**Техническое задание**

**Задание №1.1**

**Развертывание Kafka и PostgreSQL**

1. Установить и запустить Kafka и Zookeeper (локально или в Docker).
2. Создать Kafka-топик для дальнейшей работы.
3. Развернуть PostgreSQL и создать таблицу со следующими столбцами:
   * msgId
   * timeRq

*Для просмотра очереди сообщений в Kafka можно использовать Kafka Tool.*

**Задание №1.2**

**Нагрузочное тестирование Kafka**

1. Написать тестовый скрипт на JMeter или K6 (или другом инструменте НТ), который отправляет в Kafka сообщение со случайным msg\_id в теле запроса.

**Формат сообщения в Kafka:**

{

"msg\_id": "1234567890",

"timestamp": "1694171306000",

"method": "POST",

"uri": "/post-message"

}

1. Реализовать сценарий нагрузочного тестирования:
   * Нагрузка подается ступенями по 5 минут.
   * Всего 4 ступени:
     + 1-я ступень: 0,5 оп/с
     + 2-я ступень: 1 оп/с
     + 3-я ступень: 2 оп/с
     + 4-я ступень: 3 оп/с

**Задание №1.3**

**Разработка заглушки на Java (Spring)**

1. Заглушка должна:
   * Слушать Kafka-топик, созданный в п.1.1.
   * Получать сообщения и извлекать из них msg\_id и timestamp.
   * Записывать полученные данные в PostgreSQL (msgId и timeRq).

*Для просмотра сообщений в Kafka можно использовать Kafka Tool. Для контроля работы БД можно воспользоваться DBeaver.*

**Задание №1.4**

**Анализ статистики запросов в PostgreSQL**

1. Установить расширение pg\_stat\_statements в PostgreSQL.
2. Собрать статистику по наиболее ресурсоемким запросам из pg\_stat\_statements, отсортировав их по среднему времени выполнения.