

УДК 004.415

ДОСЛІДЖЕННЯ ПАТЕРНІВ ТА АРХІТЕКТУР В МОБІЛЬНІЙ РОЗРОБЦІ

Пашков Д. О.

e-mail: dmytro.pashkov@nure.ua

Науковий керівник - д.т.н, проф. Хаханова Г.В.

Харківський національний університет радіоелектроніки,

каф. АПОТ

м. Харків, Україна

This work is devoted to the exploration of design patterns and architectural approaches in modern mobile development. It highlights the importance of structured architectural thinking for building scalable, maintainable, and testable applications. The study reviews commonly used patterns in iOS development, such as delegation, observation, and singleton usage, as well as the application of SOLID principles in practical software design. Special attention is given to the evolution of architectural solutions over time. The work emphasizes that there is no universal or ideal architecture, as each approach is shaped by technological context and project requirements. This is illustrated through the transformation of the classical MVC concept into the so-called “Massive View Controller” phenomenon, which motivated the search for alternative patterns. The emergence of declarative frameworks has further influenced architectural trends, making MVVM a dominant approach in modern UI development, while also revealing its own limitations.

У сучасній мобільній розробці питання вибору архітектури та використання дизайн-патернів відіграє ключову роль у створенні якісного програмного забезпечення. Зі зростанням складності застосунків підвищуються вимоги до їх масштабованості, підтримуваності та зрозумілості коду. Саме тому дослідження підходів до організації програмної структури є актуальним завданням для розробників.

У межах роботи було розглянуто важливість застосування класичних патернів проектування, які широко використовуються в iOS-розробці, зокрема Delegate, Observer та Singleton. Їх використання дозволяє формалізувати взаємодію між компонентами системи, зменшити зв'язність модулів та підвищити повторне використання коду. Окрему увагу приділено принципам SOLID як фундаменту побудови гнучкої та розширюваної архітектури. Також було проаналізовано популярні архітектурні підходи, що застосовуються у різних технологічних контекстах. Класична архітектура MVC, попри свою концептуальну простоту та логічність у теорії, на практиці часто призводила до появи так званих Massive View Controller, що ускладнювало підтримку та розвиток проєктів. Це стало поштовхом до пошуку альтернативних підходів, таких як MVP, VIPER та інші. З появою декларативного фреймворку SwiftUI значного поширення

набула архітектура MVVM, яка краще відповідає новій парадигмі побудови інтерфейсів. Водночас навіть цей підхід не є позбавленим недоліків, зокрема ризику надмірної логіки у ViewModel або складності управління станом у великих проєктах. Таким чином, архітектурні рішення постійно еволюціонують разом із розвитком інструментів і підходів до розробки.

У результаті дослідження можна зробити висновок, що не існує універсальної або ідеальної архітектури для мобільних застосунків. Кожен підхід має свої переваги та обмеження і повинен обиратися з урахуванням специфіки проєкту, команди та технологічного середовища. Гнучкість мислення, розуміння фундаментальних принципів і здатність адаптуватися до змін є ключовими чинниками успішної роботи сучасного мобільного розробника.

Список використаних джерел:

1. Design Patterns: Elements of Reusable Object-Oriented Software / E. Gamma, R. Helm, R. Johnson, J. Vlissides. — Boston : Addison-Wesley, 1994. — 395 p.
2. Clean Architecture: A Craftsman's Guide to Software Structure and Design / R. C. Martin. — Boston : Prentice Hall, 2017. — 432 p.
3. Agile Software Development: Principles, Patterns, and Practices / R. C. Martin. — Upper Saddle River : Prentice Hall, 2003. — 529 p.
4. iOS App Architecture: Data Flow and Persistence / W. Boles. — Berkeley : Apress, 2020. — 350 p.