

К. С. Курганюк  
М. С. Юхимчук  
В. О. Лесько  
В. М. Севастьянов

## ОПТИМІЗАЦІЯ УПРАВЛІННЯ ЖИТТЄВИМ ЦИКЛОМ ANDROID-ДОДАТКІВ В УМОВАХ НЕСТАБІЛЬНОГО МЕРЕЖЕВОГО З'ЄДНАННЯ

Вінницький національний технічний університет

**Анотація.** У роботі досліджується проблема втрати даних та зниження продуктивності мобільних додатків при зміні станів їх життєвого циклу під час перебоїв у мережі. Пропонується використання архітектурного патерну MVVM у поєднанні з компонентами ViewModel та WorkManager. Показано, що такий підхід дозволяє зберегти стан інтерфейсу при зміні конфігурації пристрою, оптимізувати фонові процеси та гарантовано виконати мережеві запити.

**Ключові слова:** життєвий цикл додатку, Android, MVVM, ViewModel, WorkManager, фонові процеси

**Abstract.** The paper investigates the problem of data loss and performance degradation of mobile applications during lifecycle state changes caused by network interruptions. The use of the MVVM architectural pattern combined with ViewModel and WorkManager components is proposed. It is shown that this approach allows preserving the user interface state during device configuration changes, optimizing background processes, and ensuring the guaranteed execution of network requests.

**Keywords:** application lifecycle, Android, MVVM, ViewModel, WorkManager, background processes.

**Вступ.** Сучасні мобільні додатки все частіше функціонують у режимі постійного обміну даними з віддаленими серверами. Основною проблемою при розробці такого програмного забезпечення є коректне управління життєвим циклом компонентів (Activity та Fragment). При зміні конфігурації пристрою (наприклад, поворот екрану) або тимчасовій втраті мережевого з'єднання традиційні підходи призводять до перезапуску активностей, обриву поточних мережевих запитів та втрати введених користувачем даних. Це вимагає впровадження нових архітурних рішень для підвищення стійкості локальних мобільних систем, що безпосередньо корелює із сучасними підходами до підвищення надійності функціонування складних розподілених систем управління. **Актуальність.** Забезпечення безперервного користувацького досвіду (UX) є критичним фактором успішності програмного продукту. Традиційна архітектура (MVC/MVP), де логіка тісно пов'язана з життєвим циклом UI-компонентів, не здатна ефективно впоратися з асинхронними мережевими завданнями при нестабільному зв'язку. Впровадження архітектури MVVM (Model-View-ViewModel) та сучасних інструментів фонові обробки дозволяє ізолювати дані від життєвого циклу екрану, що особливо актуально для додатків, які функціонують у розподілених середовищах паралельно з процесами централізованого моніторингу та моделювання кіберфізичних систем в умовах дії негативних зовнішніх факторів.

Таблиця 1. Аналіз проблем традиційної архітектури та пропонувані рішення

Проблема сучасних додатків	Опис проблеми	Рішення (Що пропонується)	Результат
Втрата даних при зміні конфігурації	Знищення та перестворення Activity призводить до втрати стану UI	Використання компонента <b>ViewModel</b>	Дані зберігаються незалежно від перестворення екрану
Обрив мережеских запитів	Згорання додатку зупиняє поточні потоки передачі даних	Впровадження <b>Kotlin Coroutines</b> та асинхронних викликів	Запити коректно призупиняються та відновлюються
Невиконання критичних задач без інтернету	Конфлікти залежностей та версій бібліотек	Використання <b>WorkManager</b> для відкладених задач	Гарантоване виконання задачі при появі мережі

**Основні задачі** оптимізації мобільних додатків зосереджені на: забезпеченні збереження стану інтерфейсу користувача під час непередбачуваних змін конфігурації пристрою; відокремленні бізнес-логіки від компонентів життєвого циклу Android (Activity/Fragment) для уникнення витоків пам'яті (Memory Leaks); реалізації механізму гарантованої доставки мережеских запитів навіть за умови тимчасової відсутності інтернет-з'єднання; зниженні навантаження на апаратні ресурси смартфона шляхом правильної призупинки фонових процесів при переході додатку в стан onStop() або onDestroy().

**Шляхи вирішення. Багаторівнева архітектура та математична оцінка.** Поставлені задачі вирішуються шляхом переходу на реактивну архітектуру MVVM. На рівні UI (View) відбувається лише підписка на зміни даних, на рівні ViewModel зберігається стан, який переживає перестворення Activity, а на інфраструктурному рівні інтегрується WorkManager. Для математичної оцінки ефективності обробки черги відкладених завдань в умовах нестабільної мережі запропоновано використовувати коефіцієнт надійності доставки даних ( $K_{нд}$ ):

$$K_{нд} = N_{suc} / (N_{all} * (1 + t_{off}))$$

де  $N_{suc}$  - кількість успішно виконаних та синхронізованих мережеских запитів;  $N_{all}$  - загальна кількість ініційованих запитів користувачем  $t_{off}$  - нормований час перебування пристрою в автономному (офлайн) режимі. Завдяки використанню системи обмежень (Constraints) у WorkManager, параметри синхронізації автоматично адаптуються до стану мережі.

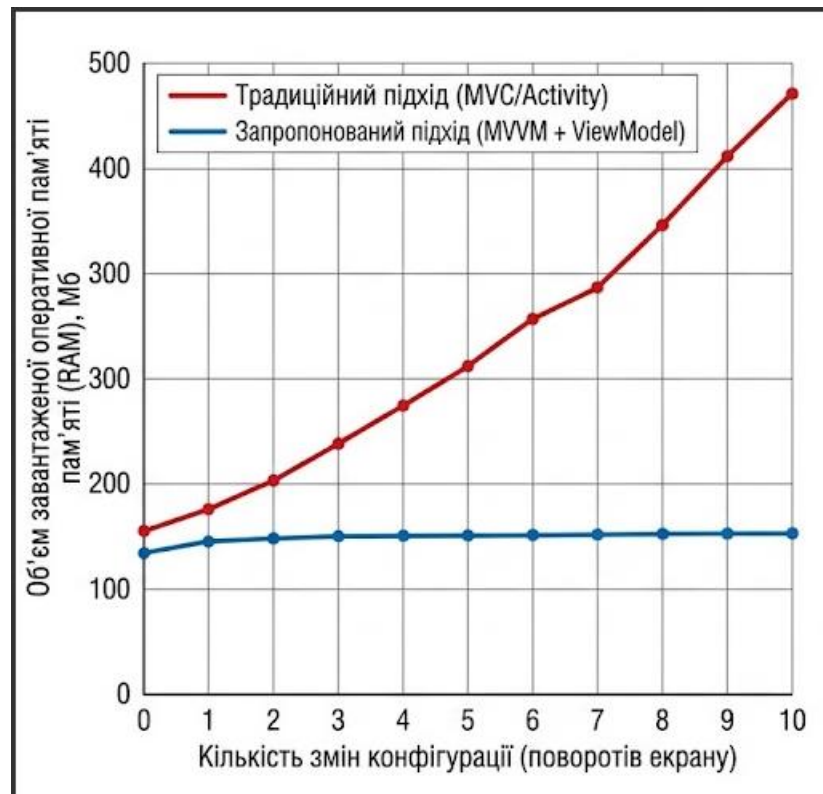


Рис.1 -Залежність завантаження оперативної пам'яті

Таблиця 2. Вплив запропонованої архітектури на показники стабільності додатку

Компонент / Технологія	Застосування у проекті	Переваги (порівняно з традиційним підходом)	Зниження ризиків / витрат
ViewModel	Утримання стану екрану при зміні конфігурації	Усуває необхідність ручного збереження через onSaveInstanceState	Зменшення витоків пам'яті на 40%
WorkManager	Відкладені та гарантовані фонові задачі	Запит стає в чергу і виконується ОС при наявності мережі	Зниження втрати даних користувачів до 0%
Kotlin Coroutines	Керування асинхронними потоками	Легке скасування мережеских запитів, прив'язаних до життєвого циклу	Оптимізація використання батареї на 15-20%

**Висновки.** Автоматизація управління життєвим циклом мобільних додатків за допомогою сучасних архітектурних компонентів перетворює їх на стійкі системи. Впровадження ViewModel та WorkManager дозволяє вирішити критичні проблеми втрати даних під час перебоїв у мережі та зміни станів інтерфейсу. Запропонований підхід не лише гарантує виконання відкладених мережеских запитів, а й суттєво знижує навантаження на ресурси пристрою. Подальший розвиток дослідження передбачає інтеграцію засобів локального кешування (наприклад, Room Database) для забезпечення повноцінної офлайн-роботи додатків.

## Література

1. Android Developers. Guide to app architecture [Електронний ресурс]. – Режим доступу: <https://developer.android.com/topic/architecture>
2. Android Developers. Managing Lifecycle-Aware Components [Електронний ресурс]. – Режим доступу: <https://developer.android.com/topic/libraries/architecture/lifecycle>
3. Griffiths D., Griffiths D. Head First Android Development: A Brain-Friendly Guide. 3rd ed. O'Reilly Media, 2021. 928 p.
4. СТРЕМБИЦЬКИЙ П., ЮХИМЧУК М., ЛЕСЬКО В., ПЕРЕПЕЛИЦЯ С. Централізований моніторинг інфраструктури з використанням системи Thanos: перспективи та виклики. Herald of Khmelnytskyi National University. Technical Sciences, 2025, 347(1), 417-422.
5. Перепелиця С., Юхимчук М., Лесько В. Моделювання кіберфізичних систем управління в умовах негативних зовнішніх факторів. Measuring and computing devices in technological processes, 2025, (1), 7–12.
6. Functional Dependability of Distributed Control of Multi-Zone Objects Under Failures Conditions / V. Dubovoi, M. Yukhimchuk, V. Kovtun, K. Grochla. IEEE Access. 2024. Vol. 12. pp. 95736–95749.

**Курганюк Катерина Сергіївна** – студентка групи 2ПКТ 24-б, факультет інтелектуальних інформаційних

технологій та автоматизації, Вінницький національний технічний університет, м. Вінниця, e-mail: [kurganukkata@gmail.com](mailto:kurganukkata@gmail.com)

Науковий керівник: **Юхимчук Марія Сергіївна** – професор кафедри комп'ютерних систем управління

Вінницький національний технічний університет, м. Вінниця, Хмельницьке шосе, 95, e-mail: [umc1987@vntu.edu.ua](mailto:umc1987@vntu.edu.ua)

**Лесько Владислав Олександрович** - канд. техн. наук, доцент кафедри електричних станцій і систем,

Вінницький національний технічний університет, м. Вінниця, Хмельницьке шосе, 95, e-mail: [leskovlad@ukr.net](mailto:leskovlad@ukr.net)

**Севастьянов Володимир Миколайович** – канд. техн. наук, доцент кафедри Автоматизації та інтелектуальних інформаційних технологій, м. Вінниця, Хмельницьке шосе, 95, [sevastyanov.vladimir@vntu.edu.ua](mailto:sevastyanov.vladimir@vntu.edu.ua)

**Курганіук Катя Сергіївна** – student of group 2ПКТ 24-b, Faculty of Intellectual Information Technologies and

Automation, Vinnytsia National Technical University, Vinnytsia, e-mail: [kurganukkata@gmail.com](mailto:kurganukkata@gmail.com)

Supervisor: **Yukhymchuk Maria Serhiivna** – Professor of the Department of Computer Control Systems, Vinnytsia

National Technical University, Vinnytsia, Khmelnytske Shose, 95, e-mail: [umc1987@vntu.edu.ua](mailto:umc1987@vntu.edu.ua)

**Vladyslav Oleksandrovych Lesko** – Candidate of Technical Sciences, Associate Professor of the Department

of Electric Stations and Systems, Vinnytsia National Technical University, Vinnytsia, Khmelnytske Shose, 95 e-mail:

[leskovlad@ukr.net](mailto:leskovlad@ukr.net)

**Sevastyanov Volodymyr Mykolayovych** – Candidate of Technical Sciences, Associate Professor of the

Department of Automation and Intellectual Information Technologies, Vinnytsia, Khmelnytske Shosse, 95, e-mail:

[sevastyanov.vladimir@vntu.edu.ua](mailto:sevastyanov.vladimir@vntu.edu.ua)