Hashing-as-a-Service

At code.star, we have to build an console application that computes the hash of the content of a text file line-by-line and writes it to an output file (provided from command line). For instance:

```
$ compute-hash ./input.txt ./output.txt
```

Given `input.txt` as follows:

```
a
b
…
```

The `output.txt` should be :

```
hash-of-a
hash-of-b
…
```

Note following:

- The order of list of hash in `output.txt` should corresponds to the same order as in `input.txt`.
- The application should be able to process large input files without getting any out-of-memeroy exceptions.

Approach

Instead of building the application from scratch, we have decided to reuse an existing REST service, which:

1. Accepts batch requests with a list of strings to hash.
2. Returns a list of hashes of that strings (in the same order).

For instance:

Submitting a job for processing: `POST /api/service`

```
{
    "id": "job-1",
    "lines": [
       "This a line",
       "And that's another line."
    ]
  }
```

You get following response back.

```
 {
    "id": "job-1",
    "lines": [
        "62812ce276aa9819a2e272f94124d5a1",
        "13ea8b769685089ba2bed4a665a61fde"
    ]
 }
```

Note that, due to *micro-service* architecture, there could be latency in responding to a request and also, that it can fail.

Example of error response: `HTTP 500 (Service Unavailable)`.

```
{
    "error": "Something went wrong!"
}
```

Detail of the service and how to run it is available here.

Tool-Set

Please use any tool that you think fits best. We would however prefer to see you using Scala to solve this problem.

We expect to receive the solution –

- with sufficient unit- and integration-tests.
- in a private GIT respository (e.g., BitBucket) with a README on how to run it.

If you have any question, please let us know.