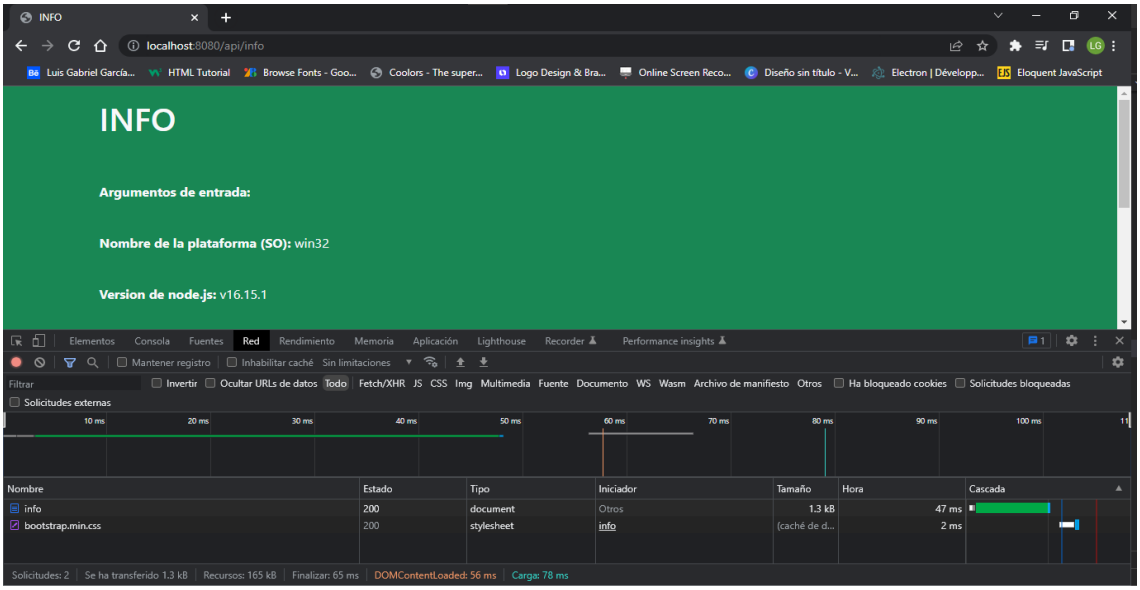
Desafío Clase 32:

Análisis de Performance

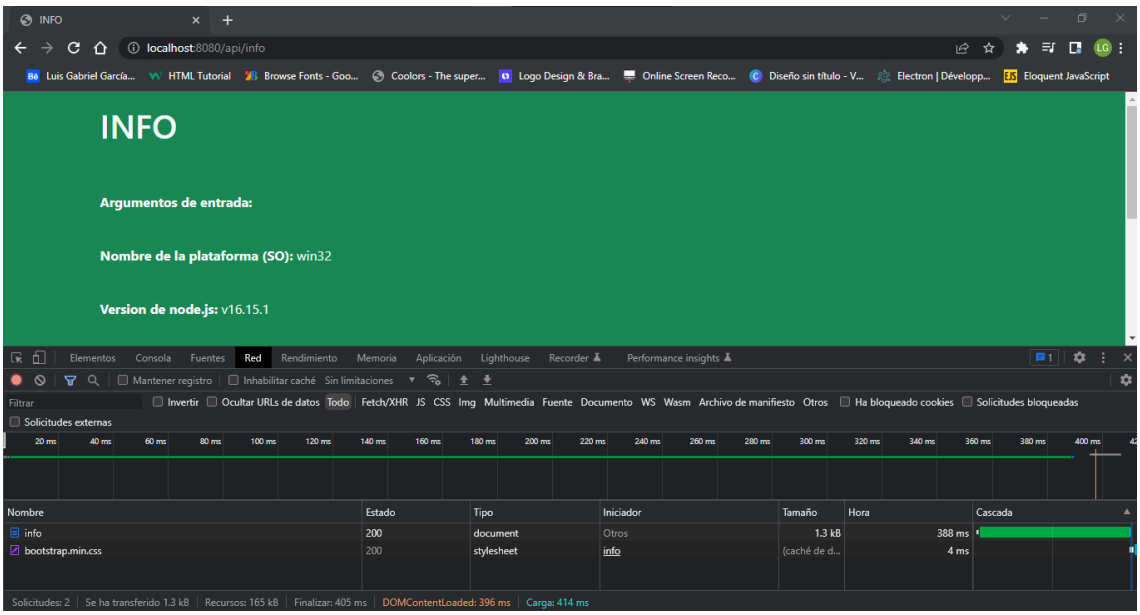
1. Uso de GZIP

Sobre la ruta api/info no se evidencias diferencias, en ambos casos (con y sin Compression) el tamaño de los datos es de 1.3kb

Sin Compression:

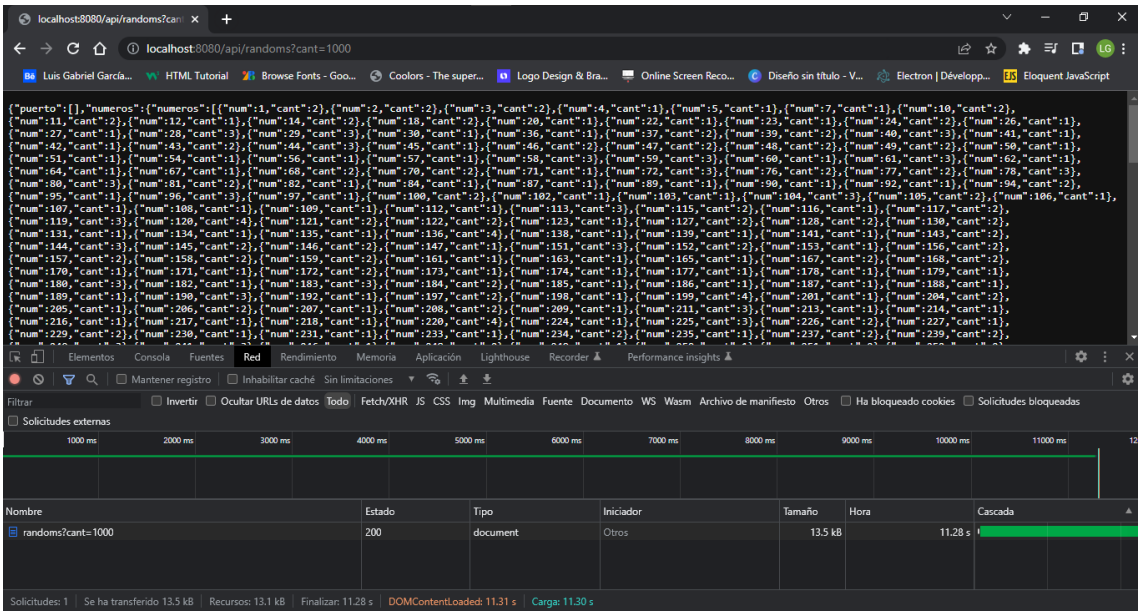


Con Compression:

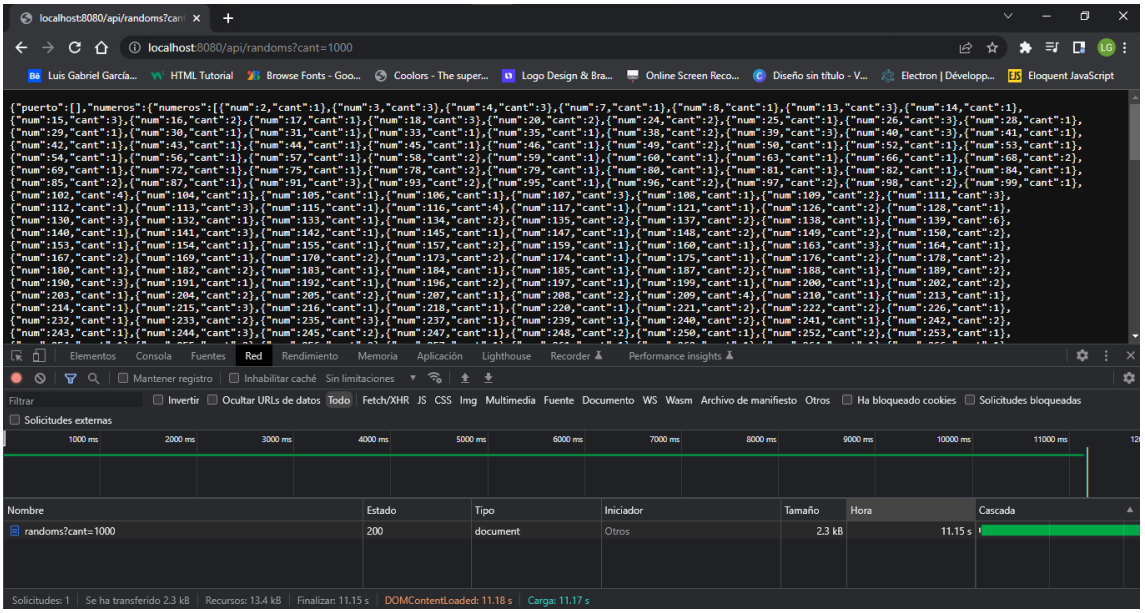


Se prueba sobre la ruta api/randoms y ahí se ve una diferencia importante:

- **Sin Compression:** 13.5kb



- **Con Compression:** 2.3kb



2. Perfilamiento del Servidor con Node:

Se realizaron pruebas de carga con Artillery emulando 50 conexiones concurrente con 20 requests cada una y con Autocannon emulando 100 conexiones concurrentes durante 20 segundos. En ambos casos se probaron con y sin console.log sobre los datos que devuelve la ruta /info.

Los resultados fueron los siguientes:

- **Perfilamiento Node con Console.log :**
  resultado_node_profiling_artillery_conConsoleLog.txt
  resultado_node_profiling_autocannon_conConsoleLog.txt

- **Perfilamiento Node sin Console.log :**
  resultado_node_profiling_artillery_sinConsoleLog.txt
  resultado_node_profiling_autocannon_sinConsoleLog.txt

Se observa una menor cantidad de Ticks en Node al probar sobre la ruta /info al evitar realizar console.log de los datos que luego se renderiza

- **Artillery con Console.log:**
  result_artillery_info_conConsoleLog.txt

- **Artillery sin Console.log:**
  result_artillery_info_sinConsoleLog.txt

Se observan tiempos de respuesta promedio menores y mayor cantidad de requests/seg en las ejecuciones sin Console.Log

- **Autocannon con Console.log:**

```
> autocannon -c 100 -d 20 http://localhost:8080/info

Running 20s test @ http://localhost:8080/info
100 connections
```

Stat	2.5%	50%	97.5%	99%	Avg	Stdev	Max
Latency	836 ms	899 ms	1868 ms	1869 ms	947.25 ms	210.25 ms	1876 ms

Stat	1%	2.5%	50%	97.5%	Avg	Stdev	Min
Req/Sec	0	0	100	200	104.95	38.29	100
Bytes/Sec	0 B	0 B	132 kB	264 kB	139 kB	50.5 kB	132 kB

```
Req/Bytes counts sampled once per second.
# of samples: 20

2k requests in 20.13s, 2.77 MB read
```

- **Autocannon sin Console.log:**

```
> autocannon -c 100 -d 20 http://localhost:8080/info

Running 20s test @ http://localhost:8080/info
100 connections
```

Stat	2.5%	50%	97.5%	99%	Avg	Stdev	Max
Latency	590 ms	669 ms	1866 ms	1874 ms	727.33 ms	238.33 ms	1884 ms

Stat	1%	2.5%	50%	97.5%	Avg	Stdev	Min
Req/Sec	0	0	100	200	135	57.23	100
Bytes/Sec	0 B	0 B	132 kB	264 kB	178 kB	75.5 kB	132 kB

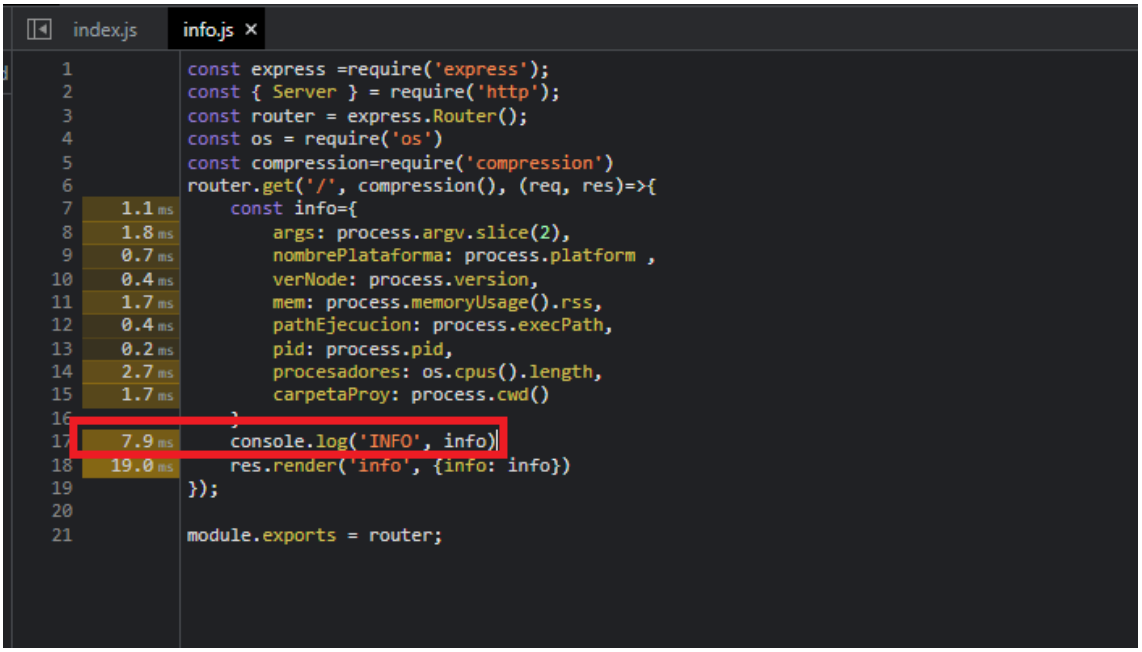
```
Req/Bytes counts sampled once per second.
# of samples: 20

3k requests in 20.16s, 3.56 MB read
```

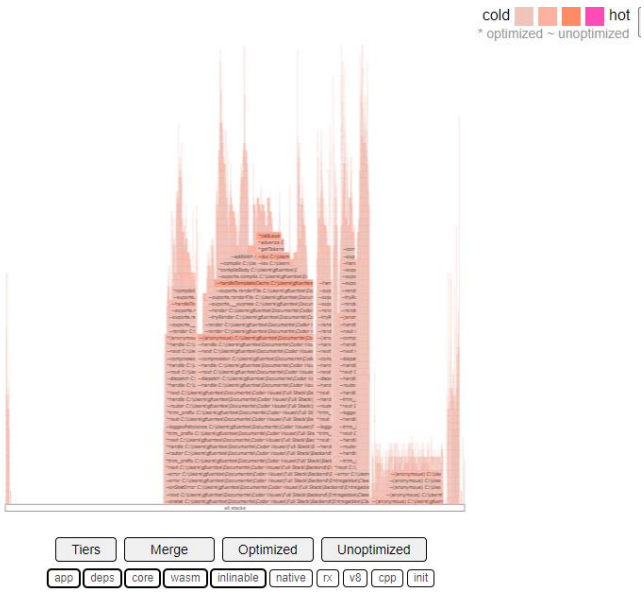
En este caso también se observan mayor cantidad de requests por segundo, mayor tasa de datos enviados y menor latencia para los casos sin el uso de Console.log

- **Inspector:**

Se observa que el la línea de console.log agrega 7.9ms de tiempo de ejecución



- **Diagrama de flama con 0x con Console.log:**



- **Diagrama de flama con 0x sin Console.log:**