Interview assignment

# The assignment

We would like you to make an application that make uses some of the technologies in UNI.

For this assignment you should

1. Consume currency exchange rates from kafka
2. Store to mongodb
3. Make rest endpoints for accessing data
4. Make junit tests for > 80% code coverage.
5. Optional

### Consume currency exchange rates from kafka

There is a KafkaConsumer in the **Interview/application** project which consumes and logs every published event.

You can expand this on the existing KafkaConsumer.

## Store to mongodb

Store the exchange data to mongodb in an appropriate format

## Make rest endpoints for accessing data

**/api/v1/exchange-rates/{base-currency}**

Latest exchange rates for base currency in json document

**/api/v1/exchange-rates/{base-currency}?datetime={datetime}**

Exchange rates for base currency at given time

**/api/v1/exchange-rates/{base-currency}/{currency}?base-currency-amount={value}**

Calculate a currency exchange amount, from amount in base-currency to currency, based on the latest rate.

**/api/v1/exchange-rates/{base-currency}/report**

Download a report in CSV, XML or XLS(X) lists average currency rates for the last 10 minutes grouped by minute.

**Example**

|  |  |  |  |
| --- | --- | --- | --- |
| Base currency | Currency | 09:00 UTC | 09:01 UTC |
| EUR | SEK | 10.5260 | 10.5280 |
| EUR | NOK | 9.8490 | 9.8530 |
| EUR | DKK |  |  |
| EUR | USD |  |  |

## Optional:

Expose openapi contracts

<https://www.baeldung.com/spring-rest-openapi-documentation>

# Getting started

## Starter projects:

There are two starter projects

**Interview/driver:**

This application will continuously publish exchange rates to the kafka topic.

**Interview/application:**

This application is the one you should be implementing.

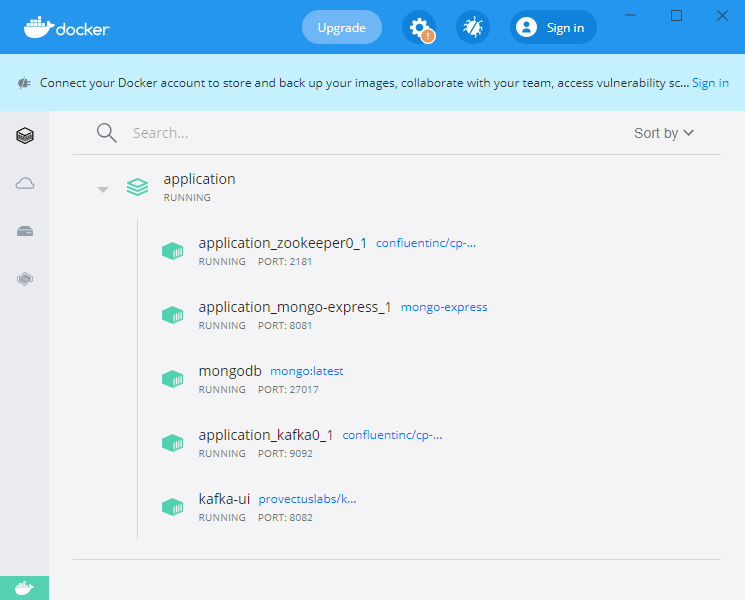
## Setup

It is not mandatory, but we recommend that you install docker desktop (it is possible to install kafka and mongo as normal apps as well).

<https://www.docker.com/products/docker-desktop>

Once docker is setup – then you can start docker compose (see HELP.md in application project)

Should look like this when running

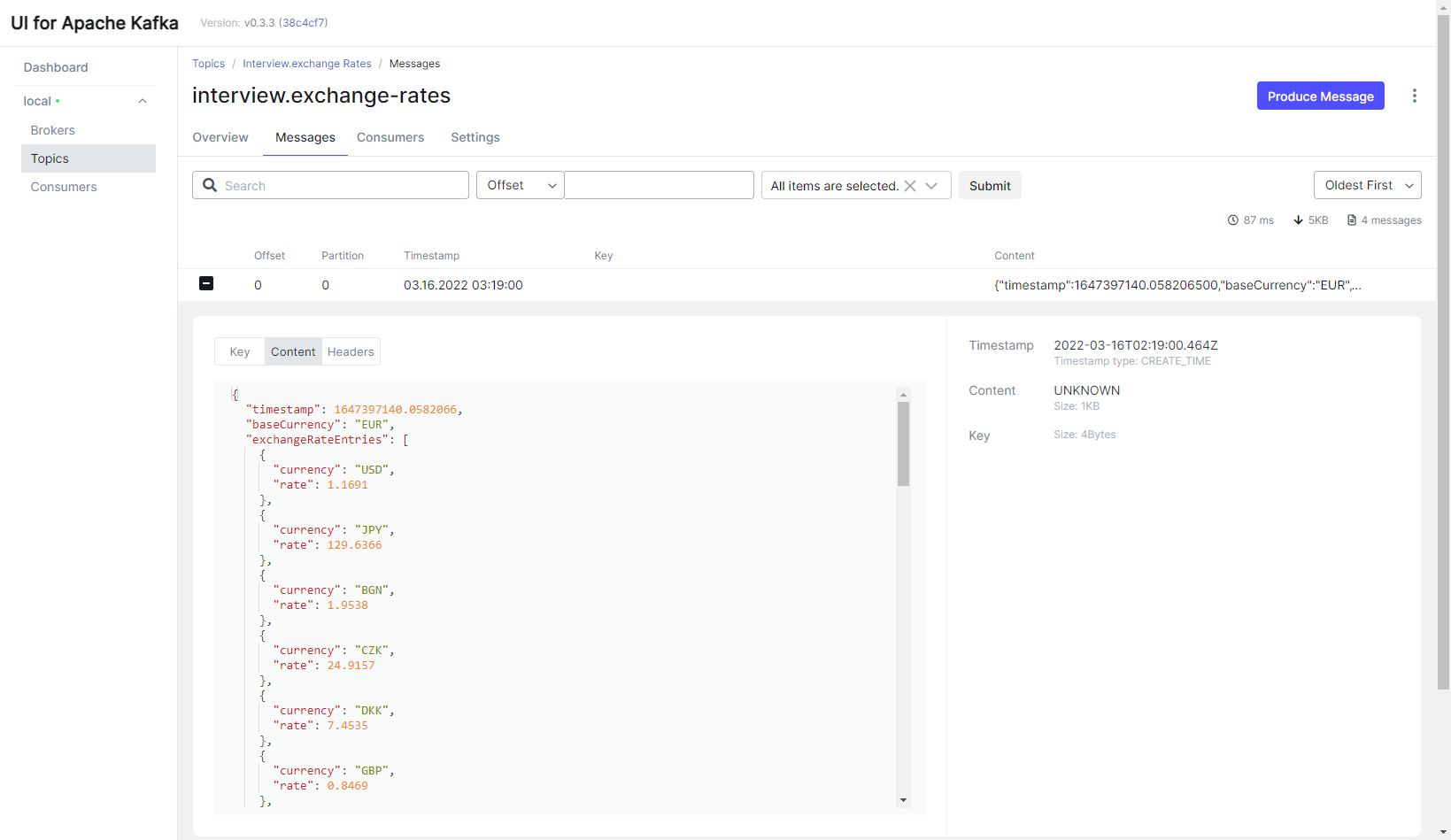


## 

## Interview/driver

When docker containers are running, start this app and it will continuously publish exchange rates to the kafka topic.

If you open kafka-ui, then you should be able to see the messages:



## Interview/application

When docker containers are running, start this app and it will continuously log the consumed exchange rates.

