Testing Document and Specification

ETL - Support 2019 - Group 2

Team Members

- Elia Vicentini (Scrum Master)
- Simone Gandini
- Luca Landolfo
- Enrico Marocchio
- Adrian Munteanu

Document Owner

Adrian Munteanu is responsible for the development and maintenance of this document.

Test Case ID:	1.1
Title:	Write data on CSV file from MySQL DataBase
Feature/Sub-feature:	Application arguments
Purpose:	To ensure that it is possible to connect to MySQL server and the data is being saved on CSV file.
Test Data:	Test Data will include info taken from MySQL DataBase
Test Actions:	1. Open cmd 2. Run the `exportdata.py` with `--mode direct` and `--noupload` (to do not insert any data) as arguments
Expected Results:	After Step 2 a CSV file named `database-direct.csv` should appear in the folder with no errors.

Test Case ID:	1.2
Title:	Write data on CSV file from API
Feature/Sub-feature:	Application arguments
Purpose:	To ensure that it is possible to connect to API and the data is being saved on CSV file.
Test Data:	Test Data will include info taken from API
Test Actions:	1. Open cmd 2. Run the `exportdata.py` with `--mode api` and `--noupload` (to do not insert any data) as arguments
Expected Results:	After Step 2 a CSV file named `database-api.csv` should appear in the folder with no errors.

Test Case ID:	1.3
Title:	Log file
Feature/Sub-feature:	Logs
Purpose:	To assure that all actions are tracked in the log file
Test Data:	Test Data will include info taken from any source (MySQL server or API)
Test Actions:	1. Open cmd 2. Run the `exportdata.py` in any mode and `--noupload` (to do not insert any data) as argument
Expected Results:	After Step 2 a log file named `exportadata_CURRENTDATE.log` should appear in the `log` folder with no errors.

Test Case ID:	1.4
Title:	Insert data into MySQL DataBase
Feature/Sub-feature:	Application arguments
Purpose:	To make sure that it is possible to connect to MySQL server and insert data
Test Data:	Test Data will include info taken from any source (API or direct)
Test Actions:	1. Open cmd 2. Run the `exportdata.py` in any mode and with `--mysql` as argument
Expected Results:	After Step 2 data from CSV file should be inserted in the MySQL DataBase with no errors.

Test Case ID:	1.5
Title:	Insert data into SQL DataBase
Feature/Sub-feature:	Application arguments
Purpose:	To make sure that it is possible to connect to SQL server and insert data
Test Data:	Test Data will include info taken from any source (API or direct)
Test Actions:	1. Open cmd 2. Run the `exportdata.py` in any mode and with `--sql` as argument

Expected Results:	After Step 2 data from CSV file should be inserted in the SQL DataBase with no errors.

Test Case ID:	1.6
Title:	Insert data into SQL DataBase
Feature/Sub-feature:	Application arguments
Purpose:	To make sure that it is possible display additional details about what the program is doing
Test Data:	Test Data will include info taken from any source (API or direct)
Test Actions:	1. Open cmd 2. Run the `exportdata.py` in any mode with `--noupload` (to do not insert any data) and `-v` as arguments
Expected Results:	After Step 2 details about what the program is doing should be displayed in the cmd.