

Особливості автоматизованого тестування веб-застосунків

Вінницький національний технічний університет

Анотація

В роботі досліджуються особливості автоматизованого тестування функціоналу веб-застосунків, обґрунтовується актуальність створення інформаційної технології тестування веб-застосунків, показані переваги автоматизованого тестування над тестуванням вручну.

Ключові слова: автоматизоване тестування, веб-застосунки, тестування функціоналу

Abstract

The problem of automated testing of web applications functionality are investigated in the work, the urgency of creating information technology for testing web applications is substantiated, the advantages of automated testing over manual testing are shown.

Keywords: automated testing, web applications, functional testing

Вступ

Сучасне програмне забезпечення є складним багатофункціональним об'єктом. Його ручна перевірка вимагає значних трудових і тимчасових витрат. На допомогу приходять засоби автоматизації тестування, які підвищують якість, забезпечують повторне використання тестів при коригуванні ПЗ. Основна задача автоматизації тестування мобільних додатків - скорочення витрат на випробування програми після її модернізації та позбавлення при людського фактору перевірок[1].

Метою роботи покращення характеристик процесу тестування функціоналу веб-застосунків.

Об'єктом дослідження є процес тестування функціоналу веб-застосунків.

Предметом дослідження інформаційна технологія тестування веб-застосунків.

Результати досліджень

Однією з головних проблем автоматизованого тестування є його трудомісткість: попри те, що воно дозволяє усунути частину рутинних операцій і прискорити виконання тестів, великі ресурси можуть витрачатися на оновлення самих тестів. Це відноситься до обох видів автоматизації. При рефакторингу часто буває необхідно оновити і модульні тести, і зміна коду тестів може зайняти стільки ж часу, скільки і зміна основного коду. З іншого боку, при зміні інтерфейсу програми необхідно заново переписати всі тести, які пов'язані з оновленими вікнами, що при великій кількості тестів може відняти значні ресурси[2].

Існує два основних підходи до автоматизації тестування: тестування на рівні коду і GUI-тестування. До першого типу належить, зокрема, модульне тестування. До другого — імітація дій користувача за допомогою спеціальних тестових фреймворків.

Найпоширенішою формою автоматизації є тестування додатків через графічний інтерфейс користувача. Популярність такого виду тестування пояснюється двома факторами: по-перше, додаток тестується тим же способом, яким його буде використовувати людина, по-друге, можна тестувати додаток, не маючи при цьому доступу до вихідного коду[3].

Зараз на ринку існує величезна різноманітність різних продуктів для автоматизованого тестування, деякі компанії-розробники програмного забезпечення створюють власні інструменти, пристосовані для тестування розроблених ними додатків. Причинами цього є висока вартість засобів автоматизованого тестування та унікальність програмного забезпечення, що тестується, яка

не дозволяє використовувати стандартні засоби автоматизації тестування. Крім засобів тестування існують так звані засоби підтримки процесу тестування, що дозволяють вести облік вимоги і тест-кейси, проводити аналіз покриття вимоги тестами, керувати ходом виконання тестування, вести облік виявлених дефектів і тому подібне[4].

Оскільки мова йде про тестування саме функціоналу веб-застосунків, то варто зазначити, що потрібно брати до уваги саме ті існуючі інформаційні технології, які призначені для вирішення цієї проблеми. Тобто йдеться про тестування саме додатків, що написані на HTML, і відповідно для них не підійдуть інструменти, фреймворки та бібліотеки, які призначені для тестування додатків, написаних наприклад на Java чи .NET, а оскільки в роботі розглядається тестування саме веб-застосунків, то важливо враховувати кроссбраузерність обраної технології, для її використання.

До таких інструментів, що відповідають даним характеристикам належить Selenium. Цей фреймворк дозволяє отримати найбільш повноцінну імітацію дій користувача, в тому числі наприклад тестування в певній версії браузера або використання флеш-плагіна. Selenium використовується для відправки HTTP запитів драйверу за допомогою протоколу JsonWireProtocol, в яких зазначено дію, яку повинен зробити браузер в рамках поточної сесії. Прикладами таких команд можуть бути команди знаходження елементів по локатору, перехід по посиланнях, натискання кнопок або перехід по посиланнях на сторінці веб-сайту. Тому для створення інформаційної технології тестування веб-застосунків можна використати можливості цього фреймворка[5].

Висновки

Хоч автоматичні тести не можуть повністю замінити ручне тестування, оскільки автоматизація всіх випробувань — дуже дорогий процес, і тому автоматичне тестування є скоріше лише доповненням ручного тестування, варто зазначити, що створення інформаційної технології тестування веб-застосунків, яка полягатиме у створенні програмного продукту, який спрощуватиме цей процес, за рахунок роботи універсальних шаблонів, які можна буде застосувати до більшості стандартних веб-застосунків і таким чином зменшити часові витрати на написання нових тестових сценаріїв під кожен окремий об'єкт тестування є актуальним.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Особливості тестування функціоналу веб-застосунків [Електронний ресурс] – Режим доступу: <https://conferences.vntu.edu.ua/index.php/all-fitki/all-fitki-2021/paper/view/11656/9800>
2. Автоматизоване тестування [Електронний ресурс] – Режим доступу: <en/topic/urbandevlopment/brief/solid-waste-management>
3. THE PROBLEM OF SOFTWARE TESTING AUTOMATYION [Електронний ресурс] – Режим доступу:http://elartu.tntu.edu.ua/bitstream/lib/27112/2/VII_MNTK_2018v2_Kravchuk_S_O-The_problem_of_software_testing_95.pdf
4. When Should a Test Be Automated? [Електронний ресурс] – Режим доступу: <https://www.stickyminds.com/article/when-should-test-be-automated>
5. Selenium [Електронний ресурс] – Режим доступу: <https://selenium-python.readthedocs.io/>

Вовчарук Павло Юрійович — студент групи 2КН-21м, факультет інтелектуальних інформаційних технологій та автоматизації, Вінницький національний технічний університет, м. Вінниця, email : magister_vp@vntu.edu.ua

Озеранський Володимир Сергійович — кандидат технічних наук, старший викладач кафедри комп'ютерних наук, Вінницький національний технічний університет, м. Вінниця, email: ozersansky@ukr.net

Vovcharuk Pavlo – student of Intelligent Information Technologies and Automation Department, Vinnytsia National Technical University, Vinnytsia, e-mail: : magister_vp@vntu.edu.ua

Ozeransky Volodumur - Ph.D., senior lecturer, Vinnytsia National Technical University, Vinnytsia, email: ozeransky@ukr.net