

УДК 004.42

Чигрин М. – ст. гр. СП-42

*Тернопільський національний технічний університет імені Івана Пулюя*

## **АРХІТЕКТУРНІ ПІДХОДИ ДО ПОБУДОВИ ВЕБ-СИСТЕМИ УПРАВЛІННЯ НАВЧАЛЬНИМ КОНТЕНТОМ**

Науковий керівник: кандидат технічних наук, доцент Михалик Д.М.

Chyhryn M.

*Ternopil Ivan Puluj National Technical University*

## **ARCHITECTURAL APPROACHES TO BUILDING A WEB-BASED LEARNING CONTENT MANAGEMENT SYSTEM**

Supervisor: PhD, Associate Professor Mykhalyk D.M.

Ключові слова: веб-система, навчальний контент, REST API, модульна архітектура  
Keywords: web system, learning content, REST API, modular architecture

Сучасний розвиток дистанційної освіти супроводжується зростанням вимог до гнучкості, масштабованості та надійності платформ управління навчальним контентом. Більшість існуючих LMS-рішень (Moodle, Canvas) є монолітними та складними в адаптації до специфічних потреб освітніх установ, що зумовлює актуальність розробки власних архітектурно обґрунтованих систем.

Метою роботи є проектування та реалізація веб-системи управління навчальним контентом з чіткою модульною структурою та розмежуванням ролей користувачів: студент, викладач та адміністратор.

В основу архітектури системи покладено принцип розділення відповідальностей між компонентами – підхід, що забезпечує незалежність бізнес-логіки від інфраструктурних деталей та спрощує підтримку і розширення системи [1]. Систему розділено на незалежні модулі: навчальний контент, домашні завдання, тестування, управління користувачами та психологічна підтримка студентів. Надійність та масштабованість хмарних інформаційних систем досягається шляхом чіткого розмежування архітектурних рівнів і рольового доступу до даних [2].

Для реалізації програмного інтерфейсу застосовано фреймворк Django Ninja, який забезпечує автоматичну генерацію OpenAPI-документації та типізацію запитів і відповідей на основі анотацій типів [3]. Автентифікацію реалізовано через JWT-токени, що дозволяє будувати stateless-взаємодію між клієнтом та сервером відповідно до принципів REST-архітектури [4]. Для підвищення продуктивності застосовано кешування на основі Redis, а зберігання медіафайлів організовано у хмарному S3-сховищі.

Діаграма розгортання (див. рис. 1) ілюструє інфраструктурну складову системи, яка базується на контейнеризації за допомогою Docker. Nginx виконує роль реверс-проксі та балансувальника навантаження, Django з Gunicorn обробляють запити REST API, PostgreSQL забезпечує персистентне зберігання даних, а Redis використовується для кешування часто запитуваних ресурсів.

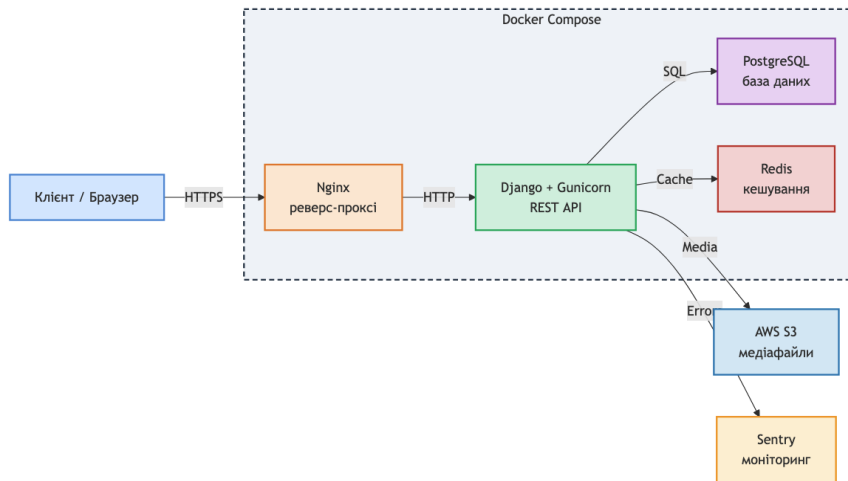


Рисунок 1 – Діаграма розгортання веб-системи управління навчальним контентом

Для забезпечення якості програмного коду налаштовано автоматизоване тестування з покриттям понад 80% за допомогою `pytest`, а моніторинг помилок у продакшн-середовищі здійснюється через Sentry. Застосування модульної архітектури у поєднанні з сучасним API-фреймворком, хмарною інфраструктурою та автоматизованим тестуванням дозволяє отримати масштабовану, надійну та легко підтримувану систему управління навчальним контентом.

Подальші дослідження можуть бути спрямовані на впровадження рекомендаційних алгоритмів для персоналізації навчального процесу та розширення модуля психологічної підтримки студентів.

#### Література:

1. Martin R. C. Clean Architecture: A Craftsman's Guide to Software Structure and Design. Boston: Prentice Hall, 2017. 432 p.
2. Semchyshyn V., Mykhalyk D. Data-driven decision-making methods and hierarchical analysis in cloud-based medical service management systems. ITTAP-2025: 5th International Workshop on Information Technologies: Theoretical and Applied Problems. Ternopil, 2025. P. 1–9.
3. Django Ninja – Fast Django REST Framework. – Електронний ресурс. – Режим доступу: <https://django-ninja.dev> (дата звернення: 20.04.2026)
4. Fielding R. T. Architectural Styles and the Design of Network-Based Software Architectures: Doctoral dissertation. University of California, Irvine, 2000. 162 p.