

## SECTION 10.

### COMPUTER AND SOFTWARE ENGINEERING

---

**Рубленко Маргарита Віталіївна**

Здобувачка вищої освіти факультету інформаційних технологій  
*Державний торговельно-економічний університет, Україна*

**Науковий керівник: Юрченко Юрій Юрійович**

Старший викладач, кафедра комп'ютерних наук та інформаційних систем  
Факультет інформаційних технологій  
*Державний торговельно-економічний університет, Україна*

---

## РІЗНИЦЯ МІЖ ASP.NET CORE ТА ASP.NET FRAMEWORK

ASP.NET Framework та ASP.NET Core є двома різними підходами до розробки веб-застосунків у середовищі .NET, які відображають еволюцію технологій Microsoft у цьому напрямі.

ASP.NET Framework є класичною платформою для створення веб-додатків, що працює виключно на операційній системі Windows і тісно інтегрована з її компонентами, такими як IIS. Вона включає в себе широкий набір готових інструментів і бібліотек, що робить її зручною для швидкої розробки, однак водночас обмежує гнучкість і масштабованість застосунків [1, 4]. ASP.NET Core, у свою чергу, є сучасною кросплатформною версією, яка була створена з урахуванням нових вимог до продуктивності, модульності та розгортання. Вона може працювати не лише на Windows, але й на Linux та macOS, що значно розширює можливості її використання [2]. Завдяки цьому ASP.NET Core активно застосовується у хмарних середовищах і контейнеризованих системах, зокрема з використанням Docker.

Однією з ключових відмінностей між цими двома технологіями є їхня архітектура. ASP.NET Framework має монолітну структуру, де більшість компонентів є частиною єдиної системи. Це означає, що розробник змушений використовувати значну частину функціоналу навіть тоді, коли вона не потрібна. Натомість ASP.NET Core побудована за принципом модульності, що дозволяє підключати лише ті компоненти, які необхідні для конкретного застосунку. Такий підхід реалізує концепцію «pay-for-what-you-use», що позитивно впливає на продуктивність і спрощує підтримку [2, 3].

Ще однією важливою відмінністю є продуктивність. ASP.NET Core

оптимізована для високонавантажених систем і демонструє значно кращі показники швидкодії порівняно з ASP.NET Framework. Це досягається завдяки новому веб-серверу Kestrel, асинхронній обробці запитів та ефективнішому використанню ресурсів системи [1]. У той час як ASP.NET Framework значною мірою залежить від IIS, ASP.NET Core може працювати як самостійний сервер або в поєднанні з іншими веб-серверами.

Суттєві зміни також торкнулись підходу до конфігурації застосунків. У ASP.NET Framework налаштування зберігаються переважно у файлах `web.config`, які мають складну структуру та менш гнучкі у використанні. У ASP.NET Core використовується спрощена система конфігурації, яка підтримує різні формати, такі як JSON, змінні середовища або командний рядок [3]. Це дозволяє легше адаптувати застосунок до різних середовищ виконання.

Важливою перевагою ASP.NET Core є також вбудована підтримка Dependency Injection, яка є частиною ядра фреймворку. У ASP.NET Framework цей механізм відсутній за замовчуванням і потребує використання сторонніх бібліотек. Ще одним аспектом є підтримка сучасних підходів до розробки, таких як створення REST API, мікросервісна архітектура та SPA-застосунки. ASP.NET Core орієнтована саме на ці сценарії і забезпечує всі необхідні інструменти для їх реалізації. У той час як ASP.NET Framework більше підходить для підтримки існуючих проєктів або систем, які вже побудовані на цій технології [4].

З точки зору розгортання, ASP.NET Core пропонує значно більшу гнучкість. Застосунки можуть бути самодостатніми, тобто включати всі необхідні залежності, або використовувати спільне середовище виконання. Це спрощує процес деплоюменту і дозволяє уникнути проблем із сумісністю версій. ASP.NET Framework, навпаки, потребує встановленого середовища .NET Framework на сервері, що може створювати додаткові труднощі.

Таким чином, ASP.NET Core є більш сучасною, гнучкою та продуктивною платформою, яка відповідає вимогам сучасної веб-розробки. Вона значно перевершує ASP.NET Framework за багатьма параметрами, включаючи кросплатформність, продуктивність і архітектурну гнучкість.

#### Список використаних джерел:

1. Microsoft. ASP.NET Core documentation [Електронний ресурс]. – Режим доступу: <https://learn.microsoft.com/aspnet/core>
2. Price M. *C# 10 and .NET 6 – Modern Cross-Platform Development*. – Birmingham: Packt Publishing, 2022. – 820 p.
3. Freeman A. *Pro ASP.NET Core 6*. – 9th ed. – New York: Apress, 2022. – 1000 p.
4. Esposito D. *Modern Web Development with ASP.NET Core 3*. – Redmond: Microsoft Press, 2020. – 320 p.