

ДОСЛІДЖЕННЯ МЕТОДІВ ПРОГРАМНОЇ РЕАЛІЗАЦІЇ COSMOS DB API НА ПЛАТФОРМІ .NET

Андрущенко М.О.

Науковий керівник – к.т.н., доцент Мазурова О.О.

Харківський національний університет радіоелектроніки
61166, Харків, просп. Науки, 14, каф. Програмної інженерії
тел.: (057) 702-13-06, e-Mail: mykola.andrushchenko@nure.ua

This work is devoted to the study of various methods for implementing the Cosmos DB API using .NET. The most appropriate APIs for implementing software using .NET have been selected. An analysis of the subject area of e-commerce was carried out, a database scheme was developed. Queries were created and metrics selected to measure the performance of each of the implementations. Developed software solutions for API for NoSQL (using Cosmos Client and Entity Framework Core Cosmos) and API for MongoDB (using Mongo Client). Experiments were carried out, performance measurements were obtained on selected metrics, and recommendations were formulated for the practical use of the API.

Сучасний підхід до розробки програмних систем спрямований на те, щоб спростити програму, зробити код дешевшим у підтримці в майбутньому. Саме тому сьогодні популярні хмарні розподілені системи, що створюються за мікросервісною архітектурою.

В системах з мікросервісною архітектурою класичні реляційні бази даних (БД) достатньо рідко. Активно використовують NoSQL БД, які дозволяють розробникам легко змінювати структуру даних [1]. Крім того, NoSQL БД легко масштабуються, тобто відповідні додатки можуть обробляти великі обсяги даних [2]. Однією з популярних NoSQL БД, що використовується при розробці хмарних розподілених систем є Cosmos DB. Під час реалізації програмних додатків на базі цієї СУБД важливо обрати такий API, який буде найбільш ефективним.

Отже, була поставлена задача дослідити методи програмної реалізації Cosmos DB API на платформі .NET та на основі експериментальних даних розробити рекомендації щодо їх практичного використання.

Було проведено аналіз існуючих Cosmos DB API, таких як for NoSQL, for MongoDB, for Apache Cassandra, for Table, for Apache Gremlin та for PostgreSQL. Для подальшого дослідження було обрано найбільш перспективні, а саме API for NoSQL (розглянуто варіанти з використанням Cosmos Client та Entity Framework Core Cosmos) та API for MongoDB (використовуючи Mongo Client).

В якості предметної області для проведення дослідження була обрана область електронної комерції, як достатньо популярна на сьогодні

сфера. Була спроектована база даних, що містить інформацію про такі сутності, як товар, замовлення, позиція в замовленні та платіж.

Було проведено планування експериментальної частини дослідження:

- розроблено серію запитів до баз даних, які будуть досліджуватися, що містять запити на отримання товарів та замовлень та CRUD операції;
- обрано метрики для оцінки продуктивності API, а саме швидкість виконання запитів (мс) та кількість ресурсів необхідних для цього (RU).

Для проведення експериментів було налаштовано середовище Azure для роботи з Cosmos DB. Були створені та налаштовані всі необхідні ресурси, в тому числі, Azure Cosmos DB Account, бази даних та контейнери.

Було реалізовано три програми, по одній для кожного з обраних підходів, за допомогою таких технологій, як .NET 6.0, Azure Cosmos DB, Cosmos Client, Entity Framework Core Cosmos, Mongo Client, Swagger. Кожна програма створює базу даних та всі необхідні контейнери при першому запуску, що є важливим в тих ситуаціях, коли база даних вперше розгортається в новому регіоні або у неналаштованому середовищі.

Було проведено експериментальні дослідження, під час якого на кожному з розроблених додатків виконувалися запити до БД та проводилися заміри обраних метрик.

Отримані результати дозволили зробити висновки та сформулювати рекомендації, щодо використання досліджених API. Так, API for NoSQL з Entity Framework Core Cosmos більш доцільно використовувати, якщо вам необхідні переваги роботи з Entity Framework, а саме Code First підхід та LINQ для формування запитів до БД. Розробка з використанням API for NoSQL з Cosmos Client буде більш ефективною, якщо необхідна підтримка SQL подібних запитів. Розробка з використанням API for NoSQL буде більш ефективною, якщо необхідно працювати з MongoDB.

Список використаних джерел:

1. Sahatqija K., Ajdari J., Zenuni X. Comparison between relational and NOSQL databases. 41st International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO). 2018. P. 216-221.

2. Meier A. SQL & NoSQL Databases: Models, Languages, Consistency Options and Architectures for Big Data Management / Andreas Meier, Michael Kaufmann. – Springer Vieweg, 2019. – 248 P. – ISBN 978-3658245481.