

# СОЦІАЛЬНІ МЕРЕЖІ ЯК ОБ'ЄКТ ПРОГРАМНОЇ ІНЖЕНЕРІЇ

Вінницький національний технічний університет

## Анотація

*У роботі розглядаються архітектурні особливості та життєвий цикл розробки (SDLC) соціальних мереж з позиції програмної інженерії. Наведено приклади використання UML-моделювання, мікросервісної архітектури та гнучких методологій для проєктування високонавантажених систем. Показано значення проєктного менеджменту для створення та підтримки масштабованих платформ.*

**Ключові слова:** програмна інженерія, соціальні мережі, архітектура програмного забезпечення, UML-діаграми, мікросервіси, життєвий цикл, управління проєктами.

## Abstract

*The paper considers the architectural features and software development life cycle (SDLC) of social networks from the perspective of software engineering. Examples of using UML modeling, microservice architecture, and agile methodologies for designing high-load systems are presented. The importance of project management for creating and maintaining scalable platforms is shown.*

**Key words:** software engineering, social networks, software architecture, UML diagrams, microservices, software development life cycle, project management.

Розробка сучасних соціальних мереж є одним із найбільш трудомістких завдань у галузі програмної інженерії через необхідність обробки великих масивів даних у реальному часі та забезпечення безперебійного доступу мільйонів користувачів. Як об'єкт програмної інженерії, соціальна мережа вимагає системного підходу вже на етапі збору та аналізу вимог. Інженери повинні чітко розмежувати функціональні вимоги, такі як механізми реєстрації, публікації контенту чи обміну повідомленнями, та нефункціональні вимоги, до яких належать продуктивність, масштабованість і високий рівень інформаційної безпеки.

На етапі проєктування критично важливим є створення візуальних моделей, що дозволяють формалізувати складну логіку системи. Використання уніфікованої мови моделювання (UML) є стандартом індустрії. Структурні діаграми, зокрема діаграми класів, допомагають описати статичну архітектуру баз даних і зв'язки між профілями користувачів та мультимедійними об'єктами. Водночас поведінкові діаграми, такі як діаграми станів та послідовностей, дозволяють деталізувати динамічні процеси — від життєвого циклу сесії користувача до алгоритмів авторизації та оновлення персоналізованої стрічки новин.

Важливою складовою інженерного процесу є вибір архітектурного стилю. Для забезпечення безперервної роботи високонавантажених соціальних платформ розробники відмовляються від монолітних систем на користь мікросервісної архітектури. Цей підхід дозволяє розділити загальний функціонал на десятки незалежних модулів, таких як сервіс обробки зображень, система миттєвих повідомлень чи модуль рекомендацій. Такий поділ спрощує розгортання оновлень та дозволяє локалізувати технічні збої. Управління даними в таких архітектурах реалізується через комбіновані рішення: графові бази даних використовуються для швидкого пошуку соціальних зв'язків, а NoSQL-системи — для швидкого кешування та обробки великих обсягів неструктурованого медіаконтенту.

Оскільки вимоги до функціонала соціальних мереж постійно еволюціонують під впливом ринку, класична каскадна модель розробки (Waterfall) є неефективною. Тому життєвий цикл програмного забезпечення (SDLC) зазвичай базується на гнучких методологіях сімейства Agile, зокрема Scrum. Ітеративний підхід гарантує регулярний випуск нових версій продукту через практики безперервної інтеграції та розгортання (CI/CD). Це дозволяє командам швидко тестувати гіпотези, збирати зворотний зв'язок від аудиторії та оперативно виправляти вразливості, що є критично важливим для захисту конфіденційних даних.

Управління процесом створення соціальної мережі вимагає залучення великих міждисциплінарних команд, тому невіддільною частиною розробки є проєктний менеджмент. Використання таких інструментів, як діаграми Ганта, дозволяє візуалізувати часові межі роботи над кожним спринтом, розподіляти завдання між розробниками та контролювати використання ресурсів. Окрему увагу приділяють забезпеченню якості (QA): для соціальних мереж обов'язковим є автоматизоване навантажувальне та стрес-тестування, що імітує пікові сплески активності під час вірусного поширення контенту.

Отже, соціальні мережі є надскладними програмними комплексами, створення яких неможливе без застосування формальних методів програмної інженерії. Поєднання ґрунтовного збору вимог, грамотного архітектурного моделювання, ітеративної розробки та ефективного менеджменту команди є надійним шляхом для побудови масштабованих та безпечних комунікаційних платформ.

#### СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Sommerville I. Software Engineering. – 10th ed. – Pearson, 2015. – 816 p.
2. Мартин Р. Чиста архітектура. Мистецтво розробки програмного забезпечення. – Фабула, 2019. – 368 с.

**Коваль Вероніка Анатоліївна** – студентка групи 2ПІІ-24б, факультет інформаційних технологій та комп'ютерної інженерії, Вінницький національний технічний університет, м. Вінниця, e-mail: nika0678910860@gmail.com

**Коваленко Олена Олексіївна** – к.т.н., доцент кафедри програмного забезпечення, Вінницький національний технічний університет, м. Вінниця, e-mail: kovalenka88@gmail.com.

**Koval Veronika** – a student of 2PII-24b, Faculty of Information Technologies and Computer Engineering, Vinnytsia National Technical University, Vinnytsia, e-mail: nika0678910860@gmail.com

**Olena Kovalenko** – PhD, Associate Professor of the Software Engineering Department, Vinnytsia National Technical University, Vinnytsia, e-mail: kovalenka88@gmail.com.