



# SISTEMAS OPERATIVOS

## **ACTIVIDAD DE APRENDIZAJE 4 PROGRAMA 2**

**Profesor:**

**VIOLETA DEL ROCIO BECERRA VELAZQUEZ**

VERDUZCO ROSALES LUIS ENRIQUE

223992388

Ingeniería en Computación

14/09/25

CUCEI DIVTIC D04

## ÍNDICE

Objetivo de la actividad.....	2
Solución de problemas.....	2
Soluciones.....	2
Conclusión.....	3

---

## Objetivo de la actividad

El objetivo de esta actividad fue desarrollar una simulación de **gestión de procesos** utilizando JavaScript para comprender cómo un sistema operativo organiza y ejecuta múltiples tareas. La meta era visualizar conceptos teóricos como el estado de los procesos (en espera, en ejecución, terminados), la planificación por lotes y la gestión del tiempo de ejecución (TME y TT) de manera fácil e interactiva.

## Solución de problemas

El proyecto fue hecho con JavaScript, lo cual es ideal para este tipo de simulaciones en el navegador. La mayor parte del trabajo se centró en la manipulación del DOM (Document Object Model). Usamos funciones como `getElementById` y propiedades como `.innerHTML` o `.textContent` para conectar la función principal del programa (la lógica de la simulación) con la interfaz gráfica. El código también usa `addEventListener` para manejar eventos, lo que nos permite interactuar con el DOM a través de clics y pulsaciones de teclado, lo cual es algo muy necesario para nuestro software.

Para que la simulación avance en el tiempo, se empleó `setInterval`. Esta función crea un bucle que se ejecuta cada 1000 milisegundos (1 segundo), simulando el paso del tiempo en el reloj global del sistema.

## Soluciones

- **Bucle de Simulación:**

La función `tick` es el contador del programa. Cada segundo, verifica el estado actual: si el procesador está libre, toma el siguiente proceso del lote de espera; si ya hay un proceso en ejecución, le incrementa el tiempo transcurrido; si el proceso termina, lo mueve a la lista de terminados.

El mayor desafío en el diseño de la función `tick` fue manejar la lógica de los cambios de estado. Era complicado hacer que un proceso se moviera correctamente del lote de espera a la ejecución sólo cuando el procesador estaba libre, y que luego, una vez completado, se pasara al siguiente lote sin interrupciones.

Esto requería múltiples condicionales anidados, lo cual hacía más complicado el entender el código. Además, tuve que asegurarme de que la función no intentará acceder a elementos de lotes o procesos que ya no existían, lo que generaba errores.

Para solucionar estos problemas tuve que repasar la lógica, haciendo que cada "tick" revisara de forma ordenada si había un proceso en ejecución o si

era necesario mover uno nuevo, y si el lote actual se había vaciado, para poder avanzar al siguiente.

## Conclusión

Este proyecto fue muy útil para entender conceptos abstractos de los sistemas operativos. Me ayudó a entender que la gestión de procesos no es lineal, sino que implica un ciclo constante de planificación, ejecución y manejo de interrupciones. La simulación me permitió ver en tiempo real cómo los procesos pasan de un estado a otro y cómo el procesador se ocupa de cada uno. La flexibilidad de JavaScript para manejar el tiempo y manipular el DOM hizo posible crear un modelo funcional y claro.

**Código Fuente:** <https://github.com/UnluckyLuren/SisOperativos.git>

**Ejecutable:** <https://unluckyluren.github.io/SisOperativos/>