SKaMP. Tests

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

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

1 Test Scenarios

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 Figure 3
Verify correct setup of the stream and data preprocessing functions	Run 'pytest' from the root folder	Data stream is configured as expected; Incoming data is not null; Returned status code - 200. Preprocessing utils return transformed data as expected	Passed. The screenshot is provided in Figure 12
Verify the correct setup of Nifi - HDFS/Kafka flow	Run the containers - follow steps in README.md	Data flows from streamin API to Kafka topics and Hive tables	Passed. The screenshot is provided in Figure 4, 5, 6, 7
Verify the correct setup of batch processing	Run the containers - follow steps in README.md	Views are available in the Cassandra tables	Passed. The screenshot is provided in Figure 8, 9, 10
Verify the correct setup of streaming processing	Run the containers - follow steps in README.md	Views are available in the Cassandra tables	Passed. The screenshot is provided in Figure 11
Verify that Superset correctly connects to Cassandra through TrinoDB and displays data in dashboards	Run the necessary containers (start_containers) in 'scripts' folder and observe the charts in Superset	Data is visible in Superset charts	PASSED. Screenshots are visible in Figure x and x

Table 1: Test scenarios

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 Figure 12
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 Figure 12
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 Figure 12
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 Figure 12

Table 2: Data pre-processing tests

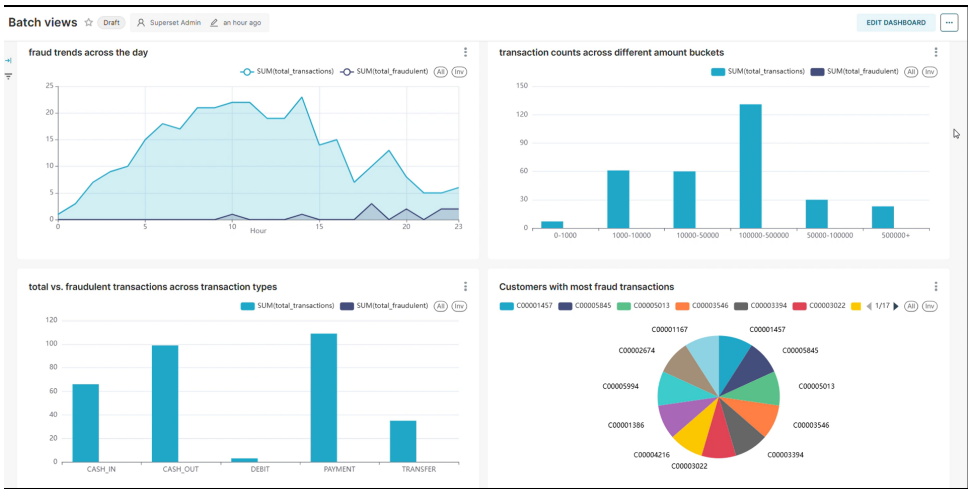

Figure 1: Batch Dashboard

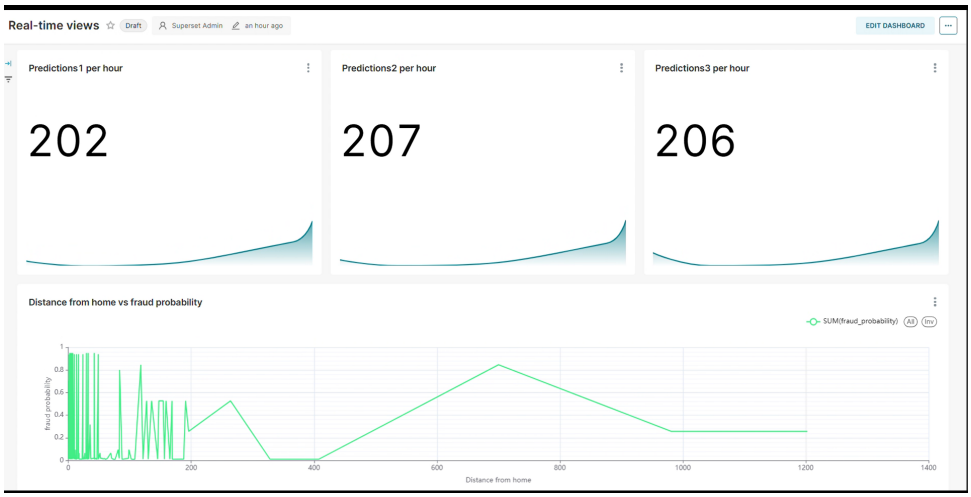

Figure 2: Speed Dashboard

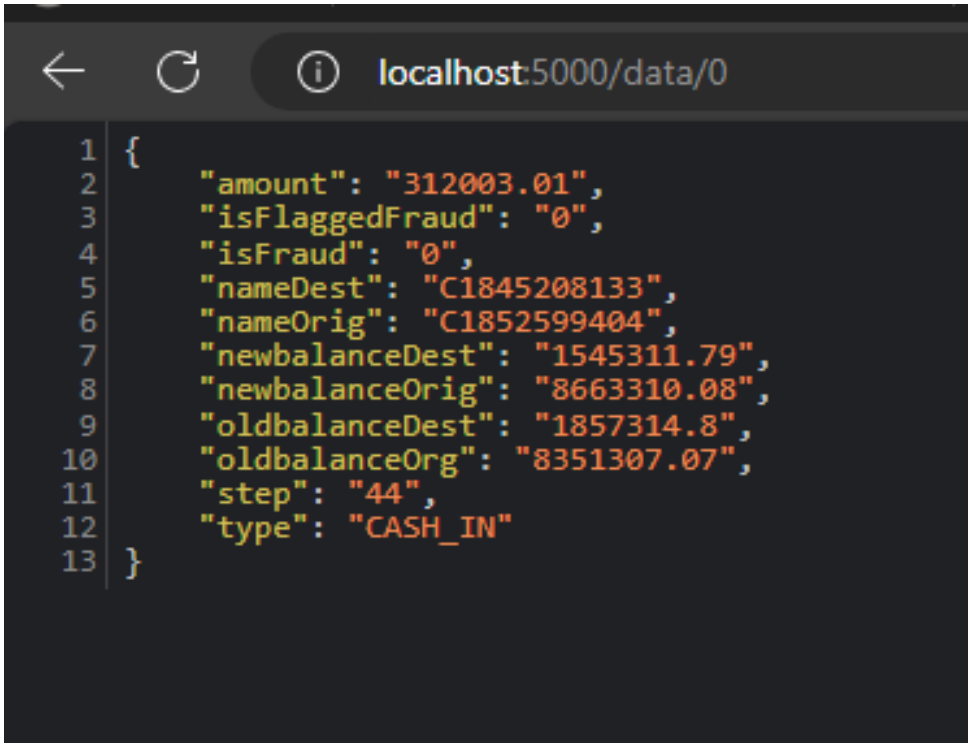

Figure 3: Data incoming via the stream

Figure 4: Kafka Dataset1

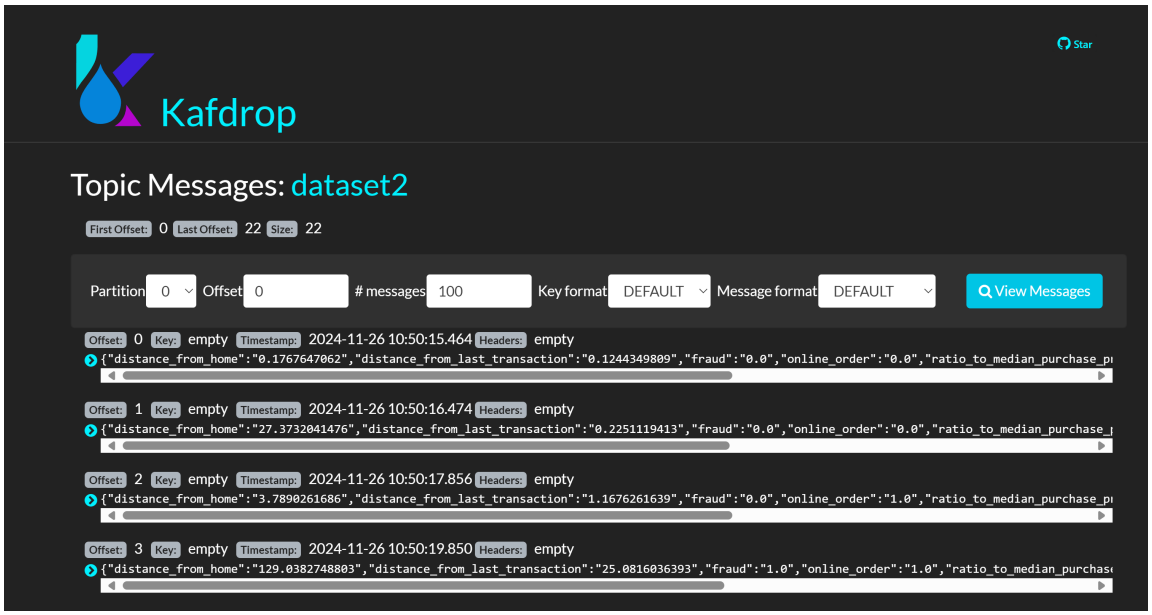

Figure 5: Kafka Dataset2

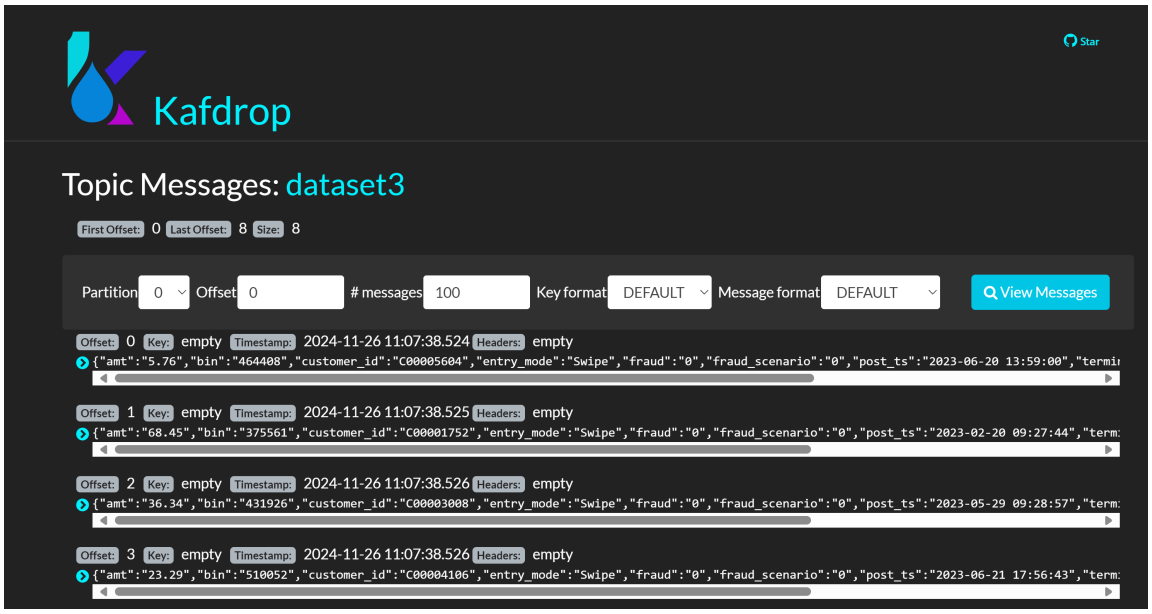

Figure 6: Kafka Dataset3

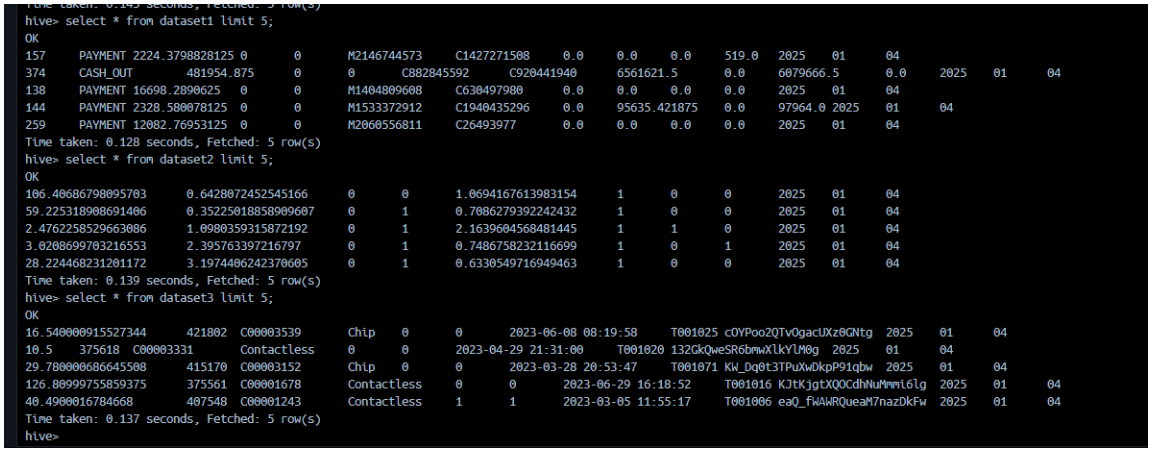

Figure 7: Hive data

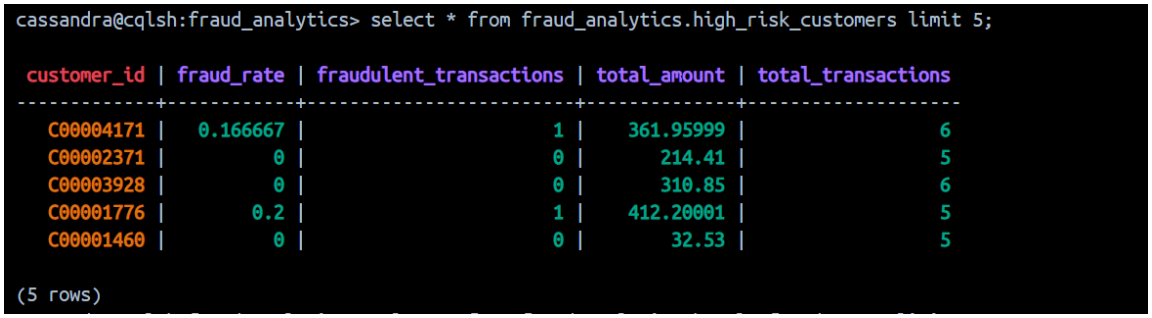

Figure 8: Cassandra batch processing views

```
cassandra@cqlsh:fraud_analytics> select * from fraud_analytics.hourly_fraud_stats limit 5;

 hour | avg_amount | total_fraudulent | total_transactions
------+------------+------------------+--------------------
   23 |   59.18323 |                6 |                 31
    5 |   54.17043 |                1 |                117
   10 |   55.43596 |                4 |                230
   16 |     55.149 |                6 |                190
   13 |   56.83678 |                3 |                242

(5 rows)
```

Figure 9: Cassandra batch processing views

```
cassandra@cqlsh:fraud_analytics> select * from fraud_analytics.fraud_by_transaction_type limit 5;

 type     | avg_amount   | fraud_rate | total_fraudulent | total_transactions
----------+--------------+------------+------------------+--------------------
 TRANSFER |  9.3788e+05 |   0.007463 |                2 |                268
 CASH_OUT |  1.7427e+05 |   0.001847 |                2 |               1083
    DEBIT |  5886.82766 |          0 |                0 |                 25
  CASH_IN |  1.7194e+05 |          0 |                0 |                708
  PAYMENT | 12990.00553 |          0 |                0 |               1165

(5 rows)
```

Figure 10: Cassandra batch processing views

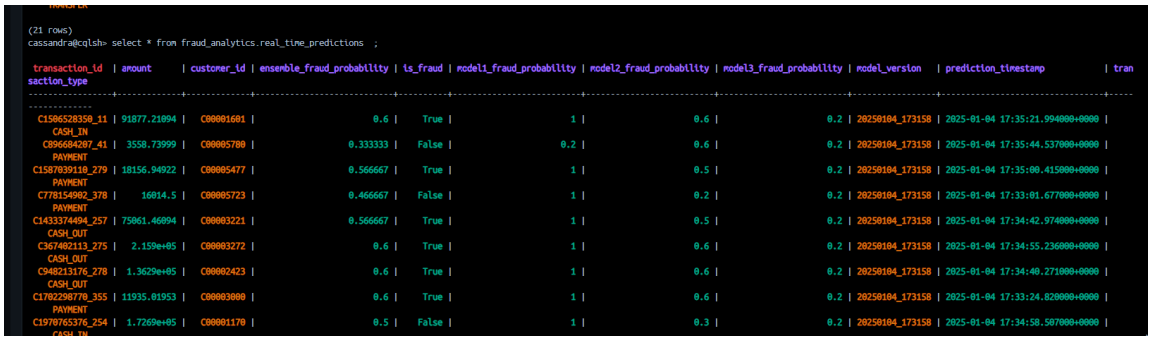

Figure 11: Cassandra stream processing views

```
============================================= 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 12: Unit testing result

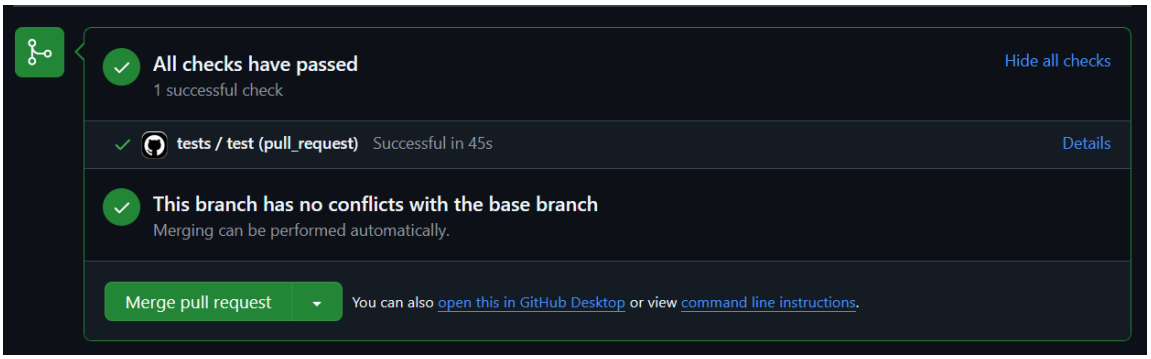

Figure 13: GitHub checks before merge