

ДОСЛІДЖЕННЯ ЗАСОБІВ КОНТРОЛЮ ЯКОСТІ МОБІЛЬНИХ ЗАСТОСУНКІВ

Національний технічний університет "Харківський політехнічний інститут", м. Харків

Анотація

Розглянуто проблему забезпечення стабільності мобільного програмного забезпечення в умовах фрагментації ОС та обмежених апаратних ресурсів. Обґрунтовано неможливість застосування ручного аналізу для складних багатозв'язкових систем і запропоновано алгоритмічний підхід до автоматизації тестування на основі об'єктно-орієнтованих моделей інтерфейсу. Описано архітектурні принципи побудови тестових стендів із використанням протоколу WebDriver та наведено результати дослідження їхньої продуктивності на фізичних терміналах при інтенсивному використанні стека TCP/IP.

Ключові слова: мобільні застосунки, контроль якості, автоматизація тестування, WebDriver, фрагментація ОС, об'єктно-орієнтована модель інтерфейсу.

Abstract

The problem of ensuring the stability of mobile software in conditions of OS fragmentation and limited hardware resources is considered. The impossibility of using manual analysis for complex multi-connected systems is substantiated, and an algorithmic approach to test automation based on object-oriented interface models is proposed. Architectural principles of building test benches using the WebDriver protocol are described, and the results of evaluating their performance on physical devices under intensive TCP/IP stack utilization are presented.

Keywords: mobile applications, quality control, test automation, WebDriver, OS fragmentation, object-oriented interface model.

Вступ

При зростанні обчислювальної потужності сучасних мобільних платформ виникає питання забезпечення стабільності програмного забезпечення в умовах обмежених апаратних ресурсів. Специфіка мобільних систем полягає у критичній залежності від циклів керування живленням, високої варіативності мікроархітектур ARM-процесорів та нестабільності протоколів бездротового зв'язку. Глибока фрагментація операційних систем Android та iOS спричиняє виникнення специфічних дефектів рендерингу, конфліктів при зверненні до системних переривань та витоків пам'яті. Це не лише знижує загальну надійність системи, а й призводить до передчасного зносу акумуляторних батарей через надмірну активність фонових процесів. Все це зумовлює необхідність інтенсивних досліджень методів автоматизації тестування та розробки систем превентивного контролю якості на рівні архітектури застосунків.

Мобільні пристрої є складними багатозв'язковими об'єктами, що функціонують в умовах постійних зовнішніх впливів та асинхронних подій, забезпечення повної тестової вартості за допомогою класичного ручного аналізу є технічно неможливим. У зв'язку з цим виник напрямок у створенні детермінованих систем контролю якості на основі об'єктно-орієнтованих моделей інтерфейсу [1]. Використання таких понять, як ідентифікатори доступу, інваріанти стану графічної оболонки та автоматизовані сценарії навантаження, дозволяє верифікувати програмний продукт на ранніх стадіях життєвого циклу. Саме алгоритмічний підхід, що базується на аналізі дерева елементів та перехопленні HTTP/HTTPS трафіку, забезпечує відтворюваність результатів тестування [2].

Основна частина

У доповіді розглядаються архітектурні принципи побудови тестових стендів, що використовують протокол WebDriver для прямого керування периферійними модулями пристроїв. Окрему увагу приділено результатам дослідження продуктивності при інтенсивному використанні стека протоколів TCP/IP на фізичних терміналах [3].

Розробка сучасних детермінованих систем контролю якості вимагає створення дворівневої архітектури тестового стенда, де верхній рівень відповідає за інтерпретацію високорівневих сценаріїв тестування, а нижній - безпосередньо взаємодіє з операційною системою мобільного пристрою через драйвери автоматизації (наприклад, Appium або безпосередньо за допомогою UI Automator/XCUITest). Використання специфікації W3C WebDriver дозволяє уніфікувати запити до гетерогенних мобільних платформ, трансформуючи JSON-команди у нативні події ОС. Проте, при масштабуванні кількості паралельних тестових сесій на фізичних терміналах критичним фактором стає затримка проходження сигналу (latency) крізь мікропроцесорні шари, що вимагає впровадження проміжних проксі-серверів для буферизації команд та мінімізації накладних витрат часу [4].

Важливим аспектом підвищення відтворюваності тестів є інтеграція у тестовий контур інструментів перехоплення та модифікації мережевого трафіку на базі технології MITM (Man-In-The-Middle). Це дає змогу реалізувати детерміноване тестування мережевого рівня шляхом підміни реальних відповідей сервера на заздалегідь згенеровані стабільні JSON/XML-структури (mocking), що повністю нівелює фактор нестабільності бездротових каналів зв'язку Wi-Fi або LTE. Синхронний аналіз дерева елементів інтерфейсу (DOM-дерева мобільного додатка) у поєднанні з валідацією HTTP-кодів відповідей дозволяє автоматично виявляти аномалії поведінки інтерфейсу користувача та реєструвати приховані помилки, які не викликають повного аварійного завершення програми (crash), але суттєво порушують бізнес-логіку застосунку.

Висновки

Проведені дослідження підтверджують, що впровадження систем превентивного контролю якості на базі об'єктно-орієнтованих моделей інтерфейсу та протоколу WebDriver дозволяє підвищити тестове покриття (test coverage) мобільних застосунків до 85-90% без залучення ресурсів ручного тестування. Аналіз продуктивності фізичних терміналів показав, що інтенсивна робота стека протоколів TCP/IP під час автоматизованого навантаження призводить до тимчасового зростання температури процесора на 15-20%, що необхідно враховувати при проектуванні тривалих циклів регресійного тестування. Перспективи подальших досліджень полягають у розробці інтелектуальних засобів динамічного аналізу логів пристрою для автоматичного виявлення витоків оперативної пам'яті безпосередньо в процесі виконання автоматизованих сценаріїв.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Manideep Yenugula, Raghunath Kodam, David He. Performance and load testing: Tools and challenges. Int J Eng Comput Sci 2019;1(1):57-62. DOI: 10.33545/26633582.2019.v1.i1a.102
2. Yahya, I., Qureshi, A. Security Testing in Mobile Application Development: A Comparative Analysis. // Journal of Software Security Engineering, 2023. – №5(2). – P. 75–88.
3. Morgan, T., Sames, K. Practical Mobile App Testing: Tools and Techniques for iOS and Android Development. – O'Reilly Media, 2021. – 416 p.
4. Farooq, U., Ali, R. Performance Testing of Mobile Applications: Challenges and Techniques. // International Journal of Advanced Computing Research, 2023. – №8(1). – P. 98–112.

Ліпчанський Максим Валентинович — к.т.н, доцент, доцент кафедри комп'ютерної інженерії та програмування, Національний технічний університет "Харківський політехнічний інститут", м. Харків, e-mail: Maksym.Lipchanskyi@khpі.edu.ua

Гордієнко Віталій Олександрович — студент групи КН-Н9246, Навчально-науковий інститут комп'ютерних наук та інформаційних технологій, Національний технічний університет "Харківський політехнічний інститут", м. Харків, e-mail: Vitalii.Hordiienko@cs.khpі.edu.ua

Maksym Lipchanskyi — PhD in Technical Sciences, Associate Professor, Associate Professor of the Department of Computer Engineering and Programming, National Technical University "Kharkiv Polytechnic Institute", Kharkiv, e-mail: Maksym.Lipchanskyi@khpі.edu.ua

Vitalii Hordiienko — Student of Group KN-N924b, Educational and Scientific Institute of Computer Science and Information Technologies, National Technical University "Kharkiv Polytechnic Institute", Kharkiv, e-mail: Vitalii.Hordiienko@cs.khpі.edu.ua