**Домашнее задание № 15**

**Лабораторная работа по теме «Потоковый сбор данных (Kafka)»**

**Автор: Сорокин Максим Евгеньевич**

**Задача:**

Вы являетесь аналитиком данных в небольшой компании, которая решает перейти к современной системе управления данными. Вам поручено выбрать и обосновать выбор хранилища данных для компании. Ваша компания работает в сфере e-commerce и продает товары через онлайн-платформу.

**Формулировка задания:**

* Выберите одно или несколько хранилищ данных (реляционная база данных, NoSQL, колоночное хранилище и т. д.), которые, по вашему мнению, наилучшим образом соответствуют требованиям компании.
* Обоснуйте свой выбор, предоставив аргументы, почему данное хранилище данных является оптимальным для конкретных требований.
* Опишите структуру данных, которые будут храниться в выбранном хранилище (таблицы, колонки и связи).
* Предложите схему интеграции данных из разных источников в хранилище.
* Опишите, как бы вы настроили систему для обработки данных в реальном времени.

**Решение:**

1. Рассматриваю продукт e-commerce по сбору и анализу данных покупателей интернет-магазина одежды.

Данные о клиентах хранятся в строго структурированной реляционной базе данных PostgreSQL. Вертикальная масштабированность, то есть при росте нагрузки растет производительность сервера.

Данные об одежде хранятся в NoSQL, т.е. без четких связей друг с другом и четкой структуры. База имеет распределенную архитектуру, поэтому хорошо масштабируются горизонтально, т.е. обеспечивается высокая производительность. Используем документоориентированную базу данных MongoDB для хранения коллекции документов в формате JSON.

1. Данные о клиенте – PostgreSQL (традиционные взаимосвязи данных о клиенте: пол, рост, вес, возраст и другие позволяющие спроектировать рекомендательную систему по подбору одежды)

Продуктовый каталог – MongoDB (обеспечивает аналитику (адаптацию под витрины) в реальном времени для быстрого обновления аналитики предпочтений, нечеткие данные о типах одежды, цветовой гамме с разрозненностью узоров одежды и классов, т.е. данные разных типов, например: где-то есть длинный воротник, где-то его нет и т.п.).

1. Минимальный набор данных PostgreSQL:

Таблица Users: (пользователь)

id: bigint

name: character varying

sex: boolean

height: double precision

weight: double precision

…

Таблица Orders: (заказ)

id: bigint

user\_id: bigint

status: character

address: character varying

price: double precision

…

Таблица Cart: (корзина)

id: bigint

users\_id: bigint

product:

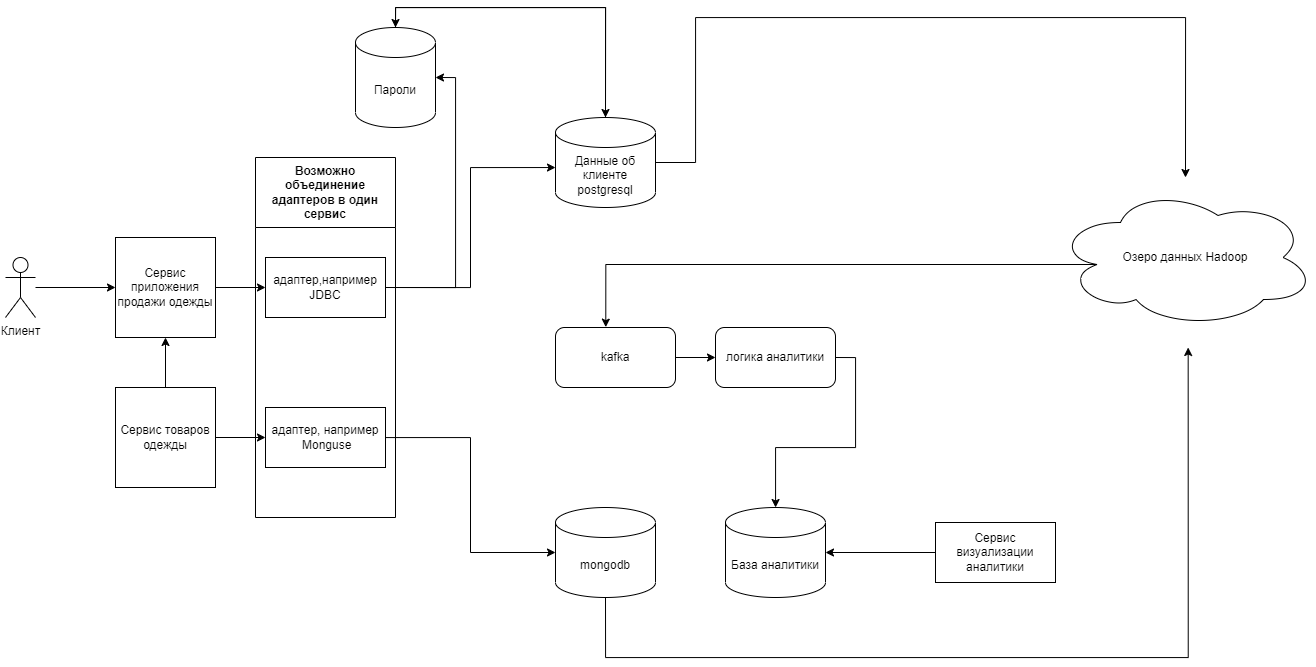
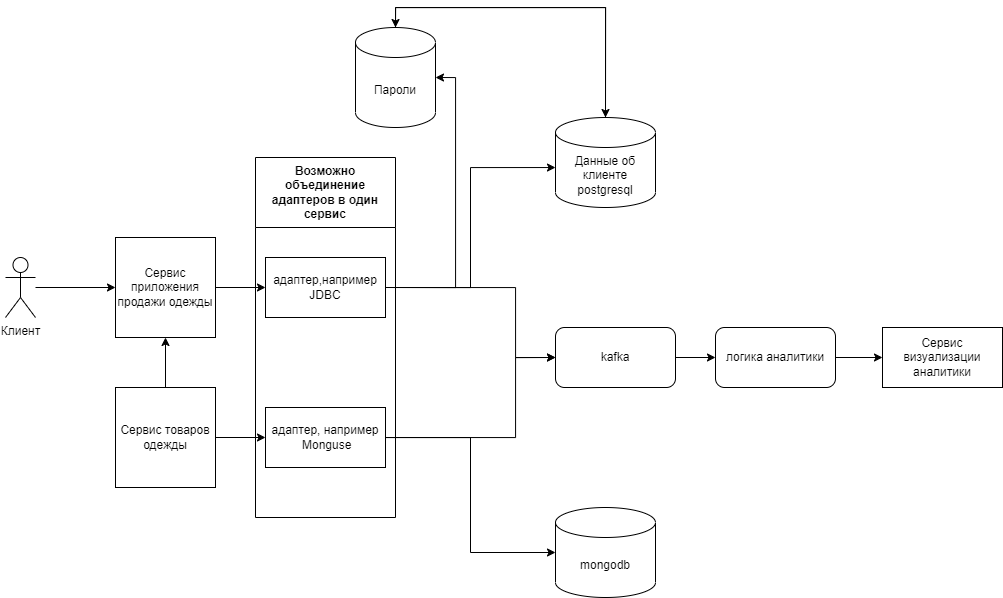
…

Минимальный набор данных MongoDB:

Product: type Document

Catalog: type Document

1. В общих чертах представим схему:



1. Для системы реального времени, использовал брокер сообщений, например kafka, rabbitQL. Озеро данных hadoop. BI инструменты мониторинга типа Ynadex Datalens.

**К решению приложены файлы:**

Схемы в Drawio

Ссылка на GitHub:

<https://github.com/SotGE/innopolis2023/tree/main/lesson15>