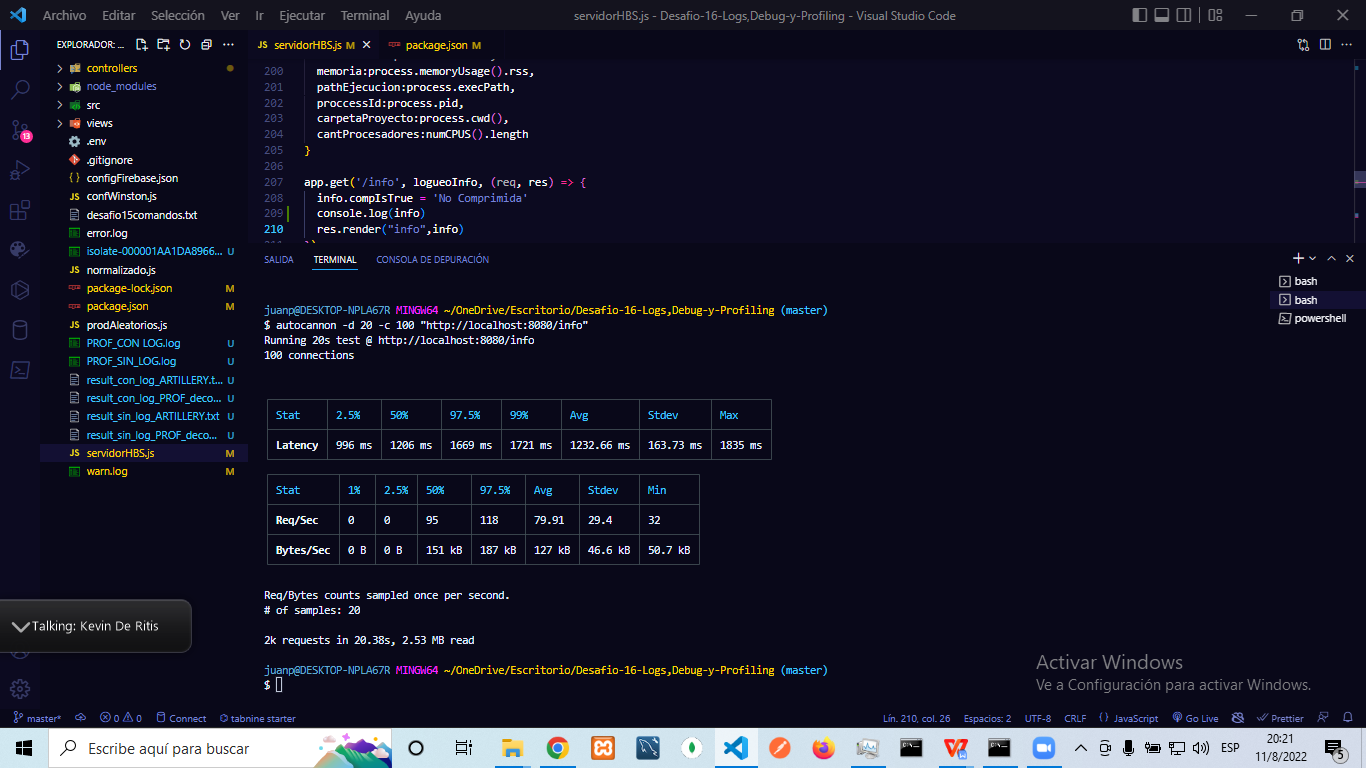
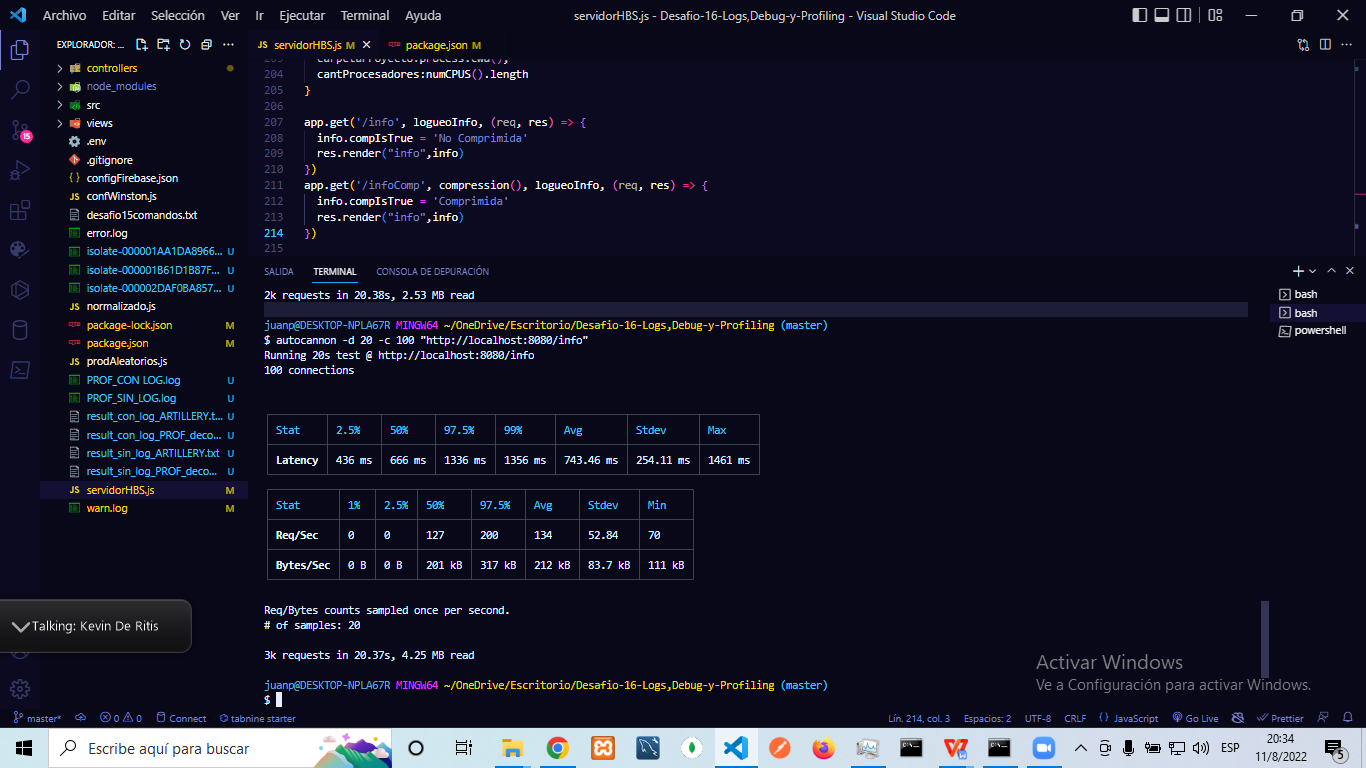
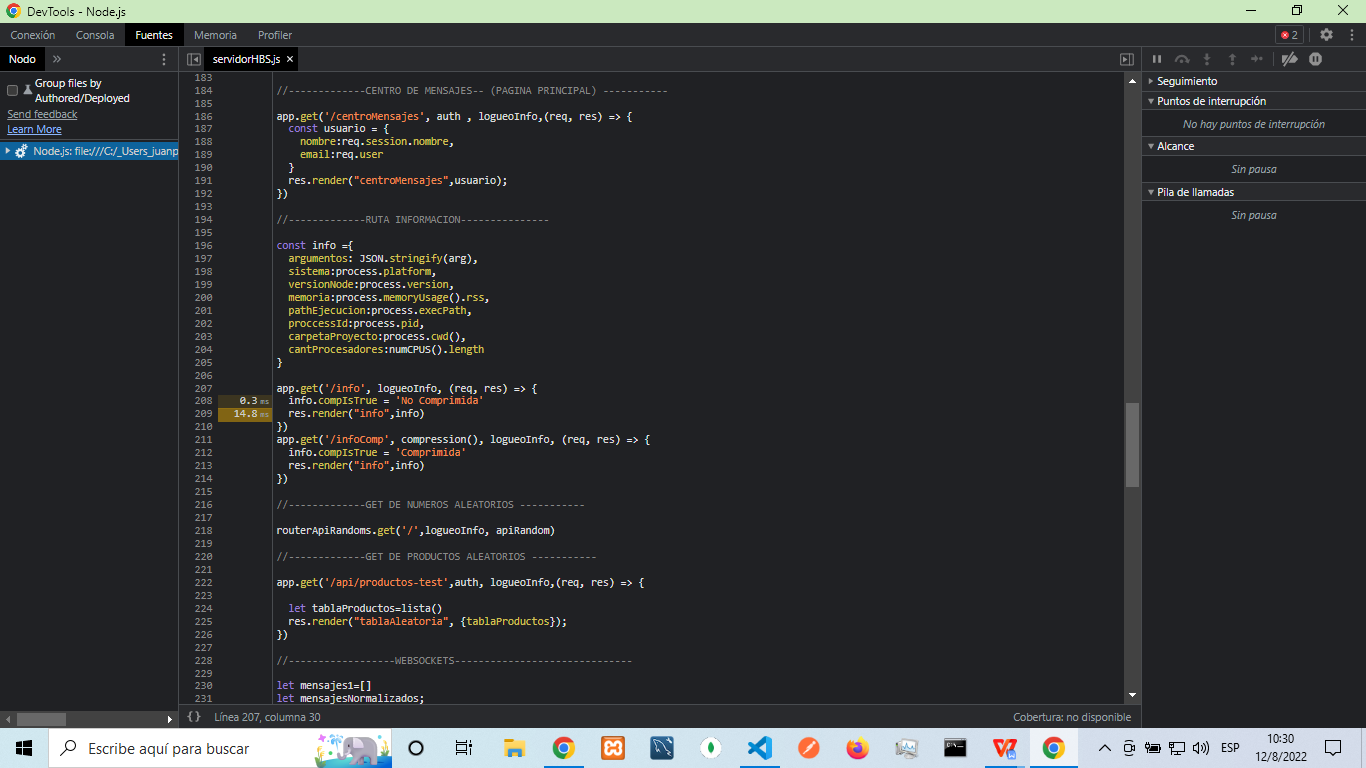
**AUTOCANNON CON LOG**



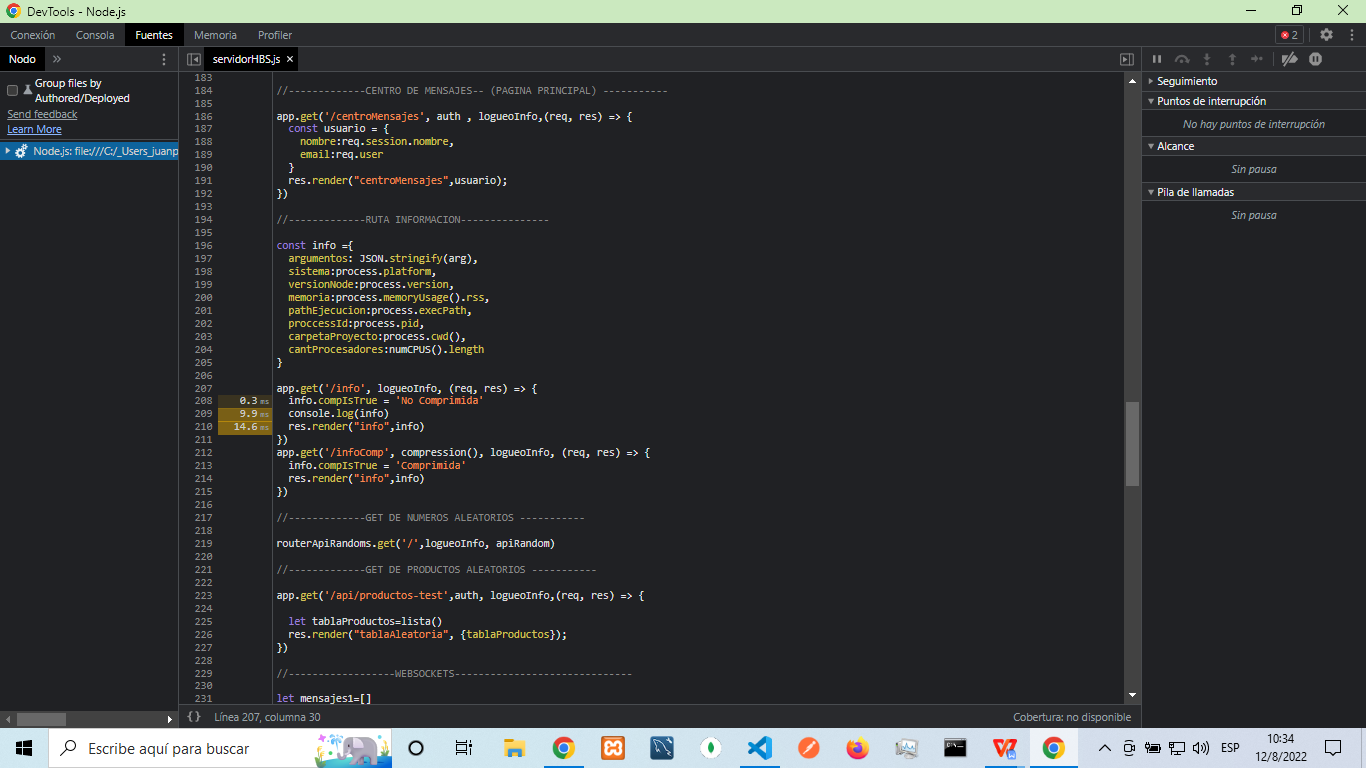
**AUTOCANNON SIN LOG**



**DATOS RUTA /info SIN console.log EN INSPECT**

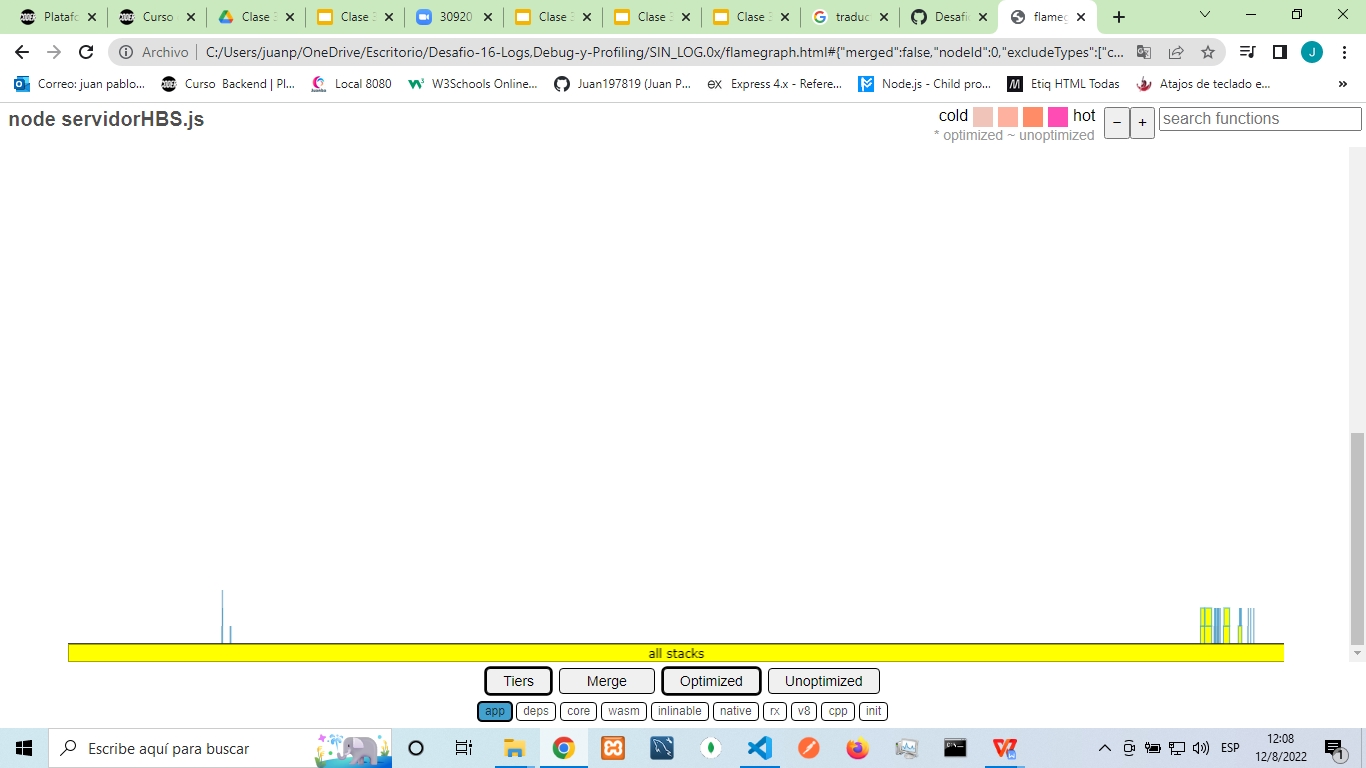
****

**DATOS RUTA /info con console.log EN INSPECT**

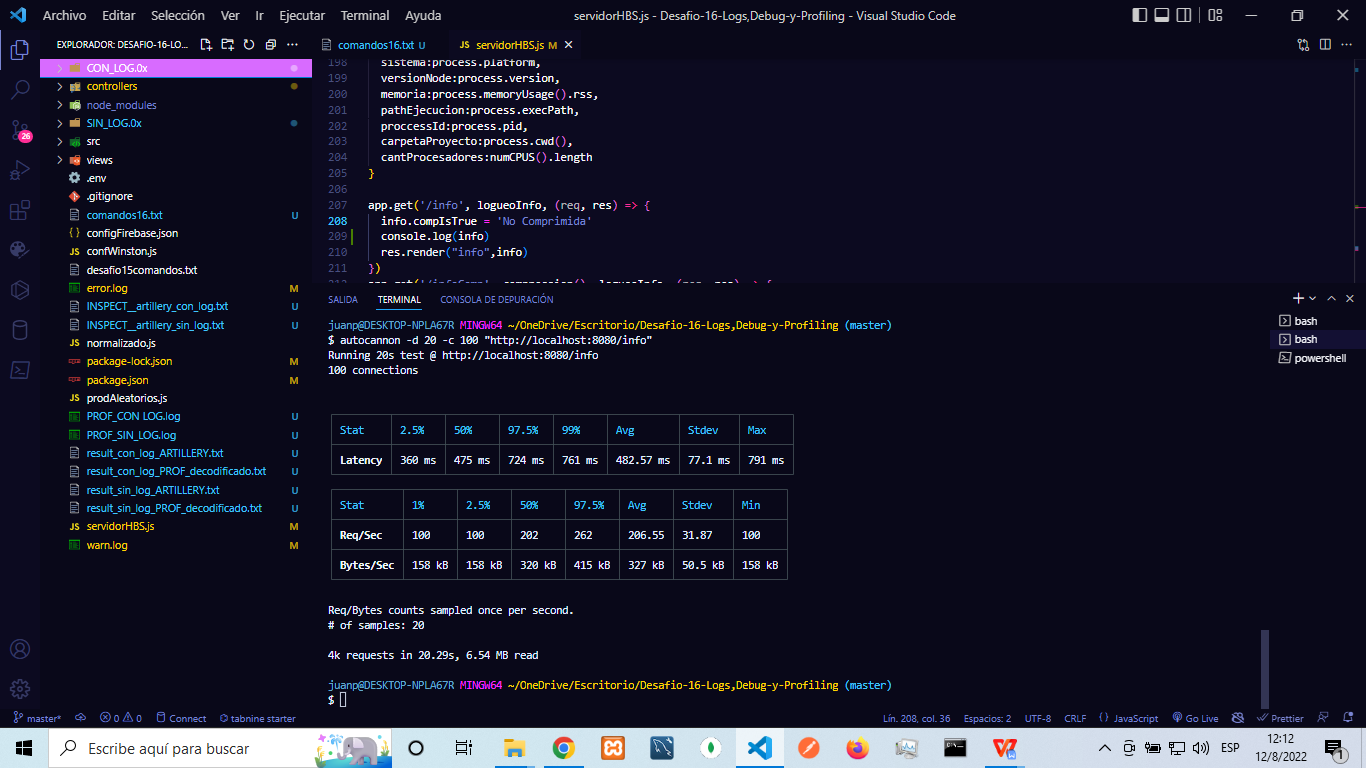
****

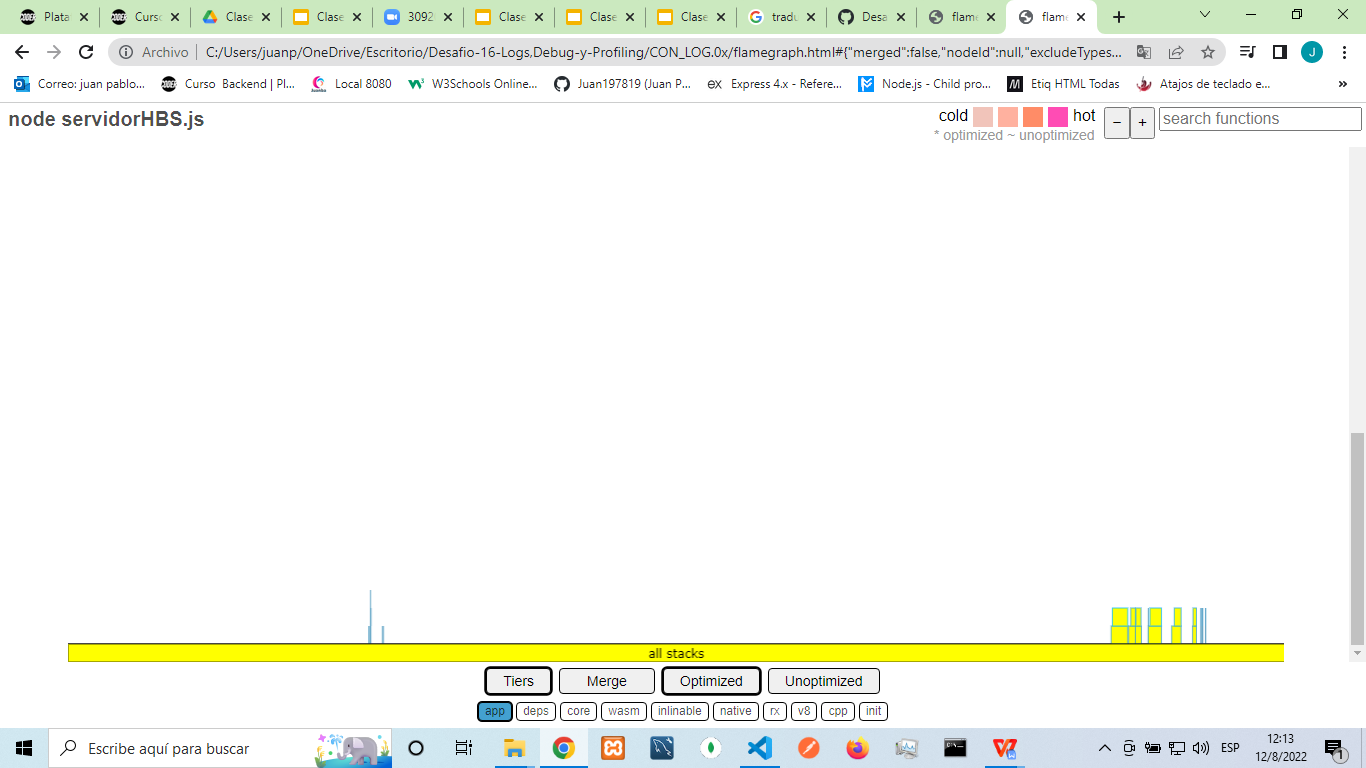
**DIAGRAMA DE FLAMA 0x SIN CONSOLE.LOG y AUTOCANNON**

****

****

**DIAGRAMA DE FLAMA 0x CON CONSOLE.LOG y AUTOCANNON**

****



**CONCLUSIONES:**

En todos los análisis se comprueba que tener un proceso bloqueante como el Console.log en una ruta va a desmejorar enormemente en su rendimiento a la app con respecto a no tenerlo o a tenerlo pero de manera no bloqueante por ejemplo usando asincronismo.

Usando autocannon podemos ver que en igual cantidad de tiempo (20 seg) en el test que tiene el console.log se llegaron a realizar unos 2000 request contra los 3000 que se lograron con el que no lo tenia, además que se duplican casi todos los tiempos de latencia.

En los informes de artillery de las diferentes pruebas se observa que el tiempo medio de respuestas siempre es mas alta con el el proceso bloqueante y en el caso del profiling de Node vemos una gran diferencia a favor de Ticks por parte del proceso bloqueante.

En el caso del grafico de flama, si bien no es tan nítida la diferencia, se puede observar como el grafico bloqueante permanece muchísimo mas tiempo en el stack con respecto al del proceso no bloqueante. (Todos los archivos se encuentran dentro del proyecto junto a la lista de comandos para cada consigna).