

## Análisis de autocannon

```
done in 21.45s.  
→ tarea git:(master) x yarn test:autocannon  
yarn run v1.22.10  
$ node src/benchmark.js http://localhost:8080/info  
Running 20s test @ http://localhost:8080/info  
100 connections
```

Stat	2.5%	50%	97.5%	99%	Avg	Stdev	Max
Latency	43 ms	66 ms	183 ms	215 ms	81.52 ms	37.19 ms	276 ms

Stat	1%	2.5%	50%	97.5%	Avg	Stdev	Min
Req/Sec	742	742	1200	1644	1217.91	249.36	742
Bytes/Sec	542 kB	542 kB	876 kB	1.2 MB	889 kB	182 kB	542 kB

Req/Bytes counts sampled once per second.

24k requests in 20.05s, 17.8 MB read  
Done in 20.98s.

## Perfilamiento en modo inspect

view.controller.js x

```
1  import { processInfo } from '../utils/minimist.util.js';  
2  import ProductoService from '../services/productos.service.js';  
3  import { logger } from '../utils/logger.js';  
4  const productoService = new ProductoService();  
5  
6  class ViewController{  
7  
8      async renderProductos(req,res){  
9          try {  
10             const productos = await productoService.listarUsuariosRandom();  
11             if(productos.length===0){  
12                 logger.warn("No se encontraron productos");  
13             }  
14             logger.info(productos);  
15             res.render('productos',{  
16                 productos  
17             });  
18             catch (error) {  
19                 logger.error(error);  
20             }  
21         }  
22     }  
23  
24     getSystemInfo(req,res){  
25         const mensaje = `<b>Argumentos de entrada:</b> ${processInfo.commandLineArgs} <br>  
26         <b>Plataforma:</b> ${processInfo.sistemaOperativo} <br>  
27         <b>Versión node:</b> ${processInfo.nodeVersion} <br>  
28         <b>Memoria rss:</b> ${processInfo.memoriaRSS} <br>  
29         <b>Path de ejecución:</b> ${processInfo.nodePath} <br>  
30         <b>Process id:</b> ${processInfo.processId} <br>  
31         <b>Carpeta del proyecto:</b> ${processInfo.projectPath} <br>  
32         <b>Numero de procesos:</b> ${processInfo.cpus}<br>`;  
33         if(process.env.NODE_ENV == 'production'){  
34             logger.info(mensaje);  
35         }else{  
36             console.log(mensaje);  
37         }  
38         res.send(mensaje);  
39     }  
40 }  
41  
42 export default ViewController;
```

### Diagrama Flama 0x

```
node src/index.js -p 8080
```

