Міністерство освіти і науки України

ДНІПРОВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ імені Олеся Гончара

Факультет фізики, електроніки та комп’ютерних систем

Кафедра електронних обчислювальних машин

Звіт

з лабораторної роботи №4

Виконав студент групи КІ-21-2 Шейко Р. О.

Керівник: Скуратовський І. А.

м. Дніпро

2024

Завдання до лабораторної роботи:

Створення мікросервісної архітектури: обрати будь-який приклад складної системи (маркетплейс, служба таксі, система керування підприємством) провести декомпозицю її на мікросервіси.

Для цього коротко тезами опишіть вимоги до неї, розбийте її на модулі/мікросервіси і продумаєте шляхи їх взаємодії.

Намалюйте діаграму і у звіті поясніть декомпозицію і вибір необхідних компонент і засобів взаємодії.

Хід роботи

Давайте спробуємо такі обрати систему **служби таксі**, проведемо її декомпозицію на мікросервіси, створимо діаграму, а також коротко спробуємо описати кожен компонент і шляхи їхньої взаємодії.

**Вимоги до системи**

1. **Реєстрація і автентифікація користувачів** (пасажирів і водіїв).
2. **Замовлення поїздки**:
   * Вибір пункту відправлення та призначення.
   * Розрахунок вартості.
   * Пошук найближчого доступного водія.
3. **Трекінг поїздки**: перегляд місцезнаходження автомобіля в реальному часі.
4. **Розрахунок і платежі**: можливість безготівкового розрахунку.
5. **Система рейтингів і відгуків** для пасажирів і водіїв.
6. **Адміністративний контроль**:
   * Модерація контенту.
   * Управління конфігураціями системи.
7. **Аналітика і звіти** для операційної діяльності.

**Декомпозиція системи на мікросервіси**

1. **Сервіс користувачів**:
   * Реєстрація та автентифікація користувачів.
   * Зберігання профілів пасажирів і водіїв.
2. **Сервіс замовлень**:
   * Управління замовленнями.
   * Пошук водіїв.
3. **Сервіс геолокації**:
   * Отримання та збереження геоданих водіїв.
   * Трекинг автомобіля в реальному часі.
4. **Сервіс платежів**:
   * Розрахунок вартості.
   * Обробка транзакцій (включаючи інтеграцію з платіжними системами).
5. **Сервіс рейтингів і відгуків**:
   * Збір і зберігання оцінок.
   * Аналіз відгуків.
6. **Сервіс аналітики**:
   * Збір даних із різних компонентів.
   * Генерація звітів.
7. **Сервіс адміністративної панелі**:
   * Інструменти для адміністраторів.
   * Моніторинг та управління мікросервісами.

**Взаємодія між мікросервісами**

* Використовуємо **REST API** або **gRPC** для синхронної комунікації.
* Для обробки подій використовуємо **message broker** (наприклад, RabbitMQ чи Kafka).
* Дані зберігаються у спеціалізованих базах даних:
  + PostgreSQL для транзакційних даних.
  + MongoDB для зберігання геоданих.
  + Redis для кешування та швидкого доступу.

**Діаграма мікросервісів**

Изображение выглядит как диаграмма, План, текст, Технический чертеж

Автоматически созданное описание

**Пояснення декомпозиції**

**Розподіл відповідальностей:**  
Кожен мікросервіс відповідає за виконання конкретного завдання, наприклад, обробку замовлень, управління платежами чи збирання рейтингів.

**Взаємодія між сервісами:**  
Для запитів, які потребують миттєвого відгуку, доцільно використовувати REST API або gRPC. Асинхронна передача подій, наприклад, створення нового замовлення, реалізується через Message Broker.

**Зберігання даних:**  
Спеціалізовані бази даних обрані для забезпечення максимальної продуктивності та можливості масштабування системи.

**Масштабованість:**  
Мікросервіси можна масштабувати окремо, залежно від рівня їх завантаження, що забезпечує ефективне використання ресурсів.

**Висновок**

Декомпозиція системи на мікросервіси дозволяє створити гнучку, продуктивну та масштабовану архітектуру. Розподіл відповідальностей між сервісами забезпечує простоту розробки, тестування та обслуговування кожного окремого компонента. Використання сучасних засобів взаємодії, таких як REST API, gRPC і Message Broker, забезпечує надійну та ефективну комунікацію між сервісами.

Спеціалізовані бази даних оптимізують роботу з даними, а незалежне масштабування кожного мікросервісу дозволяє ефективно адаптувати систему до змін у навантаженні. Такий підхід сприяє створенню стійкої до збоїв і легко розширюваної інфраструктури, що відповідає сучасним вимогам складних інформаційних систем.