Міністерство освіти і науки України

Національний технічний університет України «КПІ ім. Ігоря Сікорського»

Факультет інформатики та обчислювальної техніки

Кафедра інформатики та програмної інженерії

Проектування розподілених систем

**ЗВІТ**

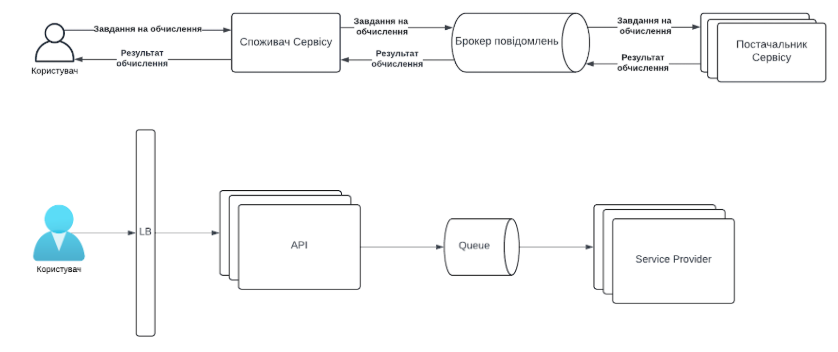
до лабораторної роботи №2

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Виконали**  **студенти** |  | ІП-31мн Кобельський Вадим |  |  |
|  |  | (№ групи, прізвище, ім’я, по батькові ) |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  | (№ групи, прізвище, ім’я, по батькові ) |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  | (№ групи, прізвище, ім’я, по батькові ) |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
| **Прийняв** |  | Обозний Д. М. |  |  |
|  |  | (посада, прізвище, ім’я, по батькові ) |  |  |

Київ 2024

**Завдання**

* Реалізувати асинхронну комунікацію між Постачальником Сервісу і Споживачем Сервісу за допомогою Брокера Повідомлень
* Постачальник Сервісу має підраховувати час обчислення і логувати його для подальшого аналізу
* Cпоживач Cервісу має підраховувати час виконання запиту і логувати його для подальшого аналізу
* Реалізувати горизонтальне масштабування засобами Брокера Повідомлень
* Реалізувати чергу с пріоритетами
* Реалізувати request-reply паттерн в асинхронній комунікації
* Порівняти результати синхронної і асинхронної комунікації



Хід роботи

Споживача сервісу було реалізовано за допомогою бібліотеки FastAPI а постачальник сервісу піднятий за допомогою асинхронного потоку щоб постійно працювати та очікувати на повідомлення з черги. Для черги було обрано RabbitMQ тому що вона “з коробки” надає чергу з пріоритетами тому не потрібно буде реалізовувати її власноруч. Завдання на виконання залишилось те ж саме, а саме множення отриманого числа на 2, щоб перевірити час виконання запитів при синхронній та асинхронній комунікації.

На рисунках 1-2 можна побачити виконання поставлених завдань.

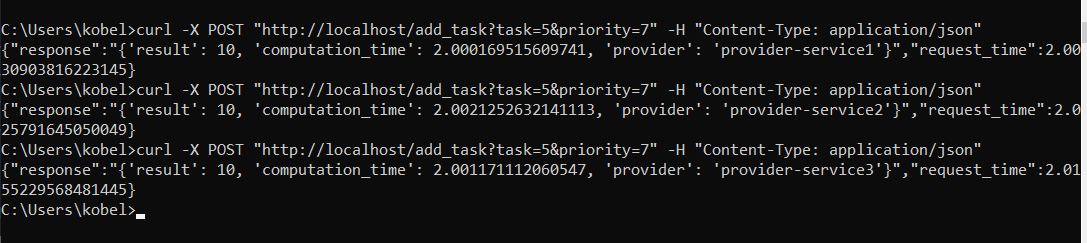


Рисунок 1 – виконання запитів до застосунку

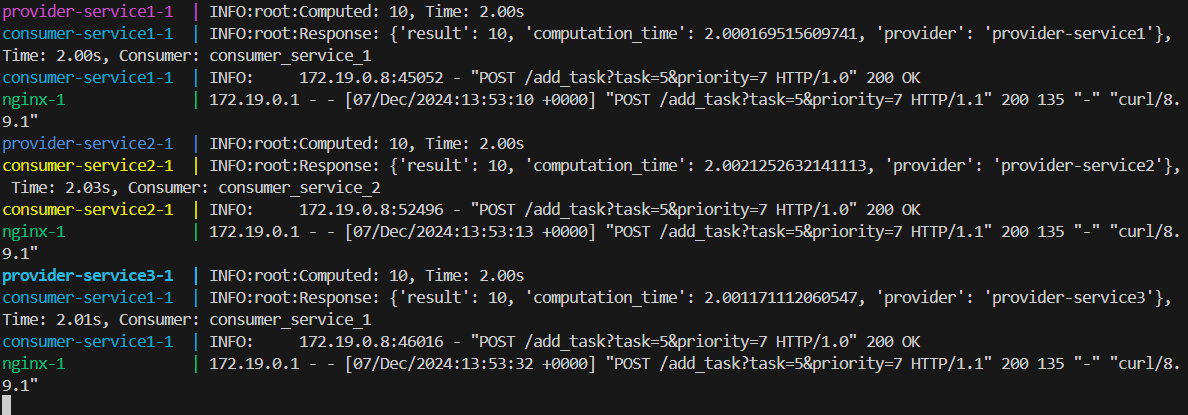


Рисунок 2 – Логи застосунку які демонструють процес Load Balancing

З рисунку 2 можна побачити що споживачів сервісу 2, а постачальників сервісу 3. З рисунку 1 можна побачити використання пріоритету для його подальшого використання в черзі для даного повідомлення.

З рисунку 1 можна побачити що середній час виконання запиту, використовуючи асинхронний підхід в комунікації між споживачем та постачальником сервісу, приблизно 2 секунди, що є більшим ніж при використанні синхронної комунікації, але цей результат отриманий при виконанні дуже простого завдання, яке не потребує використання асинхронної комунікації, але якщо збільшити кількість запитів та час виконання самого завдання то в такому випадку асинхронна комунікація буде в усіх аспектах перемагати синхронну комунікацію.

Код застосунку доступний за посиланням: