

DOI 10.36074/logos-15.11.2024.033

ЗАСОБИ ГЕНЕРАЦІЇ ПРОГРАМНОГО КОДУ ДЛЯ РОБОТИ З БАЗАМИ ДАНИХ У МОБІЛЬНИХ АНДРОЇД-ЗАСТОСУНКАХ

Бармутов Роман Максимович¹

Науковий керівник: Онищенко Борис Олегович²

1. здобувач вищої освіти факультету обчислювальної техніки,
інтелектуальних та управляючих систем

Черкаський національний університет імені Богдана Хмельницького, УКРАЇНА

2. канд.фіз.-мат.наук, доцент,

доцент кафедри програмного забезпечення автоматизованих систем

Черкаський національний університет імені Богдана Хмельницького, УКРАЇНА

ORCID ID: 0000-0001-5685-7597

З огляду на швидкий розвиток мобільних технологій, питання ефективної роботи з базами даних у мобільних застосунках стає дедалі актуальнішим. Інструменти генерації програмного коду для взаємодії з базами даних відіграють важливу роль в інтеграції, спрощенні та оптимізації програмного коду.

У даній роботі розглядатиметься дослідження та розробка інструменту, який спростить взаємодію з базами даних та звільнить розробників від рутинної праці з написання програмного коду для вбудованої системи управління базами даних у Андроїд-застосунках [1].

Для ефективної розробки Андроїд-застосунків необхідно враховувати особливості генерації програмного коду для роботи з базами даних. Такий код може автоматизувати рутинні завдання, зокрема: створення бази даних, створення таблиць та полів, встановлення зв'язків між таблицями, обробку запитів, створення складних ключів, індексацію полів.

Розглянемо та порівняємо декілька вже існуючих інструментів генерації програмного коду для баз даних, таких як Hibernate, Doctrine та Room, у таблиці 1.

Таблиця 1

Порівняння інструментів генерації програмного коду для баз даних

Критерій	Hibernate	Doctrine	Room
Реверс-інжиніринг (з таблиці в клас)	Так	Так	Ні
Генерація таблиць з класів	Так	Так	Так
Автоматична генерація SQL-коду	Так	Так	Так (через анотації)
Генерація складних SQL-запитів	Так	Так	Ні (вручну через @Query)
Оптимізація для мобільних застосунків	Ні (може мати великий оверхед)	Ні	Так (легкий і оптимізований для SQLite)
Гнучкість у налаштуванні мапінгу	Так	Так	Обмежена (в основному через анотації)
Кешування результатів	Так (вбудована підтримка кешування)	Так (через налаштування)	Так
Підтримка складних зв'язків між таблицями	Так	Так	Так (але менш гнучка)
Можливість кастомізації генерації коду	Так	Так	Обмежена
Вбудована підтримка транзакцій	Так	Так	Так
Гнучкість в написанні власних SQL-запитів	Так	Так	Ні (тільки через @Query)
Відкладене завантаження (Lazy Loading)	Так	Так	Ні

Таблиця надає загальне уявлення про можливості Hibernate, Doctrine та Room як інструментів для генерації програмного коду для баз даних. Room є оптимальним вибором для мобільних застосунків, проте в окремих аспектах поступається за гнучкістю Hibernate і Doctrine. Ця порівняльна характеристика в таблиці 1 допоможе визначити, які переваги Hibernate і Doctrine варто врахувати при створенні інструменту для генерації програмного коду баз даних, спеціально орієнтованого на Андроїд-застосунки.

Аналізуючи процес розробки Андроїд-застосунків, було визначено потребу в інструментах, які б автоматизували створення коду для роботи з базами даних. Це передбачає спрощення написання SQL-запитів, автоматичне формування структури таблиць та їхніх зв'язків, а також оптимізацію доступу до даних шляхом індексації.



SEZIONE 16.

INFORMATICA E INGEGNERIA DEL SOFTWARE

Розробка інструменту базуватиметься на Room Persistence Library [2] – офіційній бібліотеці від Google, яка спрощує роботу з базою даних SQLite. Завдяки використанню анотацій, Room дозволяє описати структуру бази даних прямо в коді, автоматично генеруючи необхідні SQL-запити. Бібліотека також надає зручні компоненти для виконання запитів і підтримує LiveData [3], що забезпечує реактивне оновлення інтерфейсу користувача при зміні даних у базі.

Висновки. Завдяки розробленому плагіну для Android Studio, розробники зможуть значно прискорити процес створення Андрюїд-застосунків, оскільки всі рутинні операції з базами даних будуть виконуватися автоматично.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ:

- [1] Учасники проєктів Вікімедіа. Генерація коду – Вікіпедія. *Вікіпедія*. URL: https://uk.wikipedia.org/wiki/Генерація_коду (дата звернення: 29.10.2024).
- [2] Foong N. W. Save data in your local database using the room persistence library. *Medium*. URL: <https://betterprogramming.pub/save-data-in-your-local-database-using-the-room-persistence-library-a9630c977234> (дата звернення: 01.11.2024).
- [3] LiveData overview | android developers. *Android Developers*. URL: <https://developer.android.com/topic/libraries/architecture/livedata> (дата звернення: 03.11.2024).