



УНИВЕРСИТЕТ ИТМО

ФГАОУ ВО «НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО»

ФАКУЛЬТЕТ ПРОГРАММНОЙ ИНЖЕНЕРИИ И КОМПЬЮТЕРНОЙ ТЕХНИКИ

ТЕСТИРОВАНИЕ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

Лабораторная работа №3

Вариант 661

Лабушев Тимофей

Группа Р3402

Санкт-Петербург

2020

Задание

Сформировать варианты использования, разработать на их основе тестовое покрытие и провести функциональное тестирование интерфейса сайта согласно варианту:

Вариант №661: ADVEGO. Копирайтинг, рерайтинг, переводы. - <http://advego.ru/>

1. Тестовое покрытие должно быть сформировано на основании набора прецедентов использования сайта.
2. Тестирование должно осуществляться автоматически - с помощью системы автоматизированного тестирования Selenium.
3. Тесты должны исполняться в браузерах Firefox и Chrome.
4. Предполагается, что тестируемый сайт использует динамическую генерацию элементов на странице, т.е. выбор элемента в DOM должен осуществляться не на основании его ID, а с помощью XPath.

Диаграмма прецедентов использования сайта

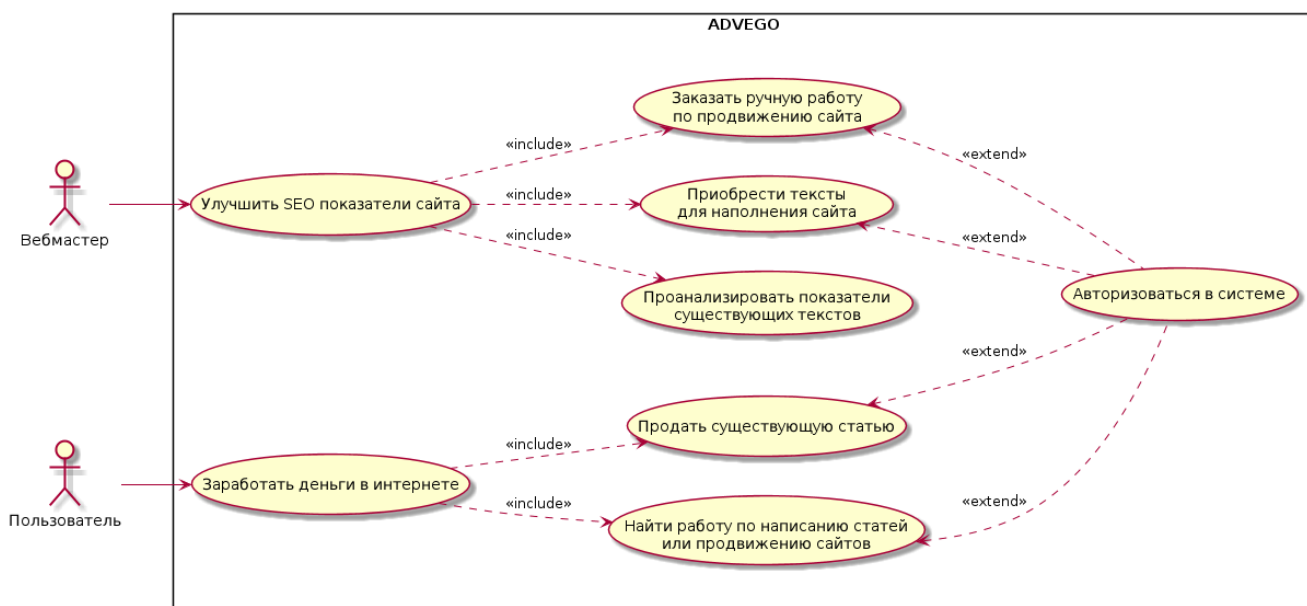


Рисунок 1. Диаграмма прецедентов использования тестируемого сайта

Описание тестового покрытия

Согласно прецедентам использования сайта были разработаны следующие сценарии тестирования:

Прецедент (класс сценариев)	Название и описание сценариев тестирования
Проанализировать показатели существующих текстов (SeoAnalysisTest)	<ul style="list-style-type: none">• <code>it spell checks user-provided texts in russian</code> Анализатор текстов находит опечатки в текстах на русском языке• <code>it spell checks user-provided texts in english</code> Анализатор текстов находит опечатки в текстах на английском языке• <code>it provides character and word statistics</code> Анализатор текстов составляет базовую статистику (количество символов и слов)• <code>it analyzes word frequencies ignoring stop words</code> Анализатор текстов составляет таблицу ключевых слов (исключая стоп-слова)• <code>it shows an error message when the text is empty</code> Анализатор показывает осмысленное сообщение об ошибке при запуске без текста
Приобрести тексты для наполнения сайта (MarketplaceTest)	<ul style="list-style-type: none">• <code>it allows users to filter articles by keywords</code> Пользователь может искать выставленные на продажу статьи по ключевым словам• <code>it allows users to sort articles by price</code> Пользователь может сортировать выставленные на продажу статьи по цене• <code>it allows users to add articles to cart</code> Пользователь может добавлять статьи в корзину
Заказать ручную работу по продвижению сайта (OrderCreationTest)	<ul style="list-style-type: none">• <code>it shows default order parameters when the page is opened</code> При открытии страницы заказ уже заполнен шаблонными значениями• <code>it fills in order parameters based on the selected template</code> Параметры заказа автоматически изменяются при выборе другого шаблона• <code>it allows users to search through templates</code> Пользователь может искать подобрать шаблон заказа, используя поиск

Найти работу по написанию статей или продвижению сайтов (JobSearchTest)	<ul style="list-style-type: none"> • it allows users to sort jobs by price Пользователь может сортировать доступные заявки по оплате за выполнение • it allows users to filter jobs by category Пользователь может искать работу в подходящих ему категориях
Продать существующую статью (ListingCreationTest)	<ul style="list-style-type: none"> • it allows users to submit new articles to marketplace Созданная пользователем статья проверяется на соответствие базовым критериям (минимальная длина текста, правильно выставленные метаданные) и помещается в магазин
Авторизоваться в системе (SignInTest)	<ul style="list-style-type: none"> • it allows users to sign in via the login popup Пользователь может войти в систему, открыв модальное окно авторизации и введя данные своей учетной записи • it shows an error message when entering invalid credentials Система показывает осмысленное сообщение об ошибке при вводе неверных данных учетной записи

Результаты тестирования

В ходе тестирования было разработано 18 тестовых сценариев, каждый из которых выполняется в браузерах Chrome (с помощью ChromeDriver) и Firefox (с помощью GeckoDriver).

Полностью автоматическое тестирование оказалось нереализуемым из-за применения на различных страницах сайта системы geCAPTCHA для защиты от ботов. Полученные сценарии можно считать *автоматизированными*: при запуске каждого из них открывается новое окно браузера, и в случае появления запроса ввода капчи тестирующий должен вручную ответить на него. По прохождении теста окно браузера закрывается.

Помимо этого, тестовое покрытие существенно ограничивается отсутствием доступа к привилегированным учетным записям. К примеру, новое объявление о работе размещается публично не моментально, а лишь после прохождения ручной проверки, что подразумевает наличие роли *модератора*. Однако составить для этой роли прецеденты и разработать автоматизированные тесты не представляется возможным, так как интерфейс модерации полностью закрыт для стороннего тестирующего.

Исходный код

<https://github.com/timlathy/itmo-fourth-year/tree/master/Software-Testing-7th-Term/Lab3>

Выводы

В ходе выполнения работы была рассмотрена система автоматизации действий веб-браузера *Selenium* в контексте написания функциональных тестов интернет-сайтов с пользовательским взаимодействием.

При написании тестов был изучен шаблон *page object*, который позволяет отделить логику тестов от разметки веб-страницы, что облегчает дальнейшую поддержку тестовых сценариев.

Был также сделан вывод, что стороннее функциональное тестирование веб-приложений малоэффективно. Отсутствие знаний о системе и трудности получения доступа к ней уменьшают возможное тестовое покрытие. Подобные условия тестирования подходят скорее для аудитов безопасности системы, в частности, довольно эффективного метода испытания на проникновение (*penetration testing*).