Обзор инструментов для автотестирования

1. Postman

Postman предназначен для проверки запросов с клиента на сервер и получения ответа от бэкенда. Основная функциональность Postman — это создание коллекций с запросами к API и создание автотестов на основе этих запросов. Код пишется на JavaScript. Postman также предлагает множество готовых сниппетов, которые можно применить для тестирования API. Здесь можно валидировать коды и содержание ответов, парсить и сохранять значения в переменные окружения или глобальные переменные, проверять их соответствие заданным значениям и т.д.



2. SOAP UI

Это приложение с открытым исходным кодом для тестирования [веб-сервисо](https://ru.wikipedia.org/wiki/Веб-сервис)в сервис-ориентированных архитектур (SOA) и передачи состояний представлений ([REST](https://ru.wikipedia.org/wiki/REST)). Его функциональные возможности включают проверку веб-службы, запуск, разработку, моделирование и макетирование, функциональное тестирование, тестирование нагрузки и соответствия. Подобно Postman в SOAP UI можно создавать тесты на JavaScript или использовать готовые сниппеты.

3. Selenium Web Driver 

Это инструмент для автоматизации реального браузера, как локально, так и удаленно, наиболее близко имитирующий действия пользователя. Он напрямую вызывает команды браузера, используя родной для каждого конкретного браузера API. Как совершаются эти вызовы и какие функции они выполняют зависит от конкретного браузера. Таким образом, Webdriver использует способ взаимодействия с браузером более близкий к действиям реального пользователя.

Для работы с Webdriver необходимо 3 основных программных компонента:

1. Браузер, работу которого пользователь хочет автоматизировать. Это реальный браузер определенной версии, установленный на определенной ОС и имеющий свои настройки (по умолчанию или кастомные).
2. Для управления браузером необходим driver браузера. Driver на самом деле является веб сервером, который запускает браузер и отправляет ему команды, а также закрывает его. У каждого браузера свой driver. Связано это с тем, что у каждого браузера свои отличные команды управления и реализованы они по-своему. Найти список доступных драйверов и ссылки для скачивания можно [на официальном сайте](http://docs.seleniumhq.org/download/) [Selenium](https://kreisfahrer.gitbooks.io/selenium-webdriver/content/GLOSSARY.html#selenium) проекта.
3. Скрипт/тест, который содержит набор команд на определенном языке программирования для драйвера браузера. Такие скрипты используют [Selenium Webdriver](https://kreisfahrer.gitbooks.io/selenium-webdriver/content/GLOSSARY.html#selenium_webdriver) bindings (готовые библиотеки), которые доступны пользователям на различных языках.

4. Cucumber 

Это один из самых популярных фреймворков для автоматизации тестирования с использованием BDD-подхода.

Cucumber-проект состоит из двух частей – это текстовые файлы с описанием сценариев (\*.feature) и файлы с реализацией шагов на одном из популярных языков программирования (например, Java).

BDD тесты – это простой текст, на человеческом языке, написанный в форме истории (сценария), описывающей некоторое поведение.  
В Cucumber для написания тестов используется Gherkin-нотация, которая определяет структуру теста и набор ключевых слов. Тест записывается в файл с расширением \*.feature и может содержать как один, так и более сценариев.



5. TestNG

Это один из наиболее популярных тестовых фреймворков для языка программирования Java, основанный на JUnit(Java) и Nunit (C#).

TestNG предназначен для:

• unit тестирования;

• функционального тестирования;

• интеграционного тестирования;

• end-to-end тестирования.

TestNG написан на Java и может быть использован как с языком Java, так и с другими основанными на Java языками (например, Groovy), а также позволяет пользователям создавать конфигурации тестов через XML файлы и разрешает включать или исключать из них различные пакеты, классы и методы в тестовых наборах.