

Trabajo Práctico 4: Servidor de LogAnalyzer - Primera Versión

Programación Concurrente - Primer cuatrimestre 2025

Profesores: Emilio López Gabeiras y Rodrigo Pazos

8 de abril de 2025

1. Introducción

Este trabajo práctico se construye sobre el servidor desarrollado para el TP3. Además, pueden utilizar la biblioteca desarrollada para el trabajo práctico de Grep, aunque para esta versión no es estrictamente necesario. El objetivo es implementar un servidor destinado a analizar archivos de logs. Debe procesar archivos enviados por los clientes y generar estadísticas sobre las excepciones encontradas en dichos logs. El servidor debe ser capaz de manejar múltiples solicitudes simultáneamente.

2. Objetivo

El servidor debe soportar dos tipos principales de solicitudes:

- **POST /upload:** Permite cargar un archivo de logs. El archivo debe tener un nombre asociado (por ejemplo, el nombre del servicio y un timestamp). El servidor procesará el archivo para contar las líneas que contienen la palabra `exception` (sin importar mayúsculas o minúsculas).
- **GET /stats:** Devuelve estadísticas globales sobre los archivos procesados, incluyendo:

- Cantidad total de excepciones encontradas.
- Número de archivos procesados.
- Listado de excepciones encontradas por archivo.

3. Características Adicionales

- **Control de concurrencia:** No se deben procesar más de 4 archivos en simultáneo. En el caso de que no se pueda procesar un archivo, se debe devolver un error 429 Too Many Requests.
- **Acceso concurrente a datos compartidos:** Este servidor será mayormente usado para consultas de las estadísticas. Considerar esto en el diseño final para maximizar la concurrencia de lectura.

Tip: pueden utilizar la librería tokio (<https://docs.rs/tokio/latest/tokio/index.html>) para este trabajo, pero solo pueden usar una struct.

4. Ejemplos de Uso

A continuación se muestran algunos ejemplos de cómo interactuar con el servidor utilizando la herramienta `curl` desde la línea de comandos.

Subir un archivo de logs exitosamente

```
curl -X POST -F "file=@archivo.txt" http://localhost:3030/upload
HTTP/1.1 200 OK
Processed file: archivo.txt
```

Obtener estadísticas globales

```
curl http://localhost:3030/stats
HTTP/1.1 200 OK
Total exceptions: 5
Files processed: 2
Per file: {"archivo1.txt": 3, "archivo2.txt": 2}
```

Error por archivo vacío o no adjunto

```
curl -X POST -F "file=" http://localhost:3030/upload
HTTP/1.1 400 Bad Request
File not found or empty
```

Error por sobrecarga del servidor (demasiados archivos en procesamiento)

```
curl -X POST -F "file=@archivo.txt" http://localhost:3030/upload
HTTP/1.1 429 Too Many Requests
Too many files being processed
```

Ruta no válida

```
curl http://localhost:3030/invalid
HTTP/1.1 400 Bad Request
Valid routes:
POST /upload - Upload a file for analysis
GET /stats - Show statistics
```