

АВТОМАТИЗАЦИЯ ПРОЦЕССА ТЕСТИРОВАНИЯ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ: ТЕОРИЯ ПОВЫШЕНИЯ НАДЕЖНОСТИ И ПРАКТИЧЕСКАЯ РЕАЛИЗАЦИЯ

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Поболь П. В.

Меженная М. М. – канд. техн. наук,
доцент каф ИПиЭ

Цель разработки – создание и апробация SMART фреймворка для автоматизации процесса тестирования веб-приложений. В настоящее время веб-приложения являются наиболее распространенным видом программных продуктов (далее следуют мобильные нативные и десктоп приложения). При этом к качеству программного обеспечения предъявляются высокие требования. Кроме того, в соответствии с современной стратегией быстрого вывода программных продуктов на рынок, процесс тестирования, как правило, проводится в сжатые сроки. В связи с этим актуальной задачей является снижение денежных и временных затрат на создание автоматических тестов с помощью стандартных средств разработки путем разработки новых фреймворков или адаптации существующих под конкретные задачи проектов.

Фреймворк (англ. *framework* — каркас, структура) — структура программной системы; программное обеспечение, облегчающее разработку и объединение разных компонентов большого программного проекта. В работе SMART фреймворк – инструмент, используемый для автоматизации веб-приложений. В качестве базы разрабатываемого фреймворка используется программный инструмент Selenium Webdriver. Благодаря интеграции с различными системами (CI, фреймворки по сборке проектов, тестовые фреймворки) и объединению простых однострочных команд в более комплексные команды, SMART фреймворк способен сэкономить большое количество временных и денежных ресурсов. Стек технологий, используемых для разработки фреймворка: Java – объектно-ориентированный язык программирования [1]; Maven – фреймворк для автоматизации сборки проектов, специфицированных на XML-языке POM; TestNG – тестовый фреймворк; ReportNG – плагин для отчетности в рамках модульного тестирования TestNG.

На рис.1,а отображен код теста, написанного с помощью Selenium Webdriver – наиболее распространенного инструмента для тестирования веб-приложений. На рис.1,б отображен код теста, написанного на базе разработанного фреймворка: сложная логика вынесена на уровень фреймворка и код теста становится гораздо проще и изящнее.

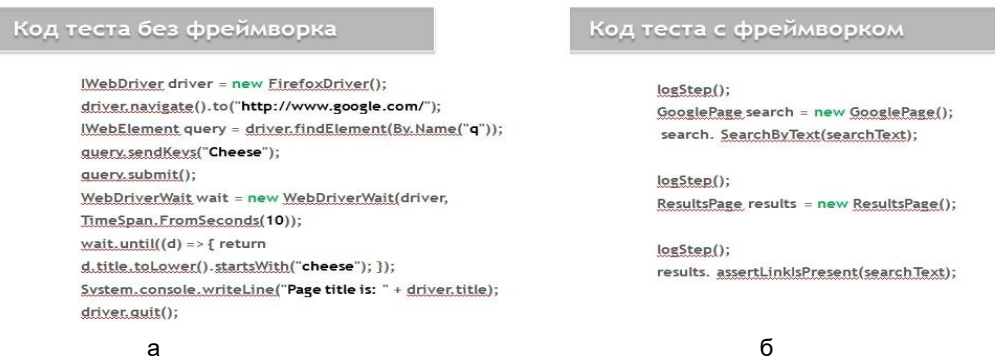


Рисунок 1 – Фрагменты автотеста, создано на базе традиционно используемого инструмента Selenium Webdriver (а) и разработанного автором фреймворка (б) (сценарии тестов идентичны)

Код теста без фреймворка выполняет (рис.1,а): переход на страницу Google.com; ввод в поле запроса слова “Cheese”; нажатие на кнопку “submit”; ожидание в течение 10 секунд; поиск на странице слова “cheese”. Код теста с фреймворком выполняет (рис.1,б): создание объекта `search` класса `GooglePage`, в котором отражена вся логика работы с этой страницей; применение метода `SearchByText(searchText)` (описан в классе `GooglePage`, но реализуется непосредственно в классе теста); аналогично – `results`, объект класса `ResultsPage`, в тесте реализуем описанный в этом классе метод `assertLinksPresent(searchText)`.

Преимущества разработанного SMART фреймворка: гибкая конфигурация параметров фреймворка; поддержка популярных браузеров (IE, FF, Chrome, Opera, Safari); простая адаптация для большинства веб-приложений; поддержка взаимодействия с базой данных и почтовыми серверами; интегрируемость с Continuous integration tools – системы непрерывной интеграции (Jenkins, Bamboo); удобный и простой формат отчетности.

Список использованных источников:

1. Философия Java / Б. Эккель : Питер, 2016. – 1168 с.