

## **СУЧАСНІ ПІДХОДИ ПРИ РОЗРОБЦІ АРХІТЕКТУРИ РОЗПОДІЛЕНИХ СИСТЕМ**

Вінницький національний технічний університет

### **Анотація**

*В роботі проведено аналіз сучасних підходів при розробці архітектури розподілених систем. Розглянуто основні компоненти, методи комунікації та інструменти для забезпечення ефективності та надійності розподіленої системи. Показано важливість масштабованості та безпеки в рамках таких архітектурних рішень.*

**Ключові слова:** розподілена система, клієнт-серверна взаємодія, масштабованість, REST API, безпека в розподілених системах.

### **Abstract**

*The paper analyzes modern approaches in developing the architecture of distributed systems. The main components, communication methods and tools for ensuring the efficiency and reliability of a distributed system are considered. The importance of scalability and security within such architectural solutions is shown.*

**Keywords:** distributed system, client-server interaction, scalability, REST API, security in distributed systems.

### **Вступ**

Інформаційні системи в більшості випадків є розподіленими і являють собою комплекс взаємопов'язаних компонентів, що працюють разом на різних фізичних машинах, мережах чи просторі, спільно виконуючи задачі. Розподілена архітектура, з свого боку, визначає структуру цих компонентів, зв'язки між ними та правила спільної роботи. Тобто, розподілена архітектура визначає організацію розподіленої системи, включаючи в себе протоколи комунікації, розміщення компонентів, стратегії управління даними та ресурсами. Це є фундаментом, на якому ґрунтується ефективна робота розподіленої системи.

В даній роботі розглядаються сучасні підходи при розробці розподіленої архітектури, включаючи клієнт-серверну взаємодію, масштабованість, використання REST API та забезпечення безпеки в розподілених системах. Мета роботи полягає в аналізі підходів для розробки розподіленої архітектури, зокрема створення ефективної та надійної системи, яка може працювати в розподіленому середовищі. Розуміння та ефективне впровадження цих концепцій має велике значення для розробників та архітекторів програмного забезпечення, оскільки дозволяє створити продуктивні та масштабовані системи, що відповідають сучасним вимогам інформаційного ринку.

### **Проектування розподіленої архітектури**

Розподілена архітектура є досить важливою при проектуванні сучасних програмних засобів, який дозволяє розробникам створювати високоефективні та надійні програмні системи з можливістю горизонтального та вертикального масштабування. Цей процес вимагає уважного аналізу та проектування різних аспектів системи, зокрема комунікаційні протоколи, архітектурні рішення та засоби забезпечення безпеки.

Ключовим аспектом в проектуванні розподіленої архітектури є розробка чіткого та ефективного механізму клієнт-серверної взаємодії [1]. Для цього використовуються різні протоколи та методи, такі як HTTP, WebSocket, а також RPC (Remote Procedure Call), які дозволяють клієнту та серверу обмінюватися даними та запитами.

Іншим важливим аспектом є масштабованість системи. Розподілені системи повинні бути здатними розширюватися зі зростанням завантаження, що досягається за допомогою розподіленого обчислення, кешування, а також використання сучасних інфраструктурних рішень, таких як контейнери [2].

Важливою проблемою в розподілених системах є забезпечення їх безпеки. Розподілені системи відкриті для різних видів атак, тому необхідно використовувати механізми аутентифікації, авторизації та шифрування даних для забезпечення конфіденційності та цілісності інформації.

Окрім того, основним моментом при проектуванні є вибір відповідної архітектурної моделі, яка може бути реалізована як мікросервісна архітектура, клієнт-серверна або інші моделі, які найкраще відповідають потребам проекту.

Схематичний приклад клієнт-серверної взаємодії в розподіленій архітектурі наведений на рисунку 1:

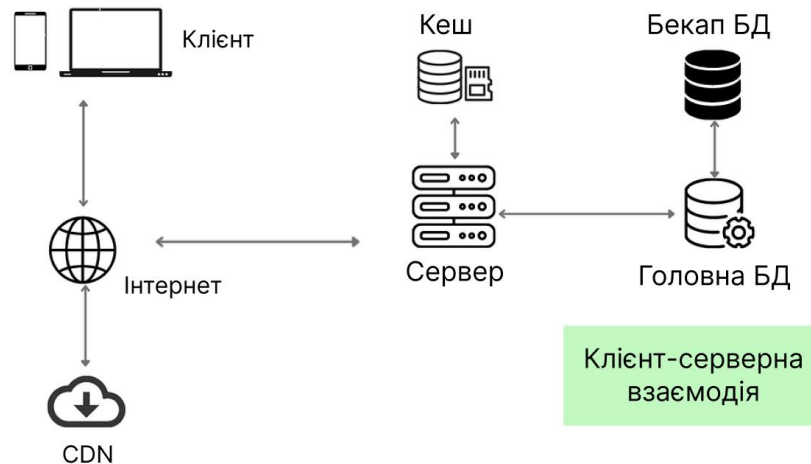


Рисунок 1 — Візуальне представлення клієнт-серверної взаємодії в розподіленій архітектурі

З рисунку 1 видно, що всі компоненти у взаємодії клієнта та сервера ізолюються один від одного та можуть бути замінені іншими. Наведена архітектура може збільшувати своє продуктивне навантаження, додаючи нові ресурси, такі як сервери, обчислювальні потужності, бази даних чи мережеві з'єднання, тобто забезпечує масштабованість проекту, що є досить важливим для забезпечення високої доступності та продуктивності системи, особливо в умовах зростаючого обсягу користувачів або даних.

Всі аспекти розподіленої архітектури вимагають великої уваги та аналізу при проектуванні програмних систем. Правильно спроектована розподілена архітектура може значно покращити ефективність та надійність системи, що є важливим аспектом у сучасному програмному проектуванні.

### Висновки

Розподілена архітектура між клієнтом та сервером є критично важливим аспектом сучасної розробки програмного забезпечення. В роботі були розглянуті ключові аспекти її проектування, включаючи клієнт-серверну взаємодію, масштабованість, використання REST API та забезпечення безпеки.

Проектування розподіленої архітектури вимагає глибокого розуміння різних компонентів системи та їх взаємодії. Важливість чіткого механізму комунікації між клієнтом та сервером не може бути недооціненою, оскільки від цього залежить швидкодія та надійність системи.

Масштабованість є ще однією важливою характеристикою розподіленої системи. Здатність системи розширюватися зі зростанням навантаження дозволяє забезпечити високу доступність та продуктивність.

Забезпечення безпеки в розподілених системах вимагає використання різних методів та механізмів для захисту даних та забезпечення конфіденційності.

Отже, розподілена архітектура є ключовим фактором для успішної розробки програмного забезпечення. Її проектування вимагає уважного аналізу та розуміння різних аспектів, а правильно спроектована архітектура може значно покращити ефективність та надійність системи.

### СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Клієнт-серверна взаємодія [Електронний ресурс] – Режим доступу до ресурсу: <https://training.qatestlab.com/blog/technical-articles/client-server-architecture/>

2. Client-Server Model [Електронний ресурс] – Режим доступу до ресурсу:  
<https://www.geeksforgeeks.org/client-server-model/>

**Никитюк В'ячеслав Олександрович** – студент групи 1КІ-22М, факультет інформаційних технологій та комп'ютерної інженерії, Вінницький національний технічний університет, Вінниця, e-mail: [slavik.nukutuk@gmail.com](mailto:slavik.nukutuk@gmail.com)

**Городецька Оксана Степанівна** – кандидат технічних наук, доцент кафедри обчислювальної техніки, Вінницький національний технічний університет, Вінниця, e-mail: [horodecka.os@gmail.com](mailto:horodecka.os@gmail.com)

**Войцеховська Олена Валеріївна** – кандидат технічних наук, доцент кафедри обчислювальної техніки, Вінницький національний технічний університет, Вінниця.

**Nykytiuk Viacheslav O.** – students, KI-20MS, Faculty of information Technologies and Computer Engineering, Vinnytsa National Technical University, email: [slavik.nukutuk@gmail.com](mailto:slavik.nukutuk@gmail.com)

**Horodetska Oksana S.** — PhD, Faculty of Information Technologies and Computer Engineering, Vinnytsia National Technical University, email: [horodecka.os@gmail.com](mailto:horodecka.os@gmail.com)

**Voytsekhovska Olena V.** — PhD, Faculty of Information Technologies and Computer Engineering, Vinnytsia National Technical University.