizar aplicaciones, detectar fallos, y tomar decisiones informadas en la administración de sistemas. Esta habilidad es altamente valorada en entornos técnicos y laborales.

Antecedentes o marco teórico

Temas clave:

- Gestión de procesos y planificación del CPU.
- Administración de memoria.
- Sistema de archivos por ejemplo `/proc` en Linux.
- Llamadas al sistema (`syscalls`).
- Herramientas tradicionales de monitoreo (`top`, `htop`, `vmstat`, `free`, `ps`).

Relación con el curso:

- **Parte teórica:** Conceptos como gestión de procesos, memoria, planificación de CPU, sistemas de archivos virtuales.
- **Laboratorio:** Uso de herramientas del sistema, programación a bajo nivel, lectura de archivos del sistema y uso de llamadas al sistema.

Objetivos (principal y específicos)

Objetivo principal:

Desarrollar un monitor de sistema desde cero que permita visualizar información clave del estado del sistema operativo en tiempo real.

Objetivos específicos:

- Mostrar el uso de CPU de forma periódica.
- Mostrar el uso de memoria RAM y espacio de intercambio.
- Listar procesos activos con información relevante (PID, nombre, estado).
- Mostrar el estado del almacenamiento (espacio usado/disponible).
- (Opcional) Mostrar estadísticas de red.
- Implementar una interfaz en consola para visualizar estos datos dinámicamente.

Metodología

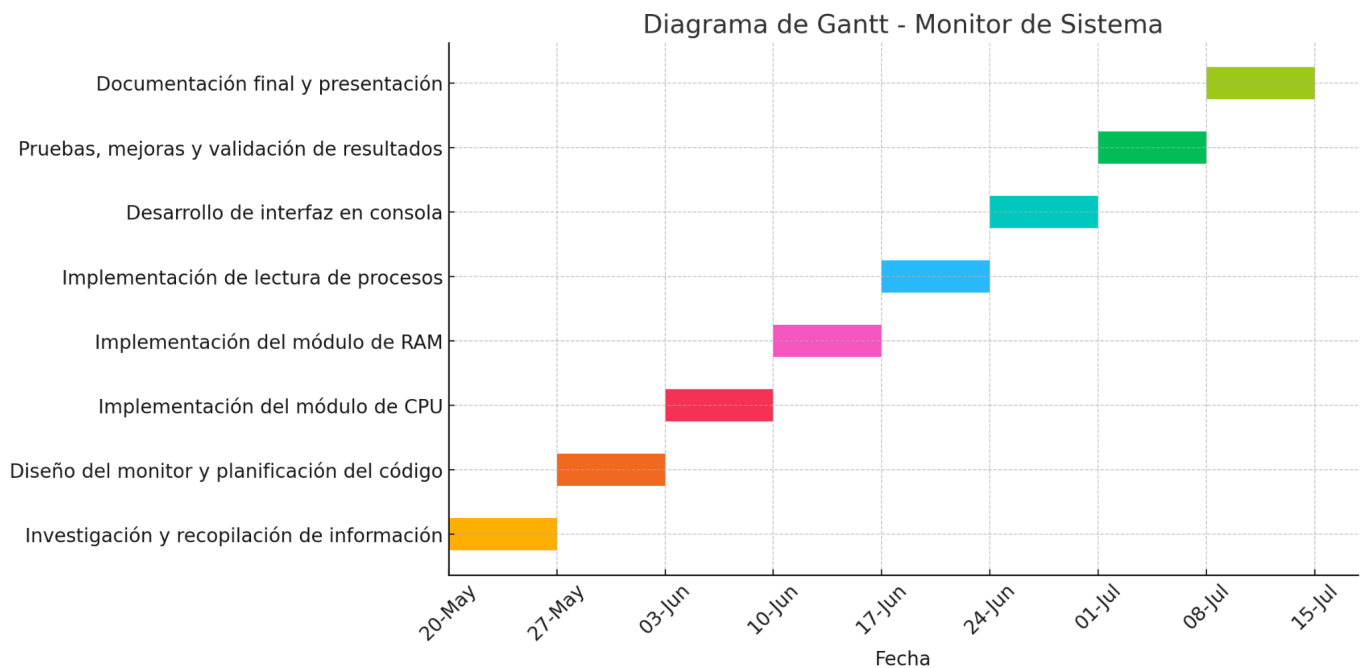
Herramientas y tecnologías:

- **Lenguaje de programación:** C (alternativamente Python si se permite).
- **Sistema operativo:** Preferiblemente Linux (por el uso del sistema de archivos `/proc`).
- **Editor/IDE:** VS Code, Vim o similar.
- **Control de versiones:** Git (opcional).
- **Formato de salida:** Interfaz en consola tipo `top`.

Actividades y pasos:

1. **Revisión bibliográfica y técnica** sobre `/proc`, estructuras del sistema y llamadas al sistema.
2. **Diseño de la estructura del monitor** (qué información mostrar y cómo).
3. **Lectura y procesamiento de datos** del sistema (CPU, RAM, procesos).
4. **Desarrollo de la interfaz de consola**.
5. **Pruebas y validación** en distintos estados del sistema.
6. **Documentación del proyecto y conclusiones**.

Cronograma



Referencias

Silberschatz, A., Galvin, P. B., & Gagne, G. (2020). *Operating System Concepts* (10th ed.). Wiley.

Love, R. (2010). *Linux System Programming*. O'Reilly Media.

Linux man pages: [proc](#), [ps](#), [top](#), [sysconf](#).

Documentación oficial de GNU/Linux sobre [/proc](#):
<https://man7.org/linux/man-pages/man5/proc.5.html>

Ejemplos y tutoriales de sistemas de monitoreo en C.