

## CODING TASK SOLUTION

### Application's purpose

Design and implement REST interface to consume data-snapshots from one client, validate and persist data in storage, distribute persisted data to other clients via REST interface.

### Technological stack used in project

Java SE8, Gradle, Spring Boot, Apache Commons

### Testing

Postman

### Short project description

In the rest controller three mapping methods have been created:

- **POST** method provides you to upload a single plain text file and gives to a client response contains message and http status code. Method also validates some uploading file issues:
  - Checks that file is not empty and has obligatory .txt extension, header, not blank primary key and suitable single record length. Otherwise exceptions are throw
- **GET** method allows you to get a record from the uploaded text file by primary key
  - If primary key was not found or some other issue occurred method throws an exception
- **DELETE** method ensures you to delete the record from the uploaded text file by primary key
  - If primary key was not found or some other issue occurred method throws an exception

### Assumptions in project

- a. To easily further process the data, the record in the text file is a map with primary key as the map-key and List of String arrays as the map-values. Moreover, the primary key is non-unique so could have more than one value
- b. File cannot be empty, must have obligatory header, not blank primary key, suitable record length (max. 4 elements in the single record), .txt extension and UTF-8/ANSI encoding
- c. All the file validations take place only during upload to ensure the correct file for the further operations
- d. File after upload has only the last line empty to ensure correctly reading
- e. Other elements in the record except the primary key may be blank

- f. Non validation for the type of primary key and the other elements in the record due to assumption that they may be of any type (not only numeric primary key and timestamp in an usual date format etc.)
- g. Every possible operation and scenario have been tested in Postman so I didn't write the Unit tests