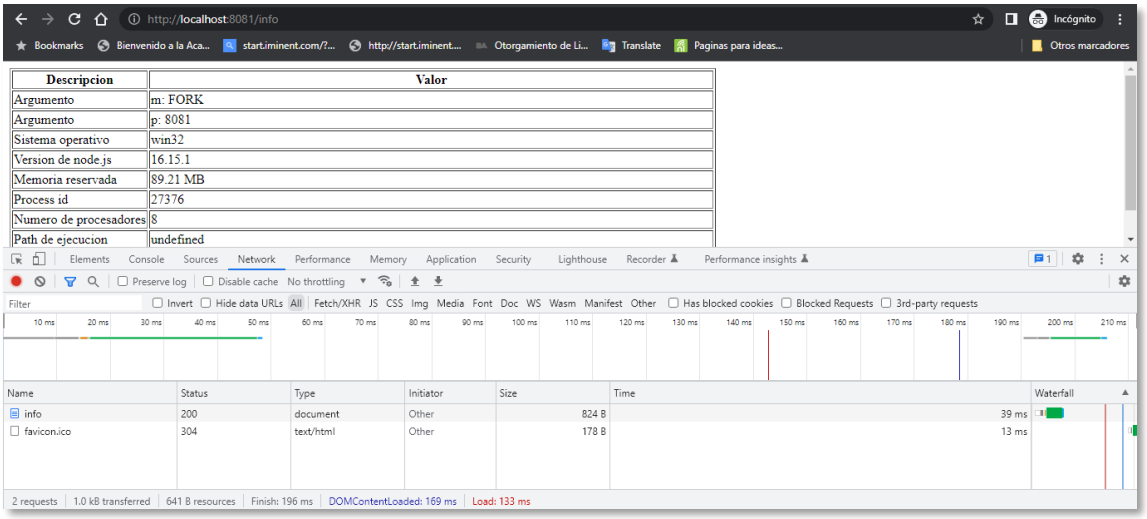
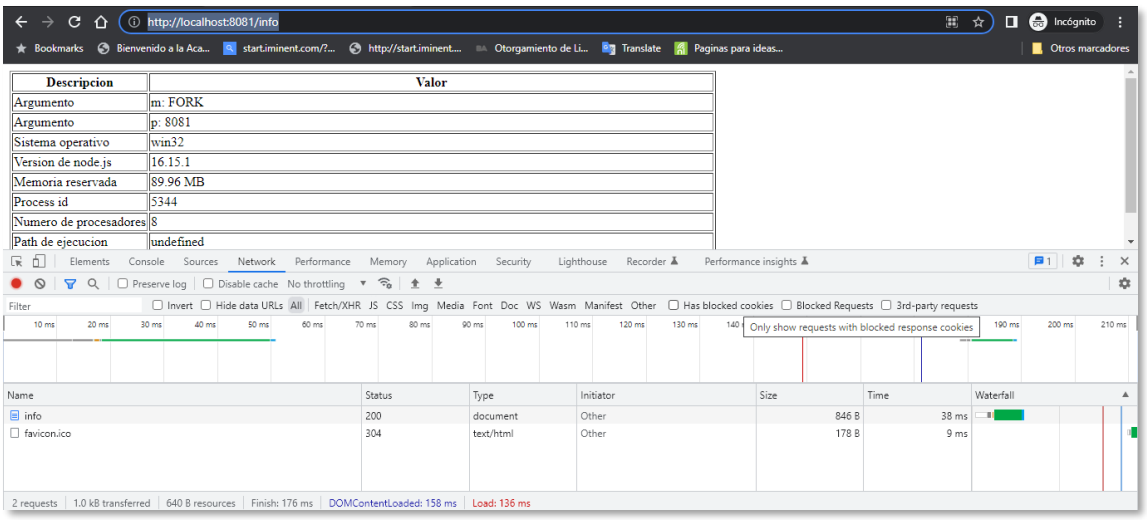
Loggers, Gzip, y Análisis de Performance.

Ruta /info sin compresión.



Descripcion	Valor
Argumento	m: FORK
Argumento	p: 8081
Sistema operativo	win32
Version de node.js	16.15.1
Memoria reservada	89.21 MB
Process id	27376
Numero de procesadores	8
Path de ejecucion	undefined

Ruta /info con compresión.



Descripcion	Valor
Argumento	m: FORK
Argumento	p: 8081
Sistema operativo	win32
Version de node.js	16.15.1
Memoria reservada	89.96 MB
Process id	5344
Numero de procesadores	8
Path de ejecucion	undefined

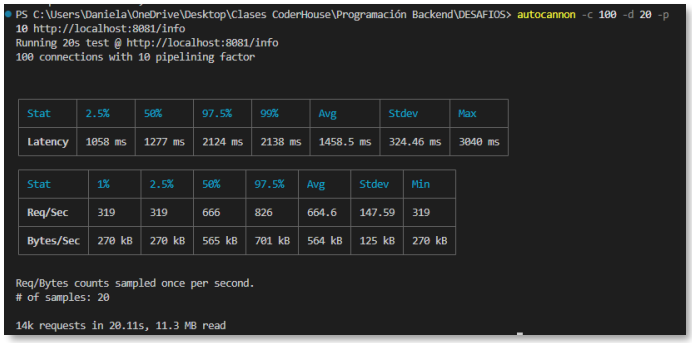
En el profiling Con Console en el documento result_prof-ConConsole.txt se observa que el consumo de node.exe es de 843 ticks.

En el profiling Sin Console en el documento result_prof-SinConsole.txt se observa que el consumo de node.exe es de 665 ticks.

En Artillery Con Console en el documento artillery_ConConsole.txt el proceso tarda 264 y el promedio es de 153

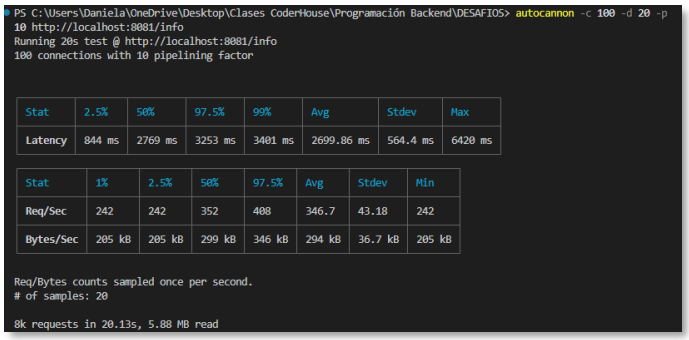
En Artillery Sin Console en el documento artillery_SinConsole.txt el proceso tarda 377 y el promedio es de 87.4

Autocannon Sin console



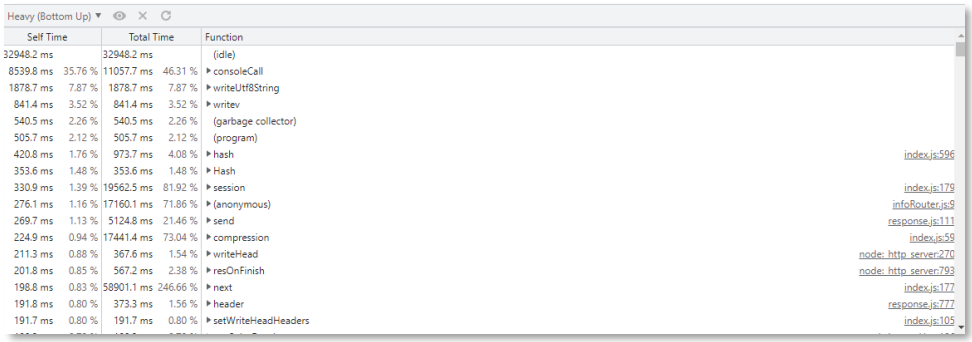
```
PS C:\Users\Daniela\OneDrive\Desktop\Clases CoderHouse\Programación Backend\DESAFIOS> autocannon -c 100 -d 20 -p
10 http://localhost:8081/info
Running 20s test @ http://localhost:8081/info
100 connections with 10 pipelining factor


| Stat | 2.5% | 50% | 97.5% | 99% | Avg | Stdev | Max |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Latency | 1058 ms | 1277 ms | 2124 ms | 2138 ms | 1458.5 ms | 324.46 ms | 3040 ms |

| Stat | 1% | 2.5% | 50% | 97.5% | Avg | Stdev | Min |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Req/Sec | 319 | 319 | 666 | 826 | 664.6 | 147.59 | 319 |
| Bytes/Sec | 270 kB | 270 kB | 565 kB | 701 kB | 564 kB | 125 kB | 270 kB |

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

14k requests in 20.11s, 11.3 MB read
```

Autocannon Con console



```
PS C:\Users\Daniela\OneDrive\Desktop\Clases CoderHouse\Programación Backend\DESAFIOS> autocannon -c 100 -d 20 -p
10 http://localhost:8081/info
Running 20s test @ http://localhost:8081/info
100 connections with 10 pipelining factor


| Stat | 2.5% | 50% | 97.5% | 99% | Avg | Stdev | Max |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Latency | 844 ms | 2769 ms | 3253 ms | 3401 ms | 2699.86 ms | 564.4 ms | 6420 ms |

| Stat | 1% | 2.5% | 50% | 97.5% | Avg | Stdev | Min |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Req/Sec | 242 | 242 | 352 | 408 | 346.7 | 43.18 | 242 |
| Bytes/Sec | 205 kB | 205 kB | 299 kB | 346 kB | 294 kB | 36.7 kB | 205 kB |

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

8k requests in 20.13s, 5.88 MB read
```

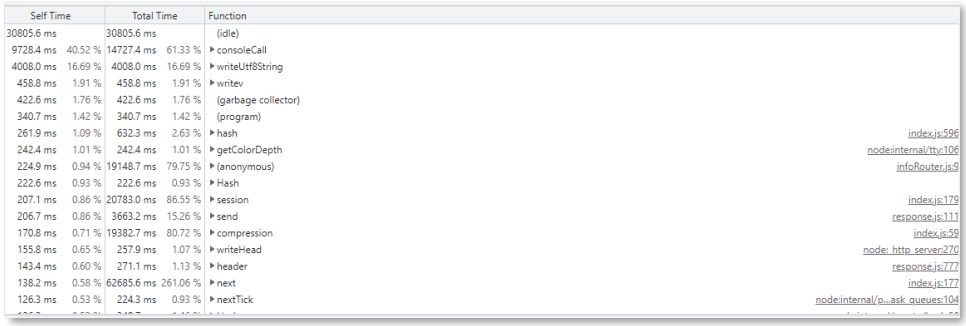
Inspector Sin Console



Heavy (Bottom Up)

Self Time		Total Time		Function	
32948.2 ms		32948.2 ms		(idle)	
8539.8 ms	35.76 %	11057.7 ms	46.31 %	consoleCall	
1878.7 ms	7.87 %	1878.7 ms	7.87 %	writeUtf8String	
841.4 ms	3.52 %	841.4 ms	3.52 %	writev	
540.5 ms	2.26 %	540.5 ms	2.26 %	(garbage collector)	
505.7 ms	2.12 %	505.7 ms	2.12 %	(program)	
420.8 ms	1.76 %	973.7 ms	4.08 %	hash	index.js:596
353.6 ms	1.48 %	353.6 ms	1.48 %	Hash	
330.9 ms	1.39 %	19562.5 ms	81.92 %	session	index.js:179
276.1 ms	1.16 %	17160.1 ms	71.86 %	(anonymous)	infoRouter.js:9
269.7 ms	1.13 %	5124.8 ms	21.46 %	send	response.js:111
224.9 ms	0.94 %	17441.4 ms	73.04 %	compression	index.js:59
211.3 ms	0.88 %	367.6 ms	1.54 %	writeHead	node: http_server:270
201.8 ms	0.85 %	567.2 ms	2.38 %	resOnFinish	node: http_server:793
198.8 ms	0.83 %	58901.1 ms	246.66 %	next	index.js:177
191.8 ms	0.80 %	373.3 ms	1.56 %	header	response.js:777
191.7 ms	0.80 %	191.7 ms	0.80 %	setWriteHeadHeaders	index.js:105

Inspector Con Console



Self Time		Total Time		Function	
30805.6 ms		30805.6 ms		(idle)	
9728.4 ms	40.52 %	14727.4 ms	61.33 %	consoleCall	
4008.0 ms	16.69 %	4008.0 ms	16.69 %	writeUtf8String	
458.8 ms	1.91 %	458.8 ms	1.91 %	writev	
422.6 ms	1.76 %	422.6 ms	1.76 %	(garbage collector)	
340.7 ms	1.42 %	340.7 ms	1.42 %	(program)	
261.9 ms	1.09 %	632.3 ms	2.63 %	hash	index.js:596
242.4 ms	1.01 %	242.4 ms	1.01 %	getColorDepth	node:internal/tty:106
224.9 ms	0.94 %	19148.7 ms	79.75 %	(anonymous)	infoRouter.js:9
222.6 ms	0.93 %	222.6 ms	0.93 %	Hash	
207.1 ms	0.86 %	20783.0 ms	86.55 %	session	index.js:179
206.7 ms	0.86 %	3663.2 ms	15.26 %	send	response.js:111
170.8 ms	0.71 %	19382.7 ms	80.72 %	compression	index.js:59
155.8 ms	0.65 %	257.9 ms	1.07 %	writeHead	node: http_server:270
143.4 ms	0.60 %	271.1 ms	1.13 %	header	response.js:777
138.2 ms	0.58 %	62685.6 ms	261.06 %	next	index.js:177
126.3 ms	0.53 %	224.3 ms	0.93 %	nextTick	node:internal/p...ask_queues:104

Diagrama de flama con 0x Con Console

Diagrama de flama con 0x Sin Console

Conclusiones:

Luego de llevar a cabo las múltiples pruebas requeridas por el entregable, se evidencia que la eliminación de los console.log mejora el rendimiento en los tiempos de respuesta y por ende se puede observar menos carga en los procesos y mejora del Performance. Por lo que es importante solo trabajar con ellos en las fases de producción y deben ser eliminados para la entrada en producción de nuestra aplicación.