

ОСОБЕННОСТИ ФРЕЙМВОРКА PROTRACTOR В ТЕСТИРОВАНИИ ВЕБ-ПРИЛОЖЕНИЙ

Фрузоров М.О., Попова Ю.Б.

БНТУ, г. Минск, Беларусь, znakki@gmail.com

БНТУ, г. Минск, Беларусь, hawkrai@yandex.ru

Введение

В настоящее время активно развиваются сетевые информационные технологии. Ещё несколько лет назад пользователи популярных офисных и других программ имели необходимость устанавливать их на компьютер или рабочую станцию. В последние годы практически все компании на рынке IT разрабатывают свои приложения, ориентируясь, в первую очередь на web, либо переносят туда свои уже готовые продукты (например, skype, microsoft office, evernote, slack и другие). Это все делается для того, чтобы пользователь мог использовать любимое приложение без предварительной установки его на все рабочие машины. Таким образом, пользователю необходимо всего лишь открыть браузер, который установлен абсолютно на каждом компьютере, и ввести адрес нужного приложения.

Основными плюсами использования веб-приложений являются простота, удобство, доступность и скорость. Для реализации таких приложений существует множество различных фреймворков разработки, среди которых одним из наиболее популярных в последние три года является AngularJS. Этот JavaScript фреймворк предназначен для создания одностраничных веб-приложений, использующих единственный HTML-документ как оболочку для всех веб-страниц. Основным отличием одностраничных приложений от прочих является тот факт, что браузер никогда не знает, догрузилась страница приложения или нет. Каждый раз JavaScript на стороне клиента создает веб-интерфейс, а значит, между загрузкой веб-страницы и моментом, когда она становится доступной, проходит некоторое время. Дальнейший переход по ссылкам меняет только структуру DOM, но не перезагружает страницу.

1. Автоматизация тестирования – сравнение технологий Java и JavaScript

До недавнего времени основным языком разработки в автоматизированном тестировании считалась Java, с использованием которой выполнение тестового сценария происходит последовательно и синхронно, т.е. необходимо выполнить каждый тестовый шаг и обязательно дождаться его результата. При тестировании web-приложений, реализованных на JavaScript (JS), необходимо помнить, что это асинхронный язык программирования, и тестируемое приложение выполняет по несколько тестируемых шагов одновременно. И, если для такого приложения попытаться написать автоматизированные тесты на Java, то возникает проблема. На рисунке 1 показаны принципы автоматизации на Java и на Javascript.

Как видно из рисунка, асинхронность JS в автоматизации тестирования позволяет выполнять тестовые шаги одновременно и использовать их результаты в любое время и на любом шаге тестового сценария. Также существует фреймворк Protractor, основная идея которого – обеспечить синхронизацию между тестовым кодом (тестовыми скриптами) и приложением, построенным на популярном фреймворке AngularJS [2]. Рассмотрим автоматизацию с помощью Protractor подробнее.

2. Особенности использования Protractor-фреймворка

Protractor – это фреймворк, который базируется на WebDriverJS, разрабатывается и поддерживается командой Angular (Google) и включает в себя:

- фреймворк для запуска из интерфейса командой строки;
- обеспечение интеграции с популярными тестовыми фреймворками, такими как Jasmine, Mocha, Cucumber;
- обеспечение обработки синхронизации;
- обеспечение дополнительного API для работы с DOM-элементами;

- обеспечение синхронизации с Angular-приложением.

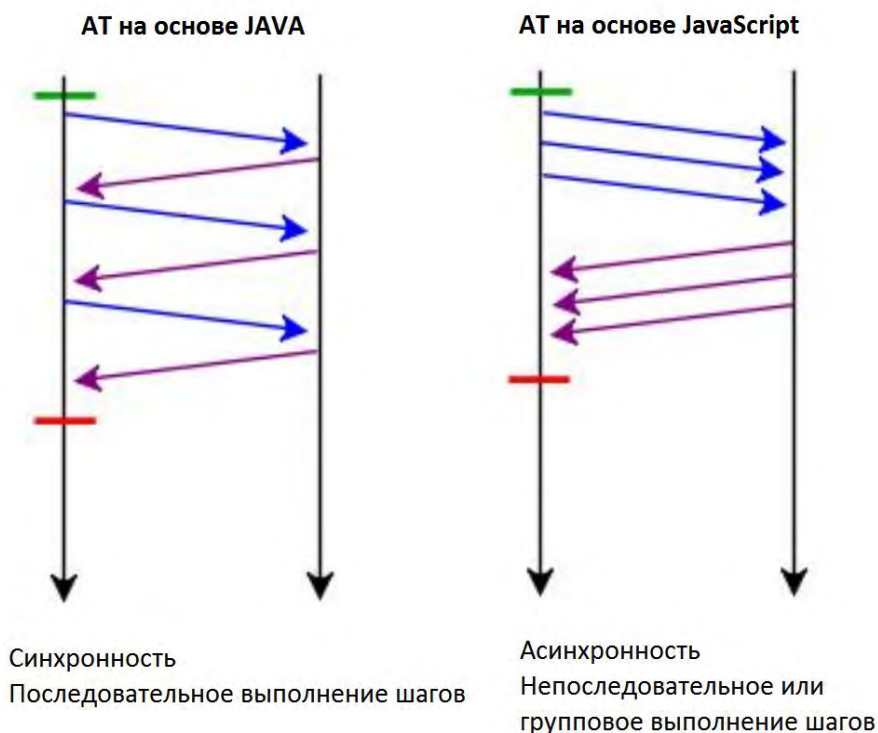


Рисунок 1 – Особенности автоматизации на Java и Javascript [1]

Главной отличительной особенностью Protractor Javascript является синхронизация с Angular-приложением, которая в общем виде выглядит так:

- 1) Существует драйвер, браузер и некоторое действие, которое вызывается из тестового приложения.
- 2) Protractor дожидается момента, когда загрузилась страница в браузере.
- 3) Дождается, когда загрузилось Angular-приложение на страницу.
- 4) Между каждым действием, которое имеется в WebDriverJS, выполняется скрипт, который обеспечивает саму синхронизацию с Angular-приложением.

Рассмотрим более подробно процесс синхронизации. Пусть имеется запрос со стороны тестового скрипта, который отправляется в драйвер. Драйвер через браузер дожидается загрузки страницы. После того, как загрузилась страница, ожидается загрузка Angular-приложения. Пока приложение выполняет какие-либо запросы на сервер, считается, что приложение не синхронизировано [2].

В Angular-приложении довольно много бизнес-логики заложено на стороне клиента, и часто складывается такая ситуация, когда необходимо получить какие-нибудь элементы на странице, однако нет возможности гарантировать, что эти элементы уже загрузились и пришли в необходимое для работы состояние. Поэтому Protractor выбирает тот момент, когда исходящих запросов нет, и Angular полностью визуализировал свое состояние, а также визуализировал все свои элементы, исходя из откликов, которые пришли от сервера. И как только этот процесс визуализации завершен, считается, что приложение синхронизировано. В качестве обратной связи в WebDriver получен ответ, и каждая операция в WebDriverJS сопровождается синхронизацией, которая представлена на рисунке 2.

Заключение

В процессе разработки любого приложения автор всегда стремится, чтобы оно выполняло ряд поставленных задач: работало согласно требованиям корректно, было востребовано на практике, имело удобный дружественный интерфейс. Все это невозможно

достичь без выбора правильных инструментов разработки и подходов тестирования.

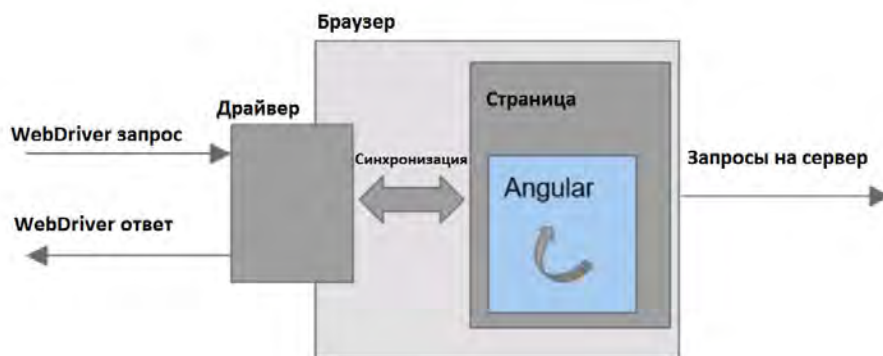


Рисунок 2 – Синхронизация с Angular приложением [3]

В данной работе рассмотрены различия автоматизации на Java и JavaScript, а также особенности использования Protractor в автоматизированном тестировании web-приложений на основе AngularJS, а именно: синхронизация с Angular-приложением, асинхронность, простота в установке и использовании.

В настоящее время автоматизация тестирования на основе JavaScript активно развивается и в будущем будет занимать лидирующую позицию на рынке тестирования web-приложений.

Список использованных источников

1. Protractor is an end-to-end test framework for AngularJS [Электронный ресурс]. – 2016. – Режим доступа: <http://www.protractortest.org/> – Дата доступа: 03.11.2016.
2. JavaScript framework AngularJS [Электронный ресурс]. – 2016. – Режим доступа: <https://angularjs.org/> – Дата доступа: 03.11.2016.
3. Protractor & AngularJS [Электронный ресурс]. – 2016. – Режим доступа: <https://egghead.io/> – Дата доступа: 03.11.2016.