URL File Processor

Problem Statement

URL File Generator is an existing system that generates files with URLs periodically. Each file contains multiple URLs, one URL per line. The generated files are kept in a designated folder on a disk. URL files are given in an inputData.zip file.

Assignment:
Build a URL File Processor that reads the files generated by URL File Generator and processes URL files as efficiently as possible.

Important points to note:

- You can build this URL File Processor in Scala or Java.
- URL File Processor is expected to run only on one machine. You can assume that machine available has limited (few GB) memory (RAM) and 4 CPU cores.
- URL File Generator may generate files of uneven size. Thus one file may be 5MB whereas other may be 1GB. Design URL File Processor accordingly to process the files as efficiently as possible. Try to make maximum & efficient use of available processing power & memory.
- What does processing a URL file mean?
    - Processing a URL file means making HTTP GET call on all URLs in a file
    - You are provided with all the files in a zip inputData.zip that were generated by URL File Generator. For test purposes all the URLs in the file point to localhost.
    - You are provided with a Mock HTTP Server in a file mockhttps.tar
    - Extract the tar mockhttps.tar
    - To start HTTP Server, execute command: *java -cp "mockhttps-1.0-SNAPSHOT/lib/*" com.ebay.ads.https.HttpServer*
    - Server runs on port 8080, so please make sure the port is available.
- Max length of URL is 4096 characters. Files are UTF-8 encoded.
- You may design URL File Processor to process files in any order.
- You may design URL File Processor to process URLs in any order.
- URL File Processor should periodically report number of URLs successfully processed, URLs failed to process, total processed URLs.
- Try to keep use of third party libraries to a minimum. You may use third party libraries for the HTTP client.
- Please submit your solution through a github link or email.

Instructions

1. Ensure you have access to the following supporting files:
    a. inputData.zip
    b. mockhttps.tar
2. Design and code a solution to the above described problem.
3. Please email your solution to the following address via a zipped attachment or github link: *DL-eBay-AdsCandidateAssignment@ebay.com*
4. Your submission needs to be received by the prescribed date and time.