

застосунку, які не охоплюються тестами, завдяки чому з'являється можливість попередити можливі недоліки під час тестування.

2. Виявлення найбільш проблемних або непостійних тестів. Необхідно постійно ідентифікувати тести, які часто не проходять або вимагають значних зусиль для підтримки чи подальшого оновлення.

3. Аналіз трендів тестування з часом. Важливим аспектом є вивчення того, як змінюється ефективність тестування та якість програмного забезпечення з плином часу.

4. Порівняння результатів між різними ітераціями тестування. Завдяки постійному порівнянню результатів тестування різних версій застосунку реалізується потреба у своєчасному покращенні процесу тестування та ідентифікації поточних проблем.

Таким чином, автоматизований збір та аналіз метрик тестування дозволяє підвищити ефективність, поліпшити якість програмного забезпечення та сприяє швидкому виявленню проблем. Для успішної реалізації автоматизованих методів збору та аналізу метрик необхідна інтеграція інструментів та систем, стандартизація процесів і управління отриманими даними.

Література:

1. Hannan, Abdul & Raja, Hashir & Aamir, Daniyal & Ashraf, Saad. (2022). Metrics for Measuring the Effectiveness of Automated Software Testing Tools. <https://doi.org/10.13140/RG.2.2.22745.60000>.
2. Bures, Miroslav. (2015). Metrics for automated testability of web applications. 83-89. <https://doi.org/10.1145/2812428.2812458>.

УДК 004.05

Іващенко Є. – ст. гр. СНм-61

Тернопільський національний технічний університет імені Івана Пулюя

ОПТИМАЛЬНИЙ РОЗПОДІЛ РУЧНИХ ТА АВТОМАТИЗОВАНИХ ТЕСТІВ У ТЕСТОВОМУ ПОКРИТТІ

Науковий керівник: к.ф.-м.н., доцент Гром'як Р.

Ivashchenko Y.

Ternopil Ivan Puluj National Technical University

OPTIMAL DISTRIBUTION OF MANUAL AND AUTOMATED TESTS IN THE TEST COVERAGE

Supervisor: Hromiak R.

Ключові слова: тестові сценарії, скрипти, ефективність розподілу.

Keywords: test scenarios, scripts, distribution efficiency.

Оптимальний розподіл ручних та автоматизованих тестів у тестовому покритті є важливою стратегією для забезпечення якості програмного забезпечення. Цей підхід враховує ефективність і результативність обох видів тестування залежно від характеристик проекту, його обсягу, інтеграцій, часу та доступних ресурсів.

Ручне тестування вимагає присутності людського фактору і включає в себе виконання тестових сценаріїв вручну. Це дозволяє тестувальникам застосовувати свій

досвід, креативність та інтуїцію для виявлення помилок, які можуть залишитися непоміченими в автоматизованих процесах. Воно дозволяє здійснювати тестування, яке вимагає людського сприйняття, таке як валідація інтерфейсу користувача або аналіз взаємодії з системою.

Автоматизоване тестування, з іншого боку, передбачає використання спеціальних інструментів і скриптів для виконання тестових сценаріїв. Це дозволяє автоматизувати повторювані тести, регресійні тести, а також валідацію широкого спектру функціональностей. Автоматизовані тести ефективні у відлагодженні, виконанні великих обсягів тестування та забезпеченні швидкості виявлення помилок. Вони дозволяють виконувати однотипні операції безперервно і забезпечують стійкість до людського фактору.

Оптимальний розподіл ручних та автоматизованих тестів у тестовому покритті полягає в забезпеченні максимальної ефективності обох підходів. Для досягнення цього можна використовувати наступний підхід:

1. Вибір критичних областей для автоматизації: необхідно розпочати з ідентифікації тестів, які мають високий рівень повторюваності або є критичними для функціональності продукту. Ці тести найбільш підходять для автоматизації, оскільки вони вимагають багаторазового виконання і не залежать від контексту користувача.

2. Визначення тестів, що потребують людського сприйняття: наступним чином необхідно ідентифікувати тестові сценарії, де важлива роль відводиться людському фактору, тобто перевірка інтерфейсу користувача, взаємодії з системою або емуляція реального користувача.

3. Аналіз ризиків і пріоритетів: необхідно враховувати рівень ризику для різних функціональних областей та пріоритети для розподілу зусиль між ручним та автоматизованим тестуванням. Розуміння цих факторів дозволить ефективно розподілити ресурси.

4. Створення і підтримка автоматизованих скриптів: кроком є забезпечення належної уваги створенню і підтримці автоматизованих тестів. Це включає створення стабільних і надійних скриптів, а також їх регулярне оновлення у відповідь на зміни у програмному забезпеченні.

5. Постійна оцінка і вдосконалення стратегії: важливо здійснювати постійний аналіз ефективності розподілу тестових зусиль та вносити відповідні коригування. Стратегія тестування повинна бути гнучкою і адаптивною до змін у вимогах і умовах проекту.

У підсумку, оптимальний розподіл ручних та автоматизованих тестів у тестовому покритті забезпечує ефективне використання доступних ресурсів і забезпечує високу якість тестування програмного забезпечення. Використання цієї стратегії дозволяє збільшити швидкість виявлення помилок, покращити надійність продукту і зменшити загальні витрати на тестування.

Література:

1. Taipale, O., Kasurinen, J., Karhu, K. Trade-off between automated and manual software testing. *Int J Syst Assur Eng Manag* 2, 114–125 (2011). <https://doi.org/10.1007/s13198-011-0065-6>.
2. Päivi Raulamo-Jurvanen. Decision Support for Selecting Tools for Software Test Automation. *SIGSOFT Softw. Eng. Notes* 41, 6 (2016), 1–5. <https://doi.org/10.1145/3011286.3011304>.