**Entregable:** LOGGERS, GZIP y ANÁLISIS DE PERFORMANCE

**Incorporar al proyecto de servidor de trabajo la compresión gzip.**

**Verificar sobre la ruta /info con y sin compresión, la diferencia de cantidad de bytes devueltos en un caso y otro.**

Genere una ruta /info y otra /infgzip, los resultados fueron los siguientes:

Tabla

Descripción generada automáticamente

**Luego implementar loggueo (con alguna librería vista en clase) que registre lo siguiente:**

* **Ruta y método de todas las peticiones recibidas por el servidor (info)**
* **Ruta y método de las peticiones a rutas inexistentes en el servidor (warning)**
* **Errores lanzados por las apis de mensajes y productos, únicamente (error)**

Texto

Descripción generada automáticamente

**Ruta y método de todas las peticiones recibidas por el servidor (info)**

Texto

Descripción generada automáticamente

Una captura de pantalla de un celular

Descripción generada automáticamente con confianza media

**Ruta y método de las peticiones a rutas inexistentes en el servidor (warning)**

Texto

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto, Aplicación, Chat o mensaje de texto

Descripción generada automáticamente

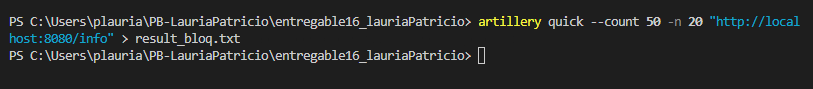
**El perfilamiento del servidor, realizando el test con --prof de node.js. Analizar los resultados obtenidos luego de procesarlos con --prof-process.**

**Utilizaremos como test de carga Artillery en línea de comandos, emulando 50 conexiones concurrentes con 20 request por cada una. Extraer un reporte con los resultados en archivo de texto.**

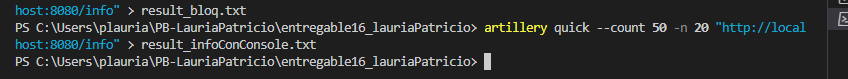
Texto

Descripción generada automáticamente

Info sin console log



Info con console log



Los resultados fueron los siguientes:

Texto

Descripción generada automáticamente

Los resultados obtenidos luego de procesarlos con --prof-process fueron:

Captura de pantalla de computadora

Descripción generada automáticamente

**Luego utilizaremos Autocannon en línea de comandos, emulando 100 conexiones concurrentes realizadas en un tiempo de 20 segundos. Extraer un reporte con los resultados (puede ser un print screen de la consola)**

**2) El perfilamiento del servidor con el modo inspector de node.js --inspect. Revisar el tiempo de los procesos menos performantes sobre el archivo fuente de inspección.**

Texto

Descripción generada automáticamente

Pantalla de computadora con letras

Descripción generada automáticamente con confianza media

Texto

Descripción generada automáticamente

Imagen que contiene Interfaz de usuario gráfica

Descripción generada automáticamente

**El diagrama de flama con 0x, emulando la carga con Autocannon con los mismos parámetros anteriores**

Imagen que contiene Interfaz de usuario gráfica

Descripción generada automáticamente