RESULTADO PROFILING CON AUTOCANNON

```
PS C:\Users\Marcos\Documents\Coderhouse\backend\clase 32> npm run autocannon

> clase-26@1.0.0 autocannon
> node autocannon.js

Running all benchmarks in parallel
Running 20s test @ http://localhost:8081/info
100 connections
```

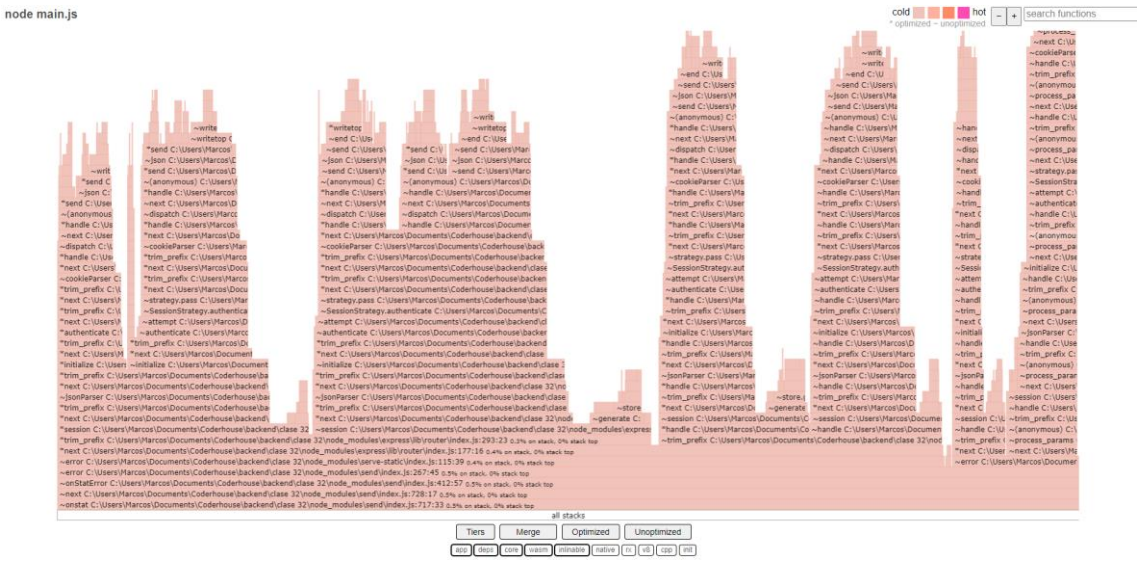
Stat	2.5%	50%	97.5%	99%	Avg	Stdev	Max
Latency	385 ms	998 ms	1489 ms	2312 ms	994.87 ms	249.92 ms	3261 ms

Stat	1%	2.5%	50%	97.5%	Avg	Stdev	Min
Req/Sec	82	82	98	112	97.4	4.96	82
Bytes/Sec	49 kB	49 kB	58.6 kB	67 kB	58.2 kB	2.97 kB	49 kB

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

2k requests in 20.05s, 1.16 MB read
PS C:\Users\Marcos\Documents\Coderhouse\backend\clase 32>
```

GRÁFICO DE FLAMA



Los resultados obtenidos indican que para la carga de trabajo solicitada (2000 req en 20 segundos) el servidor puede mantener tiempos de respuesta promedios resolviendo 87,4 request en promedio por segundo.

El grafico de flama indica que todos los procesos derivados de main.js se encuentran optimizados en el stack.