SKaMP. Tests

The goal of this report is to provide information on the performed testing of data acquisition and data pre-processing.

GitHub repository: https://github.com/salveendutt/Big-Data-Analytics.

1 Data acquisition

Test objective	Steps	Expected Result	Actual Result
Verify data incoming from stream API	1. Start the server using start_containers.bat; 2. Navigate to http://localhost:5000	Incoming data is available on /data/0	Passed. The screenshot is provided in Fig.1 and Fig.2
Verify correct setup of the stream	Run 'pytest' from the root folder	Data stream is configured as expected; Incoming data is not null; Returned status code - 200	PASSED. The screenshot is provided in Fig. 3

Table 1: Data acquisition tests

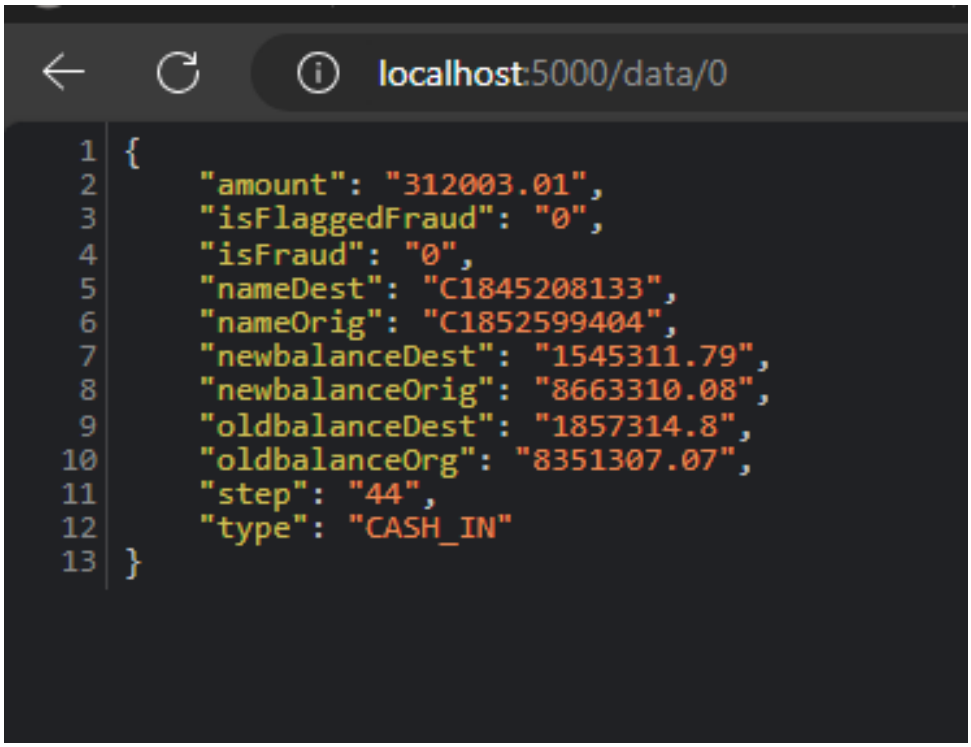

Figure 1: Data incoming via the stream

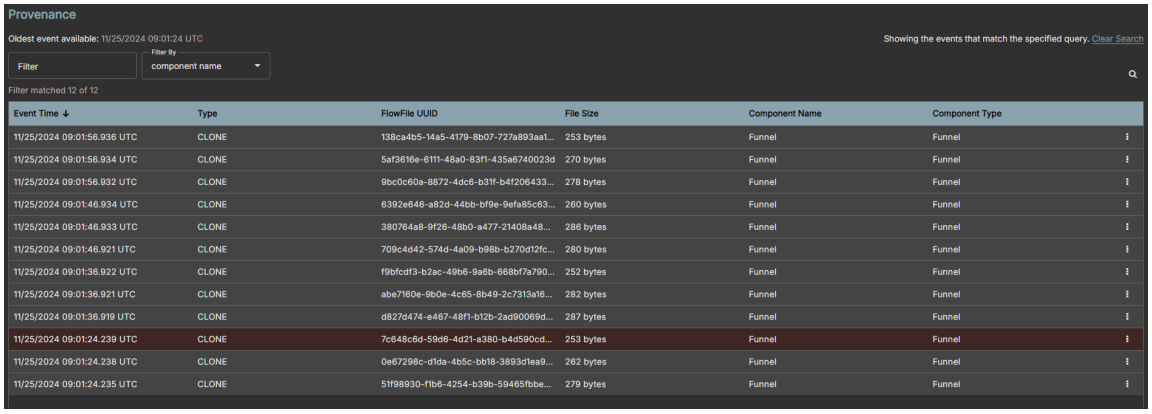

Figure 2: NiFi Data Provenance

2 Data pre-processing

Test objective	**Steps**	**Expected Result**	**Actual Result**
Verify correct data preprocessing of dataset 1	Run 'pytest' from the root folder	Feature 'type' is correctly transformed into numeric value (5 cases); Feature 'isMerchant' is correctly prepared (2 cases)	PASSED. The screenshot is provided in Fig. 3
Verify correct data preprocessing of dataset 2	Run 'pytest' from the root folder	Numeric boolean values are transformed to int from float (4 cases)	PASSED. The screenshot is provided in Fig. 3
Verify correct data preprocessing of dataset 3	Run 'pytest' from the root folder	Feature 'entry_mode' is correctly transformed into numeric value (4 cases); Unnecessary features are omitted.	PASSED. The screenshot is provided in Fig. 3
Verify correct data preprocessing of dataset 4	Run 'pytest' from the root folder	Features 'Amount', 'Class' are renamed to 'amount' and 'isFraud'; Extra features are removed	PASSED. The screenshot is provided in Fig. 3

Table 2: Data pre-processing tests

Unit testing is included in the CI/CD pipeline on GitHub and must be successful before any merge into the main branch.

```
============================================= test session starts =============================================
platform win32 -- Python 3.13.0, pytest-8.3.3, pluggy-1.5.0 -- C:\ProgramFiles\Anaconda3\envs\bigdata13\python.exe
cachedir: .pytest_cache
rootdir: C:\home\WUT\Semester_3\BigData\Big-Data-Analytics
collected 12 items

services/streaming_simulation/test_streaming_simulation.py::StreamingSimulationTestCase::test_data_stream PASSED [  8%]
tests/data_utils/test_utils.py::test_preprocess_1_payment PASSED                                            [ 16%]
tests/data_utils/test_utils.py::test_preprocess_1_cash_in PASSED                                            [ 25%]
tests/data_utils/test_utils.py::test_preprocess_1_cash_out PASSED                                           [ 33%]
tests/data_utils/test_utils.py::test_preprocess_1_debit PASSED                                              [ 41%]
tests/data_utils/test_utils.py::test_preprocess_1_unknown PASSED                                            [ 50%]
tests/data_utils/test_utils.py::test_preprocess_row_2 PASSED                                                [ 58%]
tests/data_utils/test_utils.py::test_preprocess_3_contactless PASSED                                        [ 66%]
tests/data_utils/test_utils.py::test_preprocess_3_chip PASSED                                               [ 75%]
tests/data_utils/test_utils.py::test_preprocess_3_swipe PASSED                                              [ 83%]
tests/data_utils/test_utils.py::test_preprocess_3_unknown PASSED                                            [ 91%]
tests/data_utils/test_utils.py::test_preprocess_row_4 PASSED                                                [100%]

============================================== 12 passed in 0.57s =============================================
```

Figure 3: Unit testing result

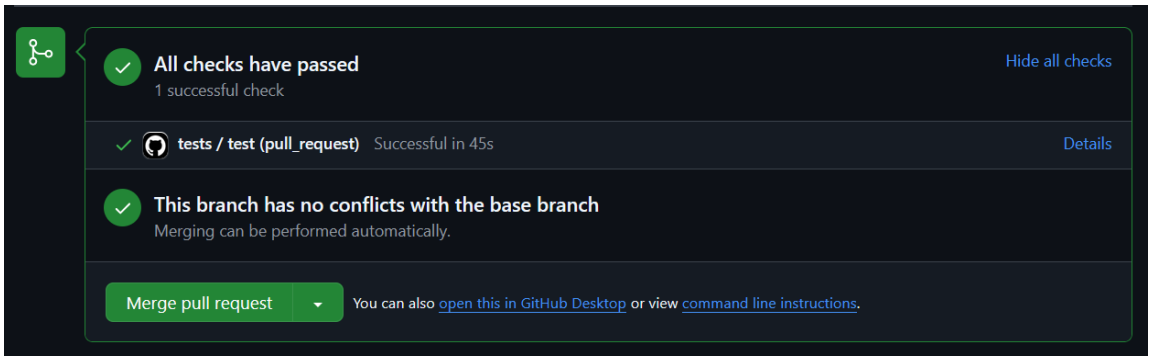

Figure 4: GitHub checks before merge