

УДК 004.415.53

А. Мельник; Л. Дмитроца, к. т. н., доц.

(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

АЛГОРИТМІЧНА ОПТИМІЗАЦІЯ ПОКРИТТЯ СТАНІВ МОБІЛЬНИХ ДОДАТКІВ НА ОСНОВІ ГРАФО-ОРІЄНТОВАНОГО ПІДХОДУ

UDC 004.415.53

A. Melnyk; L. Dmytrotsa, PhD., Assoc. Prof.

ALGORITHMIC OPTIMIZATION OF STATE COVERAGE IN MOBILE APPLICATIONS BASED ON GRAPH-ORIENTED APPROACH

Сучасна парадигма автоматизації тестування переважно спирається на лінійне написання скриптів, де кожен тест-кейс є ізольованою послідовністю імперативних команд. Такий підхід має критичний недолік, відомий як «парадокс пестициду» [1], та характеризується значною надлишковістю коду. При масштабуванні мобільних екосистем лінійні скрипти не здатні гарантувати повне покриття всіх можливих шляхів користувача через комбінаторний вибух станів інтерфейсу.

Метод Model-Based Testing (MBT) розглядає тестований додаток як скінченний автомат. Використання підходу MBT дозволяє застосувати класичні алгоритми обходу графів, такі як пошук у глибину (DFS), пошук у ширину (BFS) або алгоритми пошуку найкоротшого шляху (наприклад, Дейкстри), для генерації тестових наборів. Це дає змогу не просто створити набір сценаріїв, а сформувати математично обґрунтовану послідовність дій, яка гарантує стовідсоткове покриття всіх можливих переходів між станами системи [2]. Такий детермінований підхід виключає людський фактор, притаманний ручному проектуванню тестів, і забезпечує перевірку навіть тих граничних випадків, які часто ігноруються при традиційному підході.

Ключовою архітектурною перевагою графо-орієнтованого підходу є можливість миттєвого динамічного перерахунку тестових шляхів при будь-якій зміні бізнес-логіки або інтерфейсу користувача. У той час як традиційні скрипти потребують ручного рефакторингу кожного окремого файлу, у MBT зміни вносяться лише в один централізований артефакт — модель графа. Після оновлення вершин або ребер моделі актуальний набір тестів регенерується автоматично, що дозволяє підтримувати тестову документацію в актуальному стані без зайвих витрат часу.

Для практичної реалізації методу пропонується використання проміжних форматів опису моделі, яка транслюється в нативний код тестів через розроблений парсер. Це дозволяє досягти перевірки послідовностей довжиною N переходів (N-switch coverage) [3], що практично неможливо реалізувати вручну. Перехід до Model-Based Testing дозволяє скоротити обсяг кодової бази тестів на 30–50% при одночасному збільшенні покриття функціональних переходів, що є критичним для забезпечення надійності високонавантажених систем.

Література

1. Utting M., Legeard B. Practical Model-Based Testing: A Tools Approach. Morgan Kaufmann Publishers, 2022. 450 p.
2. Arcuri A. Automated Test Generation: A Review of Algorithms and Industrial Application. *IEEE Transactions on Software Engineering*. 2023. Vol. 48, no. 5. P. 1120–1135.
3. Binder R. Testing Object-Oriented Systems: Models, Patterns, and Tools. Addison-Wesley Professional, 2024. 1200 p.