

Міністерство освіти і науки України
Національний технічний університет України «Київський політехнічний
інститут імені Ігоря Сікорського"
Факультет інформатики та обчислювальної техніки

Кафедра інформатики та програмної інженерії

Звіт

з лабораторної роботи № 3 з дисципліни
«Сучасні технології розробки WEB-застосунків на платформі .NET (2
частина)»

**«Проектування та створення веб-застосунку в межах багаторівневої
архітектури»**

Виконав(ла)

ІП-14 Хільчук Артем Валерійович
(шифр, прізвище, ім'я, по батькові)

Перевірив

Вовк Євгеній Андрійович
(прізвище, ім'я, по батькові)

Київ 2024

ЗМІСТ

1	ЗАВДАННЯ	3
2	ВИКОНАННЯ.....	4
	ВИСНОВОК	9

1 ЗАВДАННЯ

1. У відповідності до наданих варіантів виконати тестування програмного засобу.

2. Виконати проектування архітектури програмного засобу в вигляді розподіленого моноліту або DDD, з використанням onion або clean або CQRS. Важливо при задачах фільтрації використовувати бібліотеку Ardalis.Specification, або альтернативи для побудови дерева виразів та його огортання у відповідний об'єкт.

3. Виконати моделювання системи шляхом використання C4 Model Diagram.

4. Виконати моделювання описом UML component, deployment diagram.

5. Ключові архітектурні рішення задокументувати у вигляді architectural decision record (ADR).

6. Забезпечити працездатність тестів та гарантувати що зміна архітектурного підходу не зашкодила функціональності продукту. Варіант завдання

1. Персональний блог. Авторизація/Аутентифікація: Можливість зареєструватися/увійти як автор. Створення та редагування постів: Автор може створювати, редагувати та видаляти свої пости. Перегляд постів: Відвідувачі можуть переглядати пости без реєстрації. Коментування: Відвідувачі можуть залишати коментарі під постами.

2 ВИКОНАННЯ

2.1 У відповідності до наданих варіантів виконати тестування програмного засобу.

У рамках виконання попередньої лабораторної роботи було реалізовано низку тестів – усі вони виконалися без будь-яких проблем.

2.2 Виконати проектування архітектури програмного засобу в вигляді розподіленого моноліту або DDD, з використанням з використанням onion або clean або CQRS. Важливо при задачах фільтрації використовувати бібліотеку Ardalis.Specification, або альтернативи для побудови дерева виразів та його огортання у відповідний об'єкт

У рамках даної лабораторної було вирішено використати Onion архітектуру, котра забезпечить щонайкраще зменшення зв'язності між компонентами та утримуваність проекту, з використанням CQRS, аби забезпечити найбільшого дотримання принципу єдиного призначення при розвитку застосунку.

Крім цього, буде використано низку додаткових зовнішніх пакетів. Одним з таких є Ardalis.Specification, що надає зручний інтерфейс для задання специфікацій. Іншим таким пакетом є MediatR – він реалізує патерн посередника та становить зручний засіб для управління запитами та їхніми обробниками в рамках використання патерну CQRS.

Змоделюймо отриману систему.

2.3 Виконати моделювання системи шляхом використання C4 Model Diagram.

Контекстний рівень:

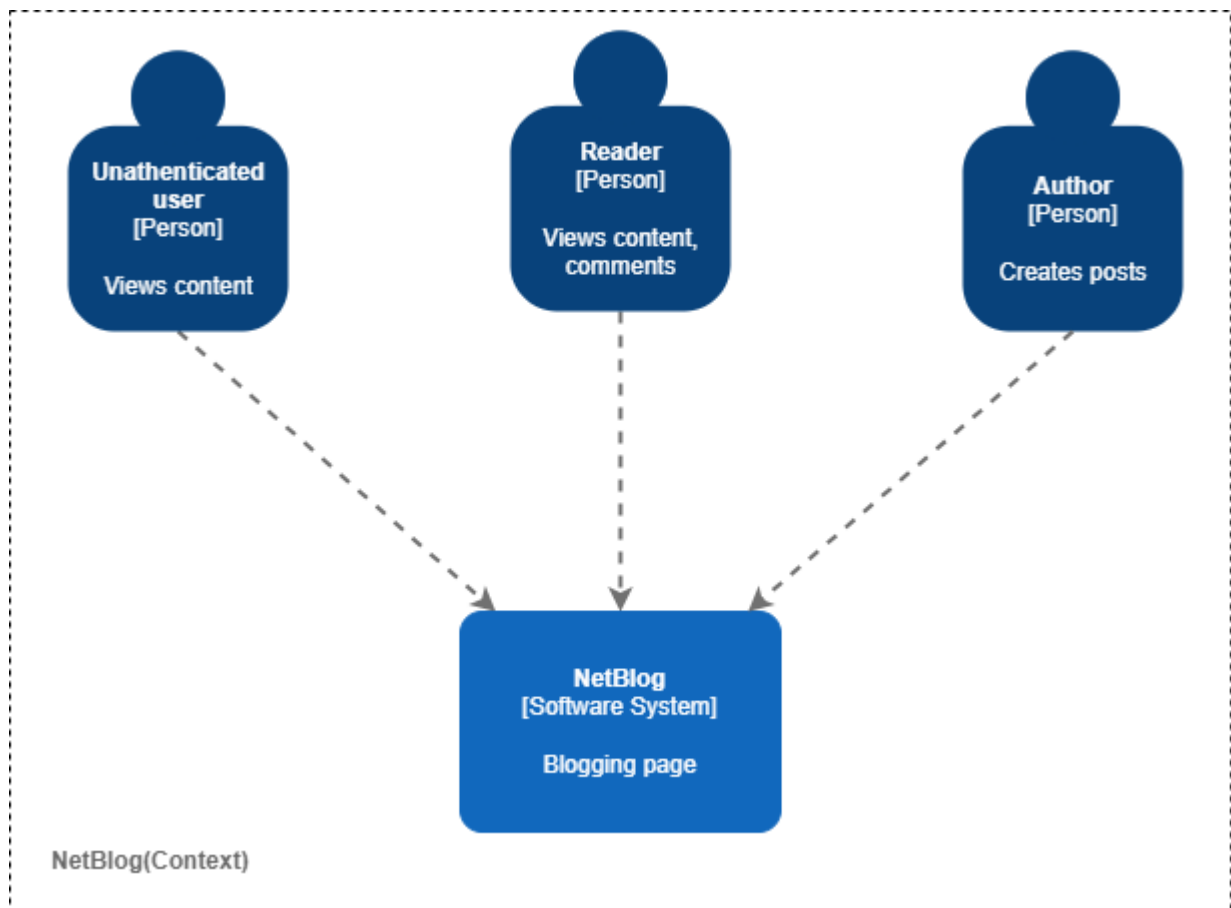


Рисунок 2.3.1 – контекстний рівень діаграми

Контейнерний рівень:

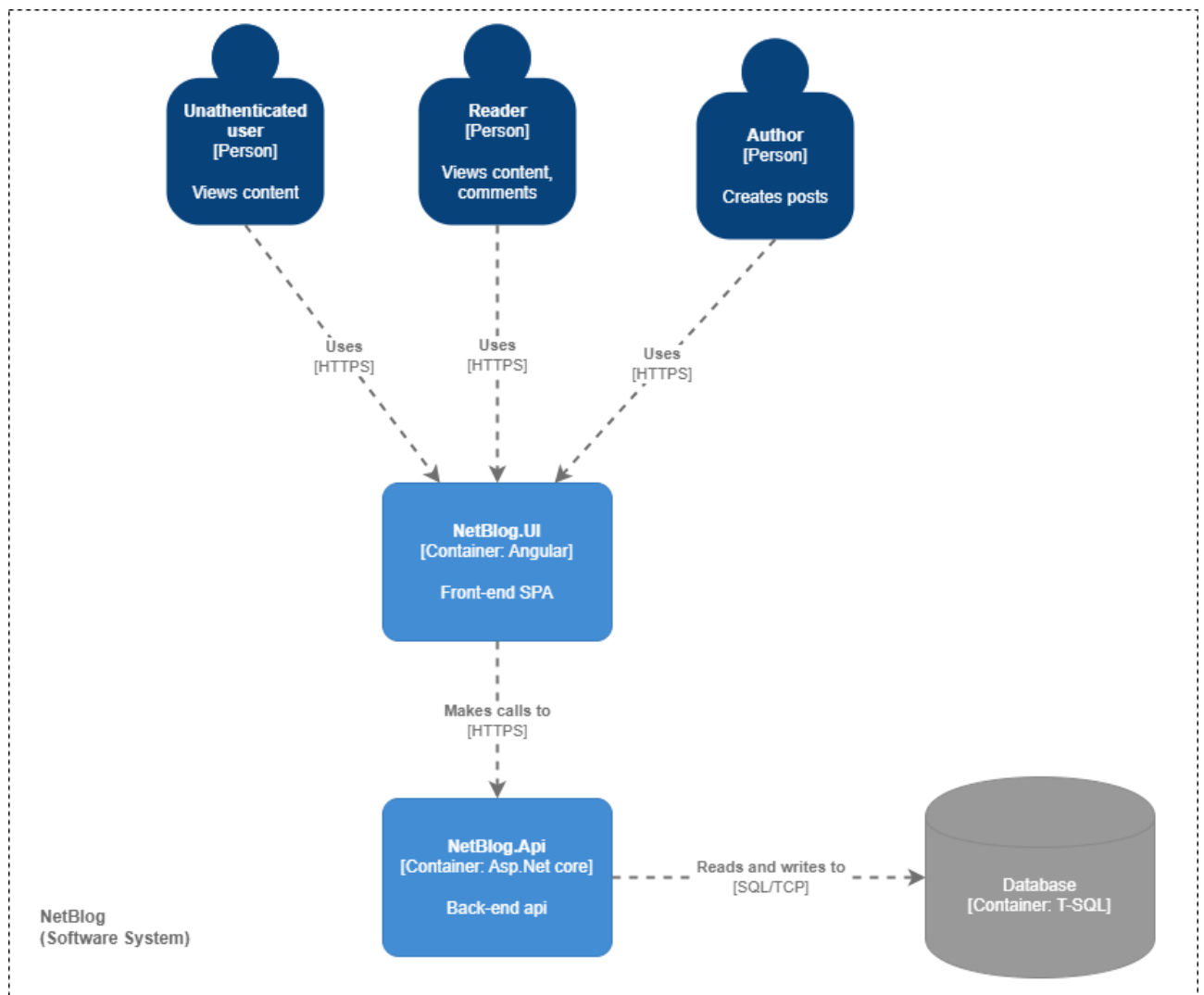


Рисунок 2.3.2 – контейнерний рівень діаграми

Компонентний рівень:

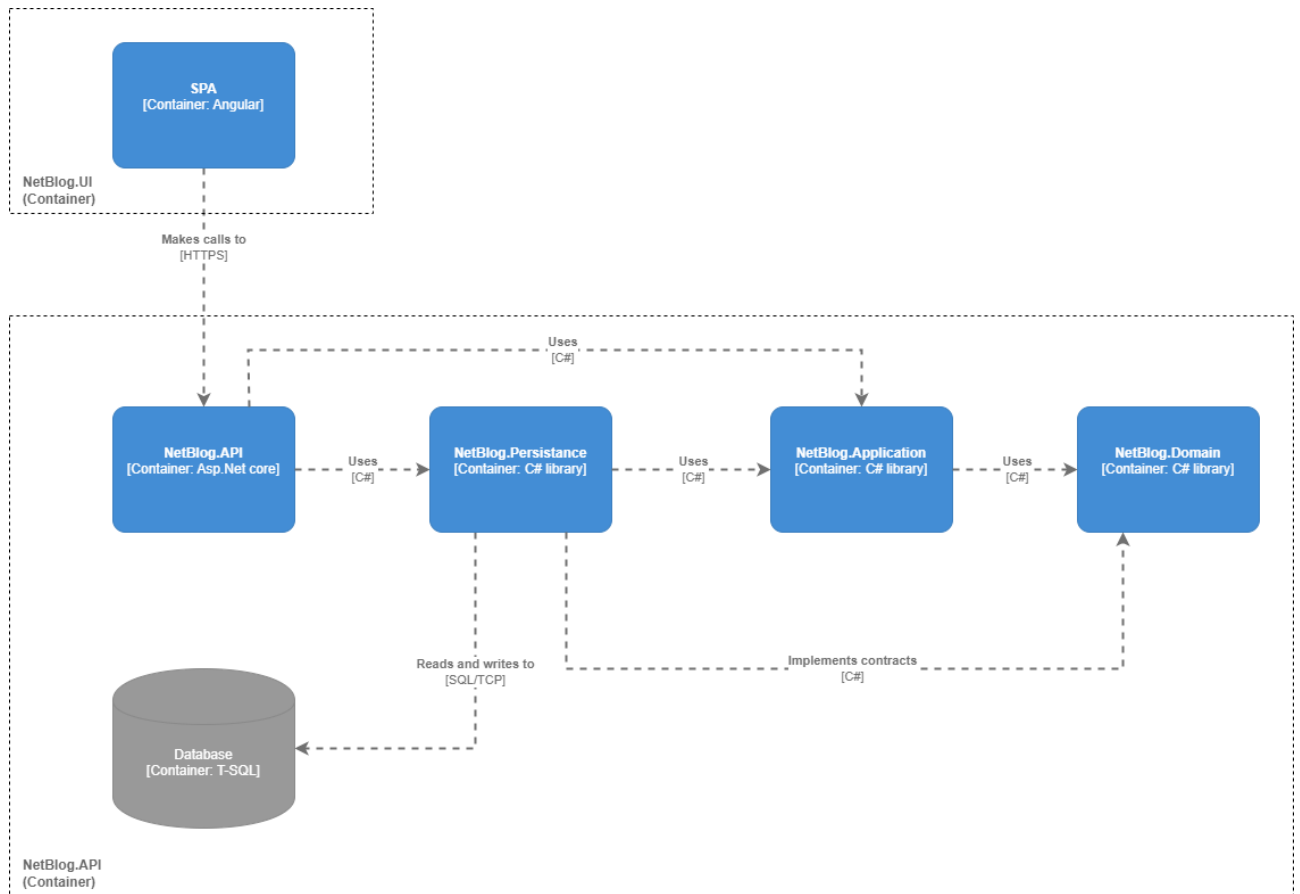


Рисунок 2.3.3 – компонентний рівень діаграми

Класовий рівень:

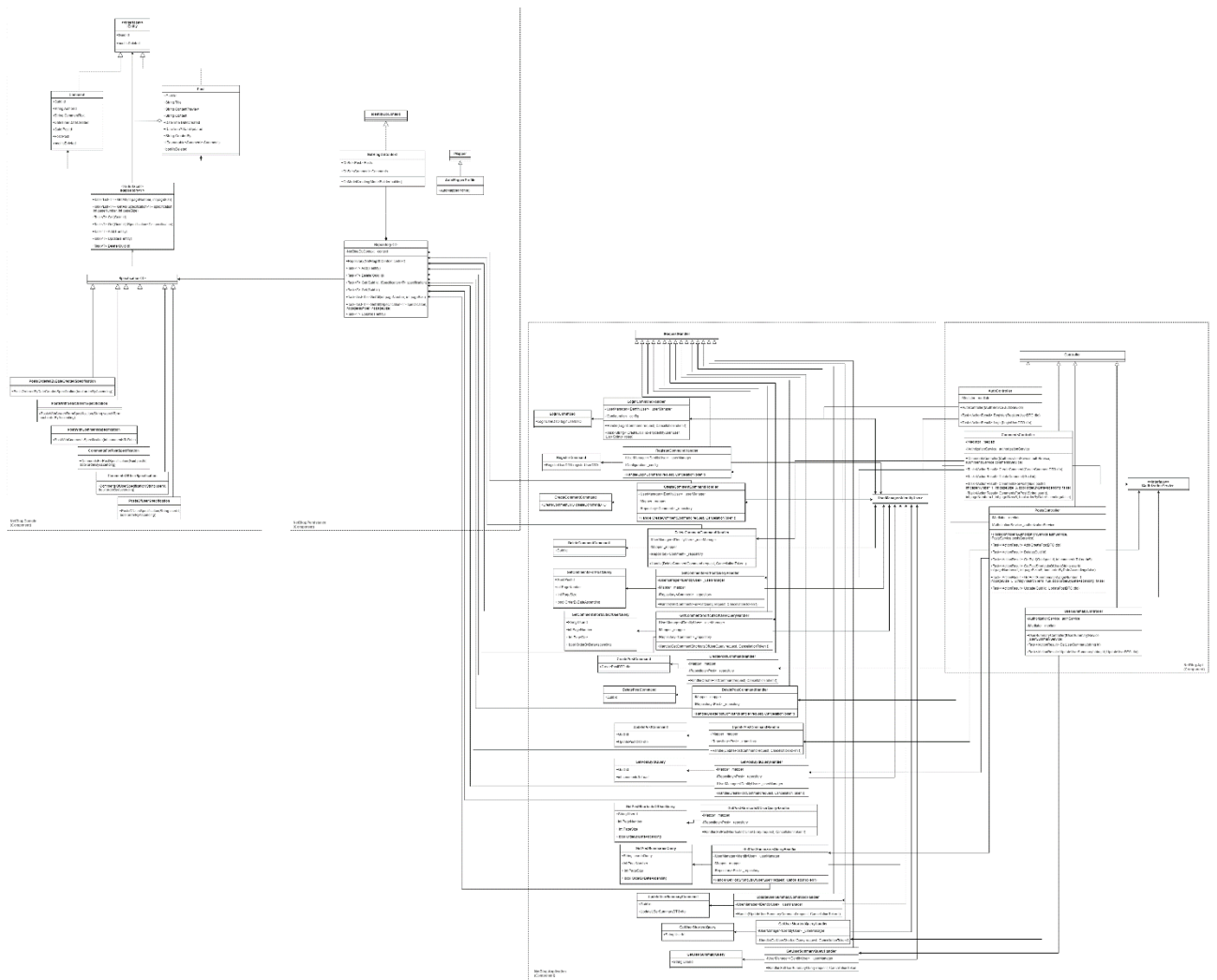


Рисунок 2.3.4 – класовий рівень діаграми

2.4 Ключові архітектурні рішення задокументувати у вигляді architectural decision record (ADR).

ВИСНОВОК

Отож, у ході виконання лабораторної роботи було проведено контроль якості розробленого застосунку шляхом реалізації та виконання низки тестів: юніт-тестів, інтеграційних та End-to-End. Під час виконання роботи було обрано конкретні види тестування і їхні допоміжні засоби для різних категорій складових додатку та обгрунтовано доцільність їхнього використання. Урешті-решт, було виконано розроблені тести та проаналізовано покриття додатку юніт тестами за допомогою засобу Coverlet: усі тести успішно пройдено, а аналіз продемонстрував покриття 95% коду застосунку. Набуто практичних навичок проведення тестування додатків за допомогою ASP.Net core, а також пакетів xUnit, Moq та інших.