**GEEKBRAINS**

**ФАКУЛЬТЕТ РАЗРАБОТЧИК-ТЕСТИРОВЩИК**

ВЫПУСКНАЯ ДИПЛОМНАЯ РАБОТА НА ТЕМУ:

**«Автоматизация тестирования с целью уменьшения трудозатрат при установке новой версии продукта на примере интернет-магазина одежды, обуви и аксессуров»**

Печерских Алёна Сергеевна

Самара, 2023 г.

# СОДЕРЖАНИЕ

[ВВЕДЕНИЕ 2](#_Toc139844797)

[1. ИНСТРУМЕНТЫ ДЛЯ АВТОМАТИЗИРОВАННОГО ТЕСТИРОВАНИЯ 4](#_Toc139844798)

[1.1. Selenium 4](#_Toc139844799)

[ЗАКЛЮЧЕНИЕ 8](#_Toc139844800)

[СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ И РЕСУРСОВ 9](#_Toc139844801)

[ПРИЛОЖЕНИЯ 10](#_Toc139844802)

# ВВЕДЕНИЕ

Цель данной дипломной работы по специальности «Разработчик-Тестировщик» – исследование инструментов для реализации автоматизации тестирования базовых сценариев использования продукта на примере интернет магазина lcwaikiki.kz, а также разработка приложения для автоматизированной валидации функционала, которое позволит уменьшить трудозатраты на проверку работоспособности веб-портала после обновления.

Автоматизация тестирования после установки новой версии продукта является критичной задачей в особенности для поддержки крупного продукта, т.к. позволяет избежать огромных затрат, которые имеют место быть в случае ручной проверки.

Исходя из цели, в дипломной работе поставлены и решены следующие задачи:

1. Рассмотреть основные способы взаимодействия с браузером. Выделить плюсы и минусы того или иного способа;
2. Выбрать инструмент взаимодействия с браузером для реализации автоматизированных тестов;
3. Ознакомиться с функционалом интернет-магазина lcwaikiki.kz, выделить базовые сценарии использования, которые должны тестироваться после установки новой версии продукта (веб-приложения);
4. Реализовать сценарии автоматизированного тестирования для проверки базового функционала lcwaikiki.kz с использованием Java;
5. Собрать финальный билд для запуска автотестов с помощью команды в Windows/Linux.

Задачи выше были решены с использованием следующих инструментов: Java как язык разработки приложения для автотестов, Intellij Idea как среда разработки, Playwright как инструмент для взаимодействия с браузером, Maven как инструмент для сборки финального приложения для запуска в Windows/Linux.

# ИНСТРУМЕНТЫ ДЛЯ АВТОМАТИЗИРОВАННОГО ТЕСТИРОВАНИЯ

## Selenium

Selenium – это проект, в рамках которого разрабатывается серия программных продуктов с исходным кодом (open source), открытым в декабре 2004 года [1, 2, 3]:

* Selenium WebDriver;
* Selenium RC;
* Selenium Server;
* Selenium Grid;
* Selenium IDE.

**Selenium WebDriver** – это программная библиотека для управления браузерами. Часто употребляется также более короткое название WebDriver. Это целое семейство драйверов для различных браузеров, а также набор клиентских библиотек на разных языках, позволяющих работать с этими драйверами.

Selenium WebDriver – это основной продукт, разрабатываемый в рамках проекта Selenium.

В рамках проекта Selenium разрабатываются драйверы для браузеров Firefox, Internet Explorer и Safari, а также драйверы для мобильных браузеров Android и iOS. Драйвер для браузера Google Chrome разрабатывается в рамках проекта Chromium, а драйвер для браузера Opera (включая мобильные версии) разрабатывается компанией Opera Software. Поэтому они формально не являтся частью проекта Selenium, распространяются и поддерживаются независимо. Но логически, конечно, можно считать их частью семейства продуктов Selenium.

Аналогичная ситуация и с клиентскими библиотеками – в рамках проекта Selenium разрабатываются библиотеки для языков Java, .Net (C#), Python, Ruby, JavaScript. Все остальные реализации не имеют отношения к проекту Selenium, хотя, возможно, в будущем, какие-то из них могут влиться в этот проект.

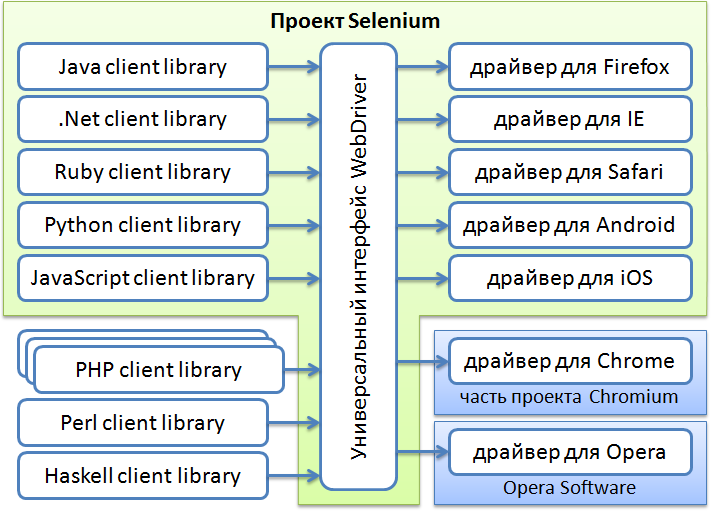


Рисунок 1 – Структура Selenium WebDriver

**Selenium RC** – это предыдущая версия библиотеки для управления браузерами. Аббревиатура RC в названии этого продукта расшифровывается как Remote Control, то есть это средство для «удалённого» управления браузером.  
Эта версия с функциональной точки зрения значительно уступает WebDriver. Сейчас она находится в законсервированном состоянии, не развивается и даже известные баги не исправляются. А всем, кто сталкивается с ограничениями Selenium RC, предлагается переходить на использование WebDriver.

Иногда Selenium RC называется также Selenium 1.0, тогда как WebDriver называется Selenium 2.0. Хотя на самом деле дистрибутив версии 2.0 включает в себя одновременно обе реализации – и Selenium RC, и WebDriver. А вот когда выйдет версия 3.0 – в ней останется только WebDriver.

С технической точки зрения WebDriver не является результатом эволюционного развития Selenium RC, они построены на совершенно разных принципах и у них практически нет общего кода. Объединяет их лишь тот факт, что обе реализации были сделаны в рамках проекта Selenium. Ну, или если быть совсем точным, WebDriver сначала был самостоятельным проектом, но в 2008 году произошло слияние и сейчас WebDriver представляет собой основной вектор развития проекта Selenium.

**Selenium Server –** это сервер, который позволяет управлять браузером с удалённой машины, по сети. Сначала на той машине, где должен работать браузер, устанавливается и запускается сервер. Затем на другой машине (технически можно и на той же самой, конечно) запускается программа, которая, используя специальный драйвер RemoteWebDriver, соединяется с сервером и отправляет ему команды. Он в свою очередь запускает браузер и выполняет в нём эти команды, используя драйвер, соответствующий этому браузеру.

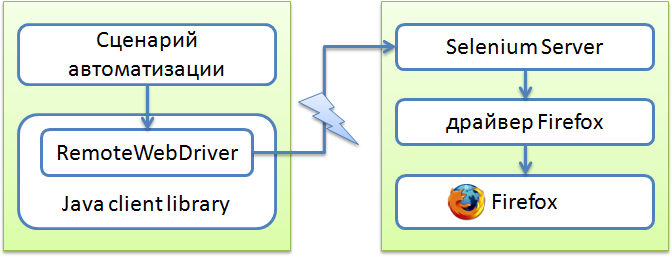


Рисунок 2 – Логика работы Selenium Server

Selenium Server поддерживает одновременно два набора команд – для новой версии (WebDriver) и для старой версии (Selenium RC).

**Selenium Grid –** это кластер, состоящий из нескольких Selenium-серверов. Он предназначен для организации распределённой сети, позволяющей параллельно запускать много браузеров на большом количестве машин. Selenium Grid имеет топологию «звезда», то есть в его составе имеется выделенный сервер, который носит название «хаб» или «коммутатор», а остальные сервера называются «ноды» или «узлы». Сеть может быть гетерогенной, то есть коммутатор и узлы могут работать под управлением разных операционных систем, на них могут быть установлены разные браузеры. Одна из задач Selenium Grid заключается в том, чтобы «подбирать» подходящий узел, когда во время старта браузера указываются требования к нему – тип браузера, версия, операционная система, архитектура процессора и ряд других атрибутов.

Ранее Selenium Grid был самостоятельным продуктом. Сейчас физически продукт один – Selenium Server, но у него есть несколько режимов запуска: он может работать как самостоятельный сервер, как коммутатор кластера, либо как узел кластера, это определяется параметрами запуска.

**Selenium IDE –** плагин к браузеру Firefox, который может записывать действия пользователя, воспроизводить их, а также генерировать код для WebDriver или Selenium RC, в котором выполняются те же самые действия. В общем, это «Selenium-рекордер».

Тестировщики, которые не умеют (или не хотят) программировать, используют Selenium IDE как самостоятельный продукт, без преобразования записанных сценариев в программный код. Это, конечно, не позволяет разрабатывать достаточно сложные тестовые наборы, но некоторым хватает и простых линейных сценариев. [1]

# ЗАКЛЮЧЕНИЕ

# СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ И РЕСУРСОВ

1. <https://habr.com/ru/articles/152653/>
2. https://ru.wikipedia.org/wiki/Selenium
3. https://www.selenium.dev/

# ПРИЛОЖЕНИЯ