

## Exercise 9: Stream Processing with Kafka (Part III) & Assignment 2

Deadline: 02.06.2024; 23:59 CEST

---

### Project Abstract

In the tenth week's exercise on Event-driven and Process-oriented Architectures we will continue investigating the use of Kafka Streams for stateful stream processing and have a brief look at an example for ksqlDB.

### ① Labs

In the practical part of the lecture we ask you to work on the following lab to continue with stateful processing in Kafka Streams and explore ksqlDB:

- Lab14Part1 – PatientMonitoringApp<sup>1</sup>: This lab shows the Kafka Streams-based implementation of the Patient Monitoring example for stateful stream processing introduced in the lecture. Here you find a topology involving streams, tables and stateful operations like joins, groupings, filters and aggregations based on time and windows. Try to run the project and study the implementation of the stream processing app.
- Lab14Part2 – Eye Tracking Events Processing (Part3)<sup>2</sup>: This lab further demonstrates the Kafka Streams-based implementation in the eye-tracking domain. The project covers both stateless and stateful processing of events *with a focus on time windowing*. It consists of three modules: a fixation producer, a click stream producer and an event processing topology (with interactive querying and monitoring). Try to run the project, study the implementation of the event processing topology and extend it as described in the *Exercise* Section of the project README.
- Lab15 – CreditCardTransactionsApp<sup>3</sup>: This lab shows an exemplary stream processing app for detecting fraudulent credit card transactions using ksqlDB, both via the ksqlDB command line interface and from a simple Java client.

### ② Suggestions for Software Project

For your software application in Assignment 2 we suggest that you implement one or several stream processing applications including both stateless and stateful stream processing:

- At least several stateless operations (Week 8)
- Usage of both streams and tables (Week 9)
- Consider data from more than one stream (Week 9)
- Interactive queries (Week 9)
- Windowed operations (Week 10)

We suggest to use Avro schemes for serialization and deserialization of records in Kafka (registry-less or with a schema registry).

---

<sup>1</sup><https://github.com/scs-edpo/lab14Part1-kafka-streams-patientmonitoring>

<sup>2</sup><https://github.com/scs-edpo/lab14Part2-kafka-streams-eyeTracking3>

<sup>3</sup><https://github.com/scs-edpo/lab15-ksqldb-credittransactions>

## Project Expectations

The labs in Part 1 of this sheet are meant to be done during class and are **not graded**. In Part 2 we expect you to prepare a report on the project-related implementations that you have done in your group. For each developed stream processing application the **topology** should be described in **graphical form** and you should **elaborate** on the chosen **stream processing patterns**. Feel free to use the provided data sets for your experiments.

### ③ Assignment 2 Hand-in

By the deadline indicated on top of this sheet your group should hand in via the **Exercise 9** assignment module:

A zip file with your presentation slides from May 23 and a PDF describing your software application including:

- a general description of the project;
- explicit references to the concepts of the lecture and exercises and how they are covered by the project<sup>4</sup>;
- diagrams showing the topologies of your stream processing app(s);
- detailed descriptions regarding the topologies of your stream processing app(s);
- a discussion of trade-offs and important architectural decisions whenever relevant (can be complemented with ADRs);
- any additional results and insights you have gained while working and experimenting with Kafka Streams, Avro and others;
- a link to a *release* version of your software application for Assignment 2 within your Git repository and additional descriptions/links related to your implementation if necessary;
- a clear indication of who in the team was responsible for which parts;
- reflections and lessons learned;
- a revised version of your reports for Assignment 1 with respect to the feedback provided by the tutors (if desired) as well as a brief document explaining the changes made when compared to the original version.

### ④ Assignment 2 Presentations

On May 23, we will allocate a slot of approx. 15–20 minutes per group for a presentation of the projects including conceptual aspects as indicated for the Assignment 2 hand-in as well as a short live demo of your prototype(s) and Q&A. In your presentations, please focus on the newly developed aspects and significant changes from Assignment 1 (if any).

We expect every member of the group to present parts of the project and to be able to answer questions related to the project.

---

<sup>4</sup>Ensure to cover the course contents more broadly rather than to focus on a single or very few aspects.