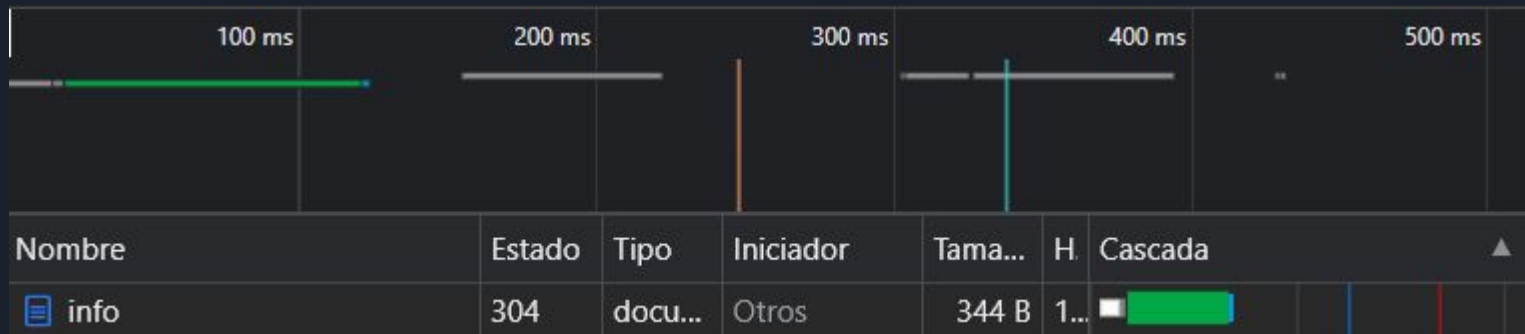




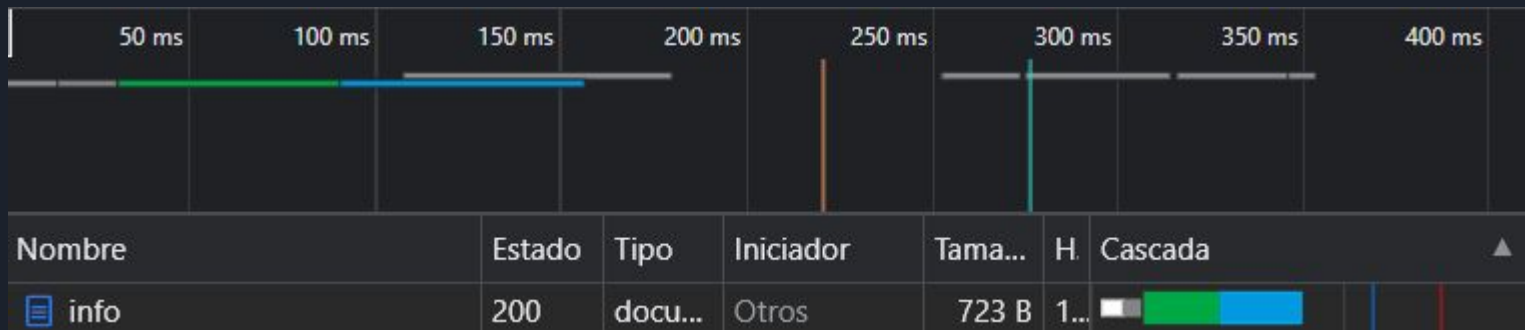
# Desafío Back End Coder House

Loggers, gzip y análisis de performance

## Ruta /info con compresión



## Ruta /info sin compresión



Profiling con - -prof (ver carpeta src/performanceAnalisys)

Prof con Console.log

[Shared libraries]:

| ticks | total | nonlib | name                             |
|-------|-------|--------|----------------------------------|
| 13708 | 96.7% |        | C:\Windows\SYSTEM32\ntdll.dll    |
| 432   | 3.0%  |        | C:\Program Files\nodejs\node.exe |

Prof sin Console.log

[Shared libraries]:

| ticks | total | nonlib | name                             |
|-------|-------|--------|----------------------------------|
| 7143  | 94.9% |        | C:\Windows\SYSTEM32\ntdll.dll    |
| 342   | 4.5%  |        | C:\Program Files\nodejs\node.exe |

## Artillery con console.log

```
http.codes.200: ..... 1000
http.request_rate: ..... 47/sec
http.requests: ..... 1000
http.response_time:
  min: ..... 1
  max: ..... 2008
  median: ..... 153
```

## Artillery sin console.log

```
http.codes.200: ..... 1000
http.request_rate: ..... 57/sec
http.requests: ..... 1000
http.response_time:
  min: ..... 1
  max: ..... 1983
  median: ..... 149.9
```

## Autocannon con console.log

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

| Stat    | 2.5%   | 50%    | 97.5%   | 99%     | Avg       | Stdev     | Max     |
|---------|--------|--------|---------|---------|-----------|-----------|---------|
| Latency | 199 ms | 961 ms | 1218 ms | 1877 ms | 896.34 ms | 275.76 ms | 2792 ms |

| Stat      | 1%      | 2.5%    | 50%     | 97.5%   | Avg     | Stdev   | Min     |
|-----------|---------|---------|---------|---------|---------|---------|---------|
| Req/Sec   | 51      | 51      | 110     | 120     | 107.25  | 14.4    | 51      |
| Bytes/Sec | 39.7 kB | 39.7 kB | 85.8 kB | 93.4 kB | 83.5 kB | 11.2 kB | 39.7 kB |

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

2k requests in 20.17s, 1.67 MB read

## Autocannon sin console.log

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

| Stat           | 2.5%   | 50%    | 97.5%   | 99%     | Avg       | Stdev     | Max     |
|----------------|--------|--------|---------|---------|-----------|-----------|---------|
| <b>Latency</b> | 146 ms | 966 ms | 1151 ms | 1906 ms | 890.54 ms | 288.21 ms | 2847 ms |

| Stat             | 1%      | 2.5%    | 50%     | 97.5%   | Avg     | Stdev   | Min     |
|------------------|---------|---------|---------|---------|---------|---------|---------|
| <b>Req/Sec</b>   | 62      | 62      | 111     | 120     | 107.75  | 12.43   | 62      |
| <b>Bytes/Sec</b> | 48.3 kB | 48.3 kB | 86.5 kB | 93.5 kB | 83.9 kB | 9.68 kB | 48.3 kB |

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

2k requests in 20.14s, 1.68 MB read

# Diagrama de flama con console.log

node src/server.js

cold hot  
\* optimized ~ unoptimized - + search functions



all stacks

# Diagrama de flama sin console.log

node src/server.js

cold hot  
\* optimized ~ unoptimized

- +

search functions



all stacks



En mi caso no note gran diferencia en rendimiento al hacer los test con y sin el console.log de la ruta /info

Si se aprecia una gran diferencia al usar `app.use(compression())`