**Министерство науки и высшего образования Российской Федерации**

Федеральное государственное бюджетное образовательное учреждение

высшего образования

**«Томский государственный университет систем управления и радиоэлектроники» (ТУСУР)**

Управление дополнительного образования института инноватики (УДО ИИ)

**ПРОЕКТ**

**РАЗРАБОТКА СТРАТЕГИИ ТЕСТИРОВАНИЯ**

ВЕБ-ПРИЛОЖЕНИЯ

по программе профессиональной переподготовки:

«Тестирование и контроль качества программного обеспечения»

Выполнил(а):

Радионов И.И. \_\_\_\_\_

Подпись\_\_\_\_\_\_\_\_\_\_\_

«\_\_\_» \_\_\_\_\_\_\_ 2023 г.

Руководитель:

доцент каф. АОИ,

канд. техн. наук

Морозова Ю. В.

Подпись\_\_\_\_\_\_\_\_\_\_\_

«\_\_\_» \_\_\_\_\_\_\_ 2023 г.

Томск, 2023

Оглавление

[Введение 3](#_Toc126701160)

[1 Планирование и тест-дизайн 4](#_Toc126701161)

[**1.1 Описание проекта и постановка задач** 4](#_Toc126701162)

[**1.2 Разработка стратегии тестирования** 6](#_Toc126701163)

[2. Тестирование 8](#_Toc126701164)

[**2.1 Функциональное тестирование** 8](#_Toc126701165)

[**2.2 Нефункциональное тестирование** 14](#_Toc126701166)

[3. Результаты тестирования и выводы по результатам испытаний 21](#_Toc126701167)

[3.1 Результаты функционального тестирования 21](#_Toc126701168)

[3.2 Результаты нефункционального тестирования 22](#_Toc126701169)

[3.3 Выводы по итогам испытаний 24](#_Toc126701170)

[Заключение 25](#_Toc126701171)

[Список используемых источников 26](#_Toc126701172)

[Приложение А 27](#_Toc126701173)

[приложение Б 28](#_Toc126701174)

# Введение

В наши дни во времена активного развития информационных технологий у человека есть потребность к быстрому и удобному доступу к информации. Высокий спрос рождает и высокое предложение, что привело к возникновению на рынке огромного количества веб-ресурсов. Современные мобильные девайсы, такие как ноутбуки, ультрабуки, смартфоны и пр. позволяют обеспечить человека информацией, однако одного лишь быстрого доступа недостаточно, ведь информационные источники должны быть не только удобны, но ещё и комфортны для восприятия. К тому же из-за высокого спроса появляются сжатые сроки для разработки того или иного продукта, чтобы успеть выйти в релиз как можно быстрее, а это может существенно снизить конечное качество. По этой причине и набирает популярность и необходимость тестирование программного обеспечения.

Специалисты по тестированию и контролю качества программного обеспечения (ПО) на сегодняшний день широко востребованы. Они занимаются вопросами поиска и локализации дефектов, способных как снизить качество, так и ограничить, а то и вовсе урезать часть функционала того или иного ПО, тем самым нанести репутационный вред компании-разработчику, не говоря уже о причинённых неудобствах конечному потребителю.

Современному специалисту в области тестирования необходимо понимать, что наряду с количеством и разнообразием того контента, который реализуют компании-разработчики, существует и широкий спектр инструментов для его тестирования. В рамках программы профессиональной переподготовки: «Тестирование и контроль качества программного обеспечения» предлагалось познакомиться как с методами ручного тестирования, так и средствами автоматизации, и в данной работе при разработке стратегии тестирования будут рассмотрены некоторые из них.

# 1 Планирование и тест-дизайн

**1.1 Описание проекта и постановка задач**

В рамках итогового проекта по программе профессиональной переподготовки: «Тестирование и контроль качества программного обеспечения» необходимо разработать стратегию тестирования веб-приложения: сайт Стройпарк - <https://stroyparkdiy.ru/>.

Данное веб-приложение представляет из себя онлайн-магазин строительных и отделочных материалов, строительной и бытовой техники. Как и все приложения такого типа, обладает соответствующим функционалом, таким как: каталог товаров с категориями, поиск с фильтром, личный кабинет, корзина, функционал сравнения товаров, а также быстрая навигация и другие сопутствующие возможности. Для удобства основные функции и меню веб-приложения представлены на интеллект-карте (рисунок 1), собранной с помощью инструмента [1]. Для рассматриваемого веб-приложения необходимо провести тестирование основного функционала, также не забыть об удобстве и корректности его отображения и работоспособности, то есть нефункциональном тестировании.

Кроме того, согласно поставленным техническим условиям, а в нашем случае учебного задания, в приложении должен быть реализован ряд определённых особенностей и нюансов:

***1.*** ***«Дружественный» графический интерфейс программы*** должен быть удобен, приятен и интуитивно понятен для подавляющего большинства пользователей.

***2.*** ***Переход по логотипу на главную страницу***, позволяющий быстро вернуться к главной стартовой странице сайта в случае необходимости.

***3.*** ***Адаптивная вёрстка***, изменяющаяся в зависимости от поведения пользователя, настроек его платформы, размеров и ориентации экрана, является чуть ли не самым главным в современной разработке приложений для повышения удобства их использования.

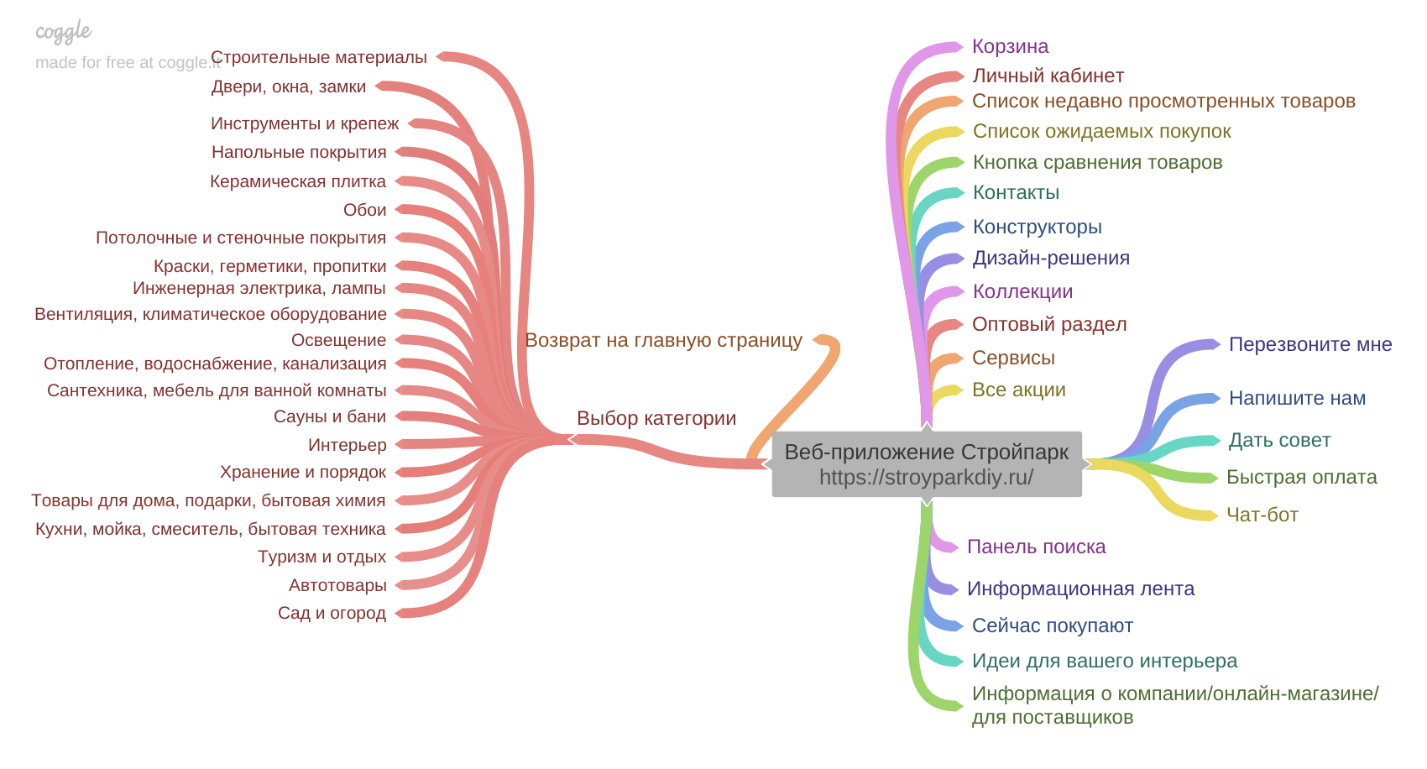


Рисунок 1 – Интеллект-карта веб-приложения Стройпарк

***4.*** ***Поддержка мобильной версии*** является также неотъемлемой частью разработки современных веб-приложений, поскольку уже на протяжении нескольких лет смартфоны являются частью жизни человека и находятся у нас под рукой в режиме 24/7.

***5.*** ***Поддержка актуальных операционных систем***, таких как Windows 7/8/10, macOS 10.14/10.15 или ***кроссплатформенность*** является также важнейшей особенностью в современной разработке приложений и плавно вытекает из предыдущей позиции.

***6.*** ***Поддержка браузеров Chrome, Yandex, Firefox, Edge, IE*** последних и предпоследних версий или ***кроссбраузерность*** также является обязательным и необходимым условием разработки веб-приложений.

***7. Поддержка правильной работоспособности на основных разрешениях экрана:*** 800x600, 1280х720, 1920х1080, поскольку большинство устройств поддерживает именно эти форматы.

***8. Поддержка версии для слабовидящих и страдающих цветовой слепотой*** для обеспечения доступа к ресурсу людям с ограниченными возможностями.

Таким образом, в основе тестирования проектного продукта должны рассматриваться вышеперечисленные особенности веб-приложения.

Чтобы решить, с чего начать тестирование, необходимо понять, для чего предназначено рассматриваемое веб-приложение. Онлайн-магазин предлагает приобрести определённый товар, соответственно необходимо начать с тестирование функционала приобретения этого самого товара. В основной функционал веб-сайта Стройпарка входит:

- каталог товаров и навигация по категориям;

- сравнение выбранных товаров по характеристикам;

- личный кабинет;

- корзина и оформление заказа;

- предоставление информации о бонусных программах, условиях оформления, способах оплаты, доставке, обмену, возврату и пр.

- а также возврат на главную страницу, как отдельный инструмент для удобства использования сайта.

Наряду с тестированием основных функций веб-приложения необходимо рассмотреть вопросы нефункционального характера. Для сайта <https://stroyparkdiy.ru/> можно провести:

- тестирование валидации;

- тестирование адаптивной вёрстки, в том числе и под разные разрешения окна или экрана;

- поддержку мобильной версии;

- поддержку кроссбраузерности и кроссплатформенности;

- нагрузочное тестирование;

- тестирование скорости загрузки сайта;

- тестирование корректности локализации и поддержке нескольких языковых версий, если таковые имеются;

- поддержку версии для слабовидящих и страдающих цветовой слепотой;

- проверку дружелюбности внешнего вида, читабельность и интуитивность интерфейса.

Именно вышеперечисленные особенности планируется протестировать в рамках проектной работы.

## **1.2 Разработка стратегии тестирования**

Учитывая поставленные задачи и требования к продукту необходимо определиться с основными видами тестирования и определиться с приоритетом тестовых задач и времени, необходимому для их выполнения.

Проект рассчитан на две недели. За этот период осуществлялось планирование работы, разрабатывались тест-дизайн и тест-кейсы с дальнейшим их выполнением. Также подготавливались автотесты там, где это было возможно и уместно, с последующей их реализацией. В конечном итоге анализировались результаты и подводились итоги.

Обо всех применяемых техниках и инструментах подробнее рассказано в следующем разделе по мере их появления и использования.

В качестве основных тестовых сред использовалась операционная система Windows 10 и браузеры Google Chrome, Mozilla FireFox и Internet Explorer. Учитывая столь сжатые сроки, пришлось отказаться от тестирования веб-приложения в операционных системах macOS и его нативном браузере Safari, впрочем также, как и других менее популярных браузерах.

Для наглядности все основные особенности условий и реализации проекта представлены в одностраничном тест-плане [2] на рисунке 2.

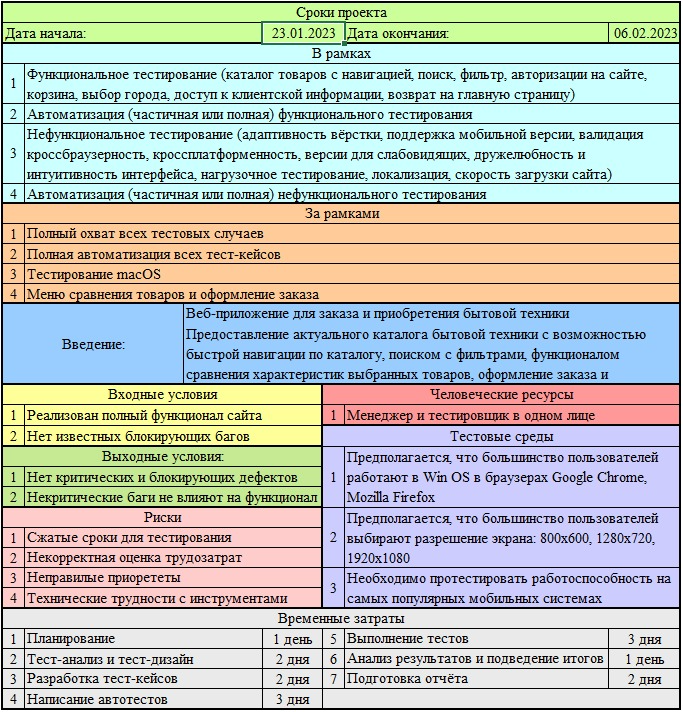


Рисунок2 – Одностраничный тест-план проекта

# 2. Тестирование

**2.1 Функциональное тестирование**

Согласно стратегии и тест-плану, тестирование начинается с проверки рабочего функционала. В первую очередь после запуска сайта пользователь проходит в каталог товаров или поиск. Начнём с тестирования каталога товаров. Далее посмотрим функционал поиска, глянем на возможности настройки фильтров, протестируем корзину, попробуем авторизоваться в личном кабинете, оценим доступ к клиентской информации и попутно проведём проверку возврата на главную страницу.

***Тестирование каталога***

*Цель:* проверка актуальности навигации по каталогу и кликабельность соответствующих ссылок в категориях.

*Ручное тестирование:* проведено.

*Возможна автоматизация:* да.

Тест-кейс (на примере категории «Строительные материалы»)

|  |  |
| --- | --- |
| Шаги | Ожидаемый результат |
| 1. Открыть главную страницу сайта <https://stroyparkdiy.ru/> | Страница загрузилась |
| 2. Навести курсор мыши на категорию «Строительные материалы» и нажать | Открылось подменю с выбором подкатегории |
| 3. Навести курсор мыши и нажать на название ключевой интересующей подкатегории «Строительные материалы» | Открылась плитка с подкатегориями «Строительные материалы» |
| 4. Нажать на категорию «Кровля» | Открывается список товаров категории |
| 5. Нажать на интересующий тип товара | Откроется список товаров с фильтром и сортировкой в начале списка. |
| 6. Перейти на предыдущую страницу | Переход осуществлён, открылась плитка с подкатегориями «Кровля» |
| 7. Повторить шаги 5-6 для каждой подкатегории, если открывается дополнительная плитка с подкатегориями, то повторяются шаги для них, пока не дойдёт до конца списка. | Открывается список товаров категории с фильтром и сортировкой в начале списка. |
| 7. Осуществить переход на главную страницу через специальную кнопку в верхнем левом углу сайта | Переход успешно осуществлён |

***Тестирование поиска***

*Цель:* проверка работоспособности поисковой строки и выдачи актуальных результатов.

*Ручное тестирование:* проведено.

*Возможна автоматизация:* да.

Тест-кейс (на примере запроса «шайба»)

|  |  |
| --- | --- |
| Шаги | Ожидаемый результат |
| 1. Открыть главную страницу сайта <https://stroyparkdiy.ru/> | Страница загрузилась |
| 2. Нажать на поисковую строку в верхней части сайта | Появляется курсор |
| 3. Ввести интересующие наименование товара «шайба» | Поиск предлагает наиболее популярные запросы по ключевому слову или даже связанные категории |
| 4. Нажать Enter и выполнить поиск товаров по запросу «шайба» | Открывается список доступных товаров по запросу «шайба» с фильтром |
| 5. Нажать на поисковую строку | Отобразится история запросов |
| 6. Нажать на кнопку «Изменить запрос» | История запросов очистится |
| 7. Осуществить переход на главную страницу через специальную кнопку в верхнем левом углу сайта | Переход успешно осуществлён |

***Тестирование фильтра***

Цель: проверка корректной работоспособности фильтра для более детальной пользовательской настройки поискового запроса.

Ручное тестирование: проведено.

Возможна автоматизация: да.

Для тестирования работы фильтра необходимо выбрать одну из категорий поиска. В примере показана категория «Ночники». С помощью техники попарного тестирования [3] подбираются уникальные комбинации параметров фильтра, целью которого является сокращение количества тестовых случаев (таблица 1).

Таблица 1 – Параметры фильтра, используемые для инструмента попарного тестирования

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | 1 | 2 | 3 | 4 | 5 | 6 |
| Цена | Менее 500 | 501-1000 | 1001-1500 | 1501 и более |  |  |
| Мощность(Вт) | 0,5 | 1 | 3 |  |  |  |
| Источник питания | USB | батарейки | сеть |  |  |  |
| Цвет | белый | желтый | розовый | синий |  |  |
| Производитель | APEYRON ELECTRICS | ARTE LAMP | RISALUX |  |  |  |

Как видно из таблицы, в качестве основных категорий для тестирования выбирались:

1.Цена (4 варианта)

2.Мощность (3 варианта)

3.Источник питания (3 варианта)

4.Цвет (4 варианта)

5.Производитель (3 варианта)

При переборе всех исходных данных генератор предложил 432 вариантов уникальных парных комбинаций. Результаты комбинаторики попарных значений позволили существенно сократить минимально необходимое количество вариантов для тестирования до 16 (Приложение А).

Тест-кейс (универсальный)

|  |  |
| --- | --- |
| Шаги | Ожидаемый результат |
| 1. Открыть главную страницу сайта <https://stroyparkdiy.ru/> | Страница загрузилась |
| 2. Навести курсор мыши на категорию «Освещение» и нажать | Открылась плитка с подкатегориями «Освещение» |
| 3. Навести курсор мыши и нажать на название «ночники» | Открывается список доступных товаров категории «ночники» с фильтром |
| 4. Задаём значения фильтра согласно сгенерированному совпадению из таблицы 1 и инициируем поиск | Значения введены, поисковик выдал значения согласно запросу |

***Тестирование личного кабинета***

*Цель:* проверка корректности работы авторизации на сайте.

*Ручное тестирование:* проведено.

*Возможна автоматизация:* да.

В данном разделе тестировались не все функции личного кабинета, а только возможности авторизации. При тестировании авторизации предусматривались как позитивные, так и негативные тест-кейсы.

Негативный тест-кейс при авторизации

|  |  |
| --- | --- |
| Шаги | Ожидаемый результат |
| 1. Открыть главную страницу сайта <https://stroyparkdiy.ru/> | Страница загрузилась |
| 2. Нажать на кнопку «Войти» | Появляется окно авторизации/регистрации |
| 3. Ввести произвольные данные в поле «e-mail» и поле «Пароль» | Появится ошибка «E-mail/пароль указан неверно» |

Позитивный тест-кейс с авторизацией

|  |  |
| --- | --- |
| Шаги | Ожидаемый результат |
| 1. Открыть главную страницу сайта <https://stroyparkdiy.ru/> | Страница загрузилась |
| 2. Нажать на кнопку «Войти» | Появляется окно авторизации/регистрации |
| 3. Ввести в каждое из полей актуальную информацию и нажать кнопку «Войти» | Авторизация проходит успешно |

Негативный тест-кейс с авторизацией по паролю

|  |  |
| --- | --- |
| Шаги | Ожидаемый результат |
| 1. Открыть главную страницу сайта <https://stroyparkdiy.ru/> | Страница загрузилась |
| 2. Нажать на кнопку «Войти» | Появляется окно авторизации/регистрации |
| 3. Ввести корректно E-mail, но произвольно вести пароль и нажать на кнопку «Войти» | Появится ошибка «E-mail/пароль указан неверно» |

Негативный тест-кейс с авторизацией по E-mail

|  |  |
| --- | --- |
| Шаги | Ожидаемый результат |
| 1. Открыть главную страницу сайта <https://stroyparkdiy.ru/> | Страница загрузилась |
| 2. Нажать на кнопку «Войти» | Появляется окно авторизации/регистрации |
| 3. Ввести произвольно E-mail, но вести корректно пароль и нажать на кнопку «Войти» | Появится ошибка «E-mail/пароль указан неверно» |

Позитивный тест-кейс с восстановлением пароля

|  |  |
| --- | --- |
| Шаги | Ожидаемый результат |
| 1. Открыть главную страницу сайта <https://stroyparkdiy.ru/> | Страница загрузилась |
| 2. Нажать на кнопку «Войти» | Появляется окно авторизации/регистрации |
| 3. Нажать кнопку «Забыли пароль?» | Появиться окно запроса E-mail |
| 5. Ввести в поле e-mail и нажать кнопку «Выслать новый пароль» | На почту придёт письмо с ссылкой на восстановление пароля |
| 6. Пройти по ссылке | Выполнится переход на страницу сайта. |
| 7. Нажать на кнопку войти | Откроются настройки в личном кабинете |
| 8. Вести новый пароль | Корректный вод нового пароля |
| 9. Нажать кнопку сохранить | Переход в личный кабинет. |

***Тестирование корзины***

*Цель:* проверка корректности работы функций корзины.

*Ручное тестирование:* проведено.

*Возможна автоматизация:* да.

Тест-кейс с проверкой функционала корзины

|  |  |
| --- | --- |
| Шаги | Ожидаемый результат |
| 1. Открыть главную страницу сайта <https://stroyparkdiy.ru/> | Страница загрузилась |
| 2. Любым доступным способом открыть список товаров (поиск или через категории) | Список открыт, имеется перечень товаров |
| 3. Напротив любого товара нажать кнопку «Купить» | Верхнем правом углу, в окне корзина, измениться количество товаров и цена |
| 4. Повторить шаг 3 с любыми товарами четыре раза | Шаги пройдены, товары добавлены, кнопка «Купить» напротив товара изменится на «В корзине» |
| 5. Перейти в корзину через меню в верхней части экрана со значком корзины и набранной предварительной суммой заказа или через кнопку «В корзине» у любого из добавленных товаров | Переход в корзину совершён |
| 6. Удалить один любой товар, уменьшив его количество до 0 нажатием на кнопку «-» | Товар удаляется из корзины, и цена успешно пересчитается |
| 7. Нажать «Оформить заказ» | Переход к способу оплаты |

***Тестирование доступа к клиентской информации***

*Цель:* проверка соответствия ссылок и страниц с пользовательской информацией.

*Ручное тестирование:* проведено.

*Возможна автоматизация:* сделана.

Протестировать соответствие ссылок и страниц с пользовательской информацией можно как вручную, так и автоматизированно. Для решения вопроса автоматизации применялась среда разработки Eclipse IDE [4]. Для определения нужных селекторов применялся инструмент Selenium IDE [5], а запускались тесты с помощью инструмента Selenium WebDriver [6].

Тест-кейс для проверки доступа к клиентской информации

|  |  |
| --- | --- |
| Шаги | Ожидаемый результат |
| 1. Открыть главную страницу сайта <https://stroyparkdiy.ru/> | Страница загрузилась |
| 2. Внизу страницы, в разделе «Интернет-магазин» нажать на кнопку «Как заказать» | Откроется страница «Как сделать заказ» |
| 3. Осуществить переход на главную страницу через специальную кнопку в верхнем левом углу сайта | Откроется главная страница |
| 4. Повторить Шаг 2 для кнопки «Как оплатить» | Откроется страница «Оплата товара» |
| 5. Повторить Шаг 3 | Откроется главная страница |
| 6. Повторить Шаг 2 для кнопки «Самовывоз» | Откроется страница «Как сделать заказ на самовывоз из магазина» |
| 7. Повторить Шаг 3 | Откроется главная страница |
| 8. Повторить Шаг 2 для кнопки «Возврат товара» | Откроется страница «Возврат/обмен товара» |

Листинг с кодом для автоматизированного тестирования представлен в Приложении А.

**2.2 Нефункциональное тестирование**

Поскольку к нефункциональному тестированию относится всё то, что никак не влияет на работоспособность основных функций приложения, то на данном этапе предстоит провести тестирование ряда особенностей. В данном разделе рассматривались: тестирование страницы на соответствие стандартов при разработке веб-приложений, адаптивность вёрстки при разном рабочем разрешении, тестирование поддержки мобильных устройств, кроссбраузерность и кроссплатформенность, нагрузочное тестирование, оценка скорости загрузки сайта, тестирование локализации, тестирование версий для слабовидящих и страдающих цветовой слепотой, а также дружелюбность и интуитивность интерфейса.

***Тестирование валидации***

*Цель:* проверка соответствия веб-приложения стандартам разработки сайтов.

*Ручное тестирование:* проведено.

*Возможна автоматизация:* да.

Основные принципы и стандарты разработки веб-приложений изложены в [7]. В рамках данной работы использовался инструмент [8].

Тест-кейс для проведения валидации сайта

|  |  |
| --- | --- |
| Шаги | Ожидаемый результат |
| 1. Открыть инструмент для валидации: <https://validator.w3.org/nu/> | Сайт загрузился, страница открыта |
| 2. В поле для проверки адреса ввести <https://stroyparkdiy.ru/> | Адрес вписан |
| 3. Нажать на кнопку «Check» | Запустится процесс проверки сайта на соответствие стандартов, и появится результат проверки в виде отчёта |

***Тестирование адаптивности вёрстки и разрешения***

*Цель:* проверка адаптивности сайта под различные разрешения и ориентации экрана.

*Ручное тестирование:* проведено.

*Возможна автоматизация:* нет.

Проверку адаптивности интерфейса сайта можно осуществить с помощью использованного инструмента DevTools [9].

Тест-кейс для проверки адаптивности вёрстки

|  |  |
| --- | --- |
| Шаги | Ожидаемый результат |
| 1. Открыть тестируемый сайт: <https://stroyparkdiy.ru/> | Сайт загрузился, страница открыта |
| 2. Запустить инструмент DevTools нажатием клавиши F12 | Открылось окно инструмента |
| 3. Выбрать режим адаптивного дизайна сочетанием клавиш Ctrl+Shift+M | Открылся интерфейс адаптивного дизайна |
| 4. Ввести разрешение экрана 800x600 | Разрешение сменится, сайт открывается правильно, все элементы отображаются корректно, нигде нет смещений |
| 5. Изменить ориентацию экрана с вертикальной на горизонтальную | Разрешение сменится, сайт открывается правильно, все элементы отображаются корректно, нигде нет смещений |
| 6. Повторить шаги 4-5 для разрешения экрана 1280х720 | Разрешение сменится, сайт открывается правильно, все элементы отображаются корректно, нигде нет смещений |
| 7. Повторить шаг 4 для разрешения экрана 1920х1080 | Разрешение сменится, сайт открывается правильно, все элементы отображаются корректно, нигде нет смещений |

***Тестирование мобильной версии***

*Цель:* проверка адаптивности сайта под различные мобильный устройства.

*Ручное тестирование:* проведено.

*Возможна автоматизация:* нет.

Проверку адаптивности интерфейса сайта под различные мобильные устройства возможно также осуществить с помощью ранее использованного инструмента DevTools. В рамках данного тестирования использовалось несколько рабочих окружений с разными исходными данными:

- iPhone 12/13 + Pro (iOS 14.6 / Safari)

- Nexus 6P (Android 8.0 / Google Chrome)

- Nokia 8110 4G (KaiOS / Mozilla Firefox).

Тест-кейс для проверки адаптивности вёрстки под разные мобильные устройства

|  |  |
| --- | --- |
| Шаги | Ожидаемый результат |
| 1. Открыть тестируемый сайт: <https://stroyparkdiy.ru/> | Сайт загрузился, страница открыта |
| 2. Запустить инструмент DevTools нажатием клавиши F12 | Открылось окно инструмента |
| 3. Выбрать режим адаптивного дизайна сочетанием клавиш Ctrl+Shift+M | Открылся интерфейс адаптивного дизайна |
| 4. Нажать на выбранное устройство и выбрать опцию «Изменить список» | Выбрать позиции из доступных, указанные в окружении и применить |
| 5. Выбрать окружение | Сайт открывается правильно, все элементы отображаются корректно, нигде нет смещений |
| 6. Повторить шаг 5 для других окружений | Сайт открывается правильно, все элементы отображаются корректно, нигде нет смещений |

***Тестирование кроссбраузерности*** и ***кроссплатформенности***

*Цель:* определить, насколько совместимо и адаптивно веб-приложение под различные операционные системы и браузеры.

*Ручное тестирование:* проведено.

*Возможна автоматизация:* да.

Тестирование кроссбраузерности и кроссплатформенности осуществлялось как средствами родной ОС, так и с помощью специального сервиса-инструмента для тестирования сайтов на разных версиях браузеров и систем [10].

Тестовые окружения:

OS: Win 10 и Win 7

Браузеры: IE, Chrome, Firefox

Универсальный тест-кейса для инструмента

|  |  |
| --- | --- |
| Шаги | Ожидаемый результат |
| 1. Открыть инструмент для тестирования кроссбраузерности <https://www.browserling.com/> | Сайт загрузился, страница открыта |
| 2. Выбрать необходимое окружение, в поле для сайта указать https://stroyparkdiy.ru/ | Сайт открывается правильно, все элементы отображаются корректно, нигде нет смещений |
| 3. Повторить шаг 2 для других окружений | Сайт открывается правильно, все элементы отображаются корректно, нигде нет смещений |

Универсальный тест-кейс для родной системы Win 10

|  |  |
| --- | --- |
| Шаги | Ожидаемый результат |
| 1. Открыть интересующий браузер | Браузер запущен |
| 2. В адресной строке указать сайт: https://stroyparkdiy.ru/ | Сайт открывается правильно, все элементы отображаются корректно, нигде нет смещений |
| 3. Повторить шаги 1-2 для других браузеров, согласно окружению | Сайт открывается правильно, все элементы отображаются корректно, нигде нет смещений |

***Нагрузочное тестирование***

*Цель:* определение уровня стабильности и работоспособности сайта при имитации большого количества одновременно подключённых пользователей.

*Ручное тестирование:* да.

*Возможна автоматизация:* проведено.

Нагрузочное тестирование осуществлялось с помощью инструмента Loaddy [11].

Тест-кейс для нагрузочного тестирования

|  |  |
| --- | --- |
| Шаги | Ожидаемый результат |
| 1. Открыть инструмент для нагрузочного тестирования: <https://loaddy.com/> | Инструмент запущен |
| 2. В строку для проверки вставить анализируемый сайт: <https://stroyparkdiy.ru/> и нажать на кнопку «Бесплатная проверка» | Откроется страница с таймером до окончания тестирования страницы |
| 3. Дождаться окончания тестирования | Откроется страница с результатами тестов |

***Тестирование скорости загрузки сайта***

*Цель:* определить, насколько быстро запускается веб-приложение на десктопной и мобильной платформах.

*Ручное тестирование:* да.

*Возможна автоматизация:* проведено.

Тестирование скорости загрузки сайта определялось с помощью автоматизированного инструмента [12].

Тест-кейс для определения скорости загрузки сайта

|  |  |
| --- | --- |
| Шаги | Ожидаемый результат |
| 1. Открыть инструмент для тестирования скорости загрузки: <https://developers.google.com/speed/pagespeed/insights> | Инструмент запущен |
| 2. В строку для проверки страницы указать интересующую: <https://stroyparkdiy.ru/> и нажать кнопку «Анализировать» | Подождать, пока до конца пройдёт проверка |

***Тестирование локализации***

*Цель:* определить, насколько качественно подготовлено содержимое веб-приложений в вопросах орфографии и грамматики.

*Ручное тестирование:* нет.

*Возможна автоматизация:* проведено.

Данное тестирование можно проводить и при выполнении тест-кейсов из других разделов. Данный инструмент идёт как плагин к браузеру, устанавливается стандартными способами и работает, автоматически проверяя каждую загруженную страницу. При выявлении проблем с орфографией и грамматикой выдаст предупреждение. Тестирование локализации осуществлялось с использованием инструмента LanguageTool [13]. Наличие тест-кейса в данном виде тестирования избыточно.

***Тестирование версий для слабовидящих и страдающих цветовой слепотой***

*Цель:* определить, имеется ли версия для людей с ограниченными возможностями.

*Ручное тестирование:* да.

*Возможна автоматизация:* нет.

На современных сайтах необходимо наличие версии для слабовидящих и людей, страдающих цветовой слепотой для расширения возможности посещения сайта большим кругом людей.

Тест-кейс для тестирования версии для слабовидящих

|  |  |
| --- | --- |
| Шаги | Ожидаемый результат |
| 1. Открыть в любом браузере сайт: <https://stroyparkdiy.ru/> | Сайт открыт |
| 2. Найти кнопку или значок для перехода на страницу для слабовидящих людей | Открылась версия для слабовидящих |

***Дружелюбность и интуитивность интерфейса***

*Цель:* определить, насколько сайт адаптивен и понятен, как быстро и свободно пользователь может найти интересующую информацию.

*Ручное тестирование:* да.

*Возможна автоматизация:* нет.

Оценить и протестировать веб-приложение по такому параметру как «Дружелюбность и интуитивность интерфейса» можно в целом субъективно, однако можно и объективно оценить этот параметр по скорости нахождения интересующей информации.

# 3. Результаты тестирования и выводы по результатам испытаний

Согласно подготовленному тест-плану проведено тестирование сайта <https://stroyparkdiy.ru/> по тест-кейсам для функционального и нефункционального видов тестирования. Для заведения багов использовалась баг-трекинговая система Mantis [14].

3.1 Результаты функционального тестирования

***Тестирование каталога***

Выполнено кейсов: 8

Найдено багов: 0

***Тестирование поиска***

Выполнено кейсов: 10

Найдено багов: 0

***Тестирование фильтра***

Выполнено кейсов: 16

Найдено багов: 0

***Тестирование личного кабинета***

Выполнено кейсов: 5

Найдено багов: 0

***Тестирование корзины***

Выполнено кейсов: 2

Найдено багов: 0

***Тестирование доступа к клиентской информации***

Выполнено кейсов: 4

Найдено багов: 0

Подготовлен автоматизированный тест, написанный в Eclipse IDE с подключением Selenium WebDriver, результаты которого представлены на рисунке 3.

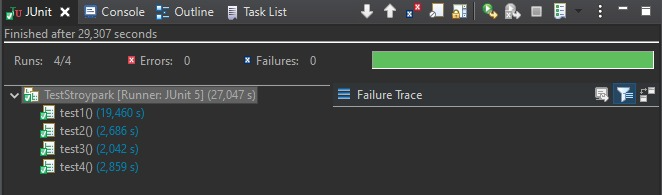


Рисунок3 – Результат автотеста

3.2 Результаты нефункционального тестирования

***Тестирование валидации***

Выполнено кейсов: 1

Найдено багов: 0

Ошибок: 4

Предупреждений: 43

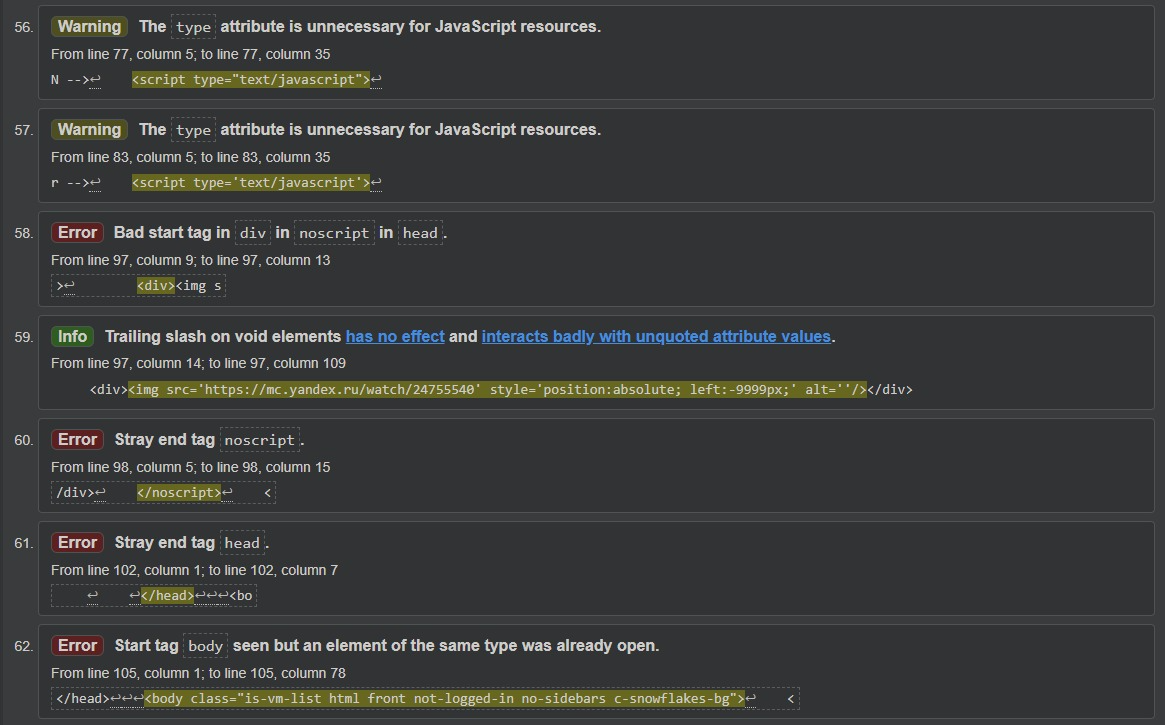


Рисунок 4 – Результаты тестирования стандартов разработки веб-страниц

***Тестирование адаптивности вёрстки и разрешения***

Выполнено кейсов: 1

Найдено багов: 0

***Тестирование мобильной версии***

Выполнено кейсов: 1

Найдено багов: 0

***Тестирование кроссбраузерности и*** ***кроссплатформенности***

Выполнено кейсов: 2

Найдено багов: 0

***Нагрузочное тестирование***

Выполнено кейсов: 1

Найдено багов: 0

Результаты нагрузочного тестирования представлены на рисунке 5. Тест показал, что при нагрузке происходит проседание производительности сайта, снижается отклик.

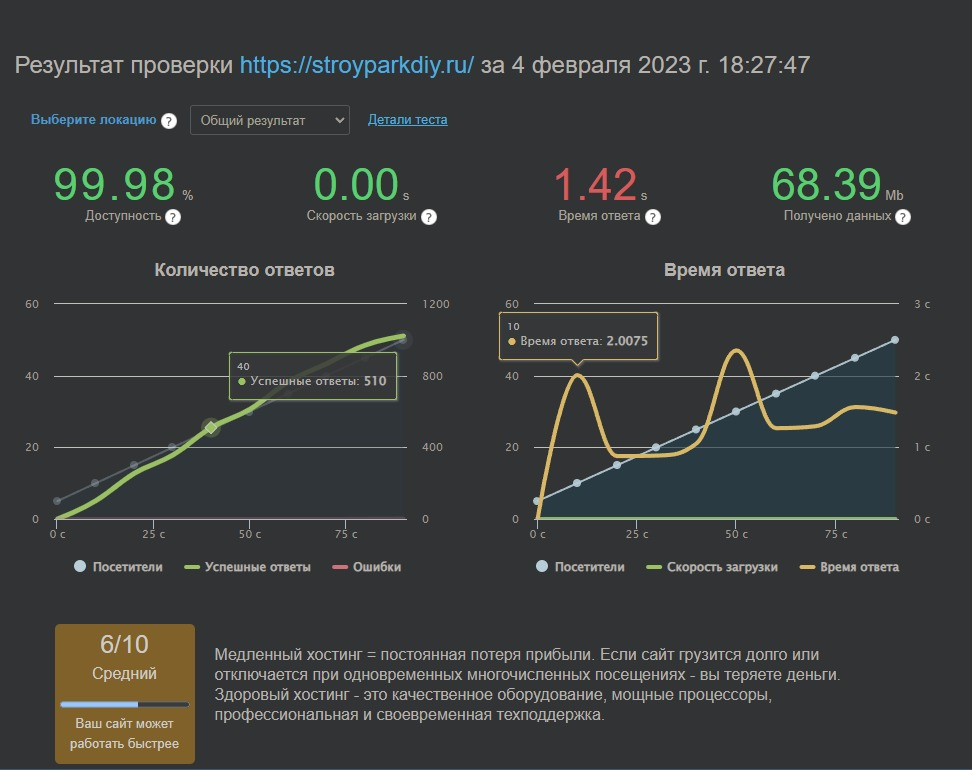


Рисунок5 – Результат нагрузочного тестирования

***Тестирование скорости загрузки сайта***

Выполнено кейсов: 2

Найдено багов: 0

Обнаружена очень низкая скорость и производительность исследуемого сайта как для настольных, так и мобильных версий, отчёт представлен на рисунке 6.

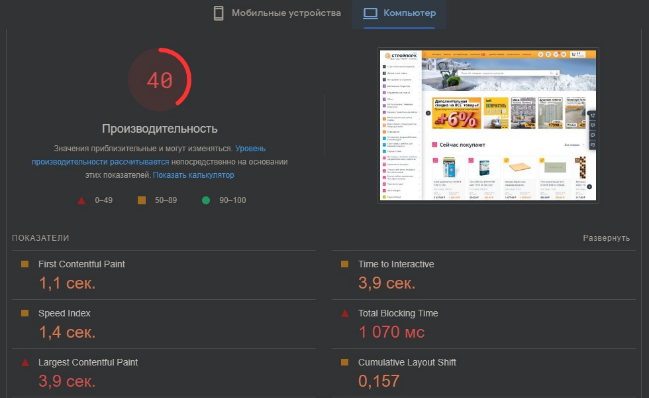
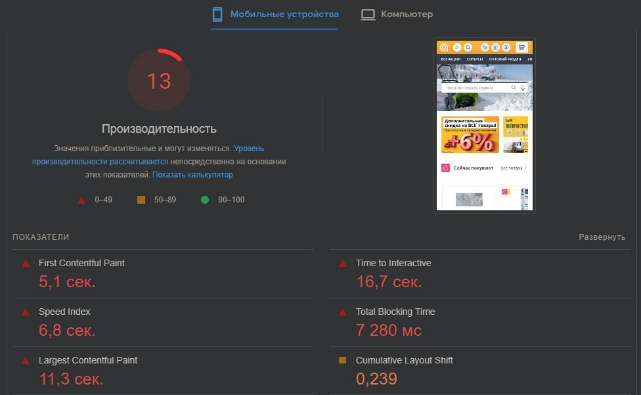


Рисунок6 – Результат скорости загрузки

***Тестирование локализации***

В ходе выполнения других тест-кейсов багов и проблем не обнаружено.

***Тестирование версий для слабовидящих***

Выполнено кейсов: 2

Найдено багов: 2

На сайте <https://stroyparkdiy.ru/> не имеется версии для слабовидящих и людей, обладающих цветовой слепотой.

|  |  |
| --- | --- |
| ID | 2305 |
| Priority | High |
| Severity | Major |

|  |  |
| --- | --- |
| ID | 2306 |
| Priority | High |
| Severity | Major |

***Дружелюбность и интуитивность интерфейса***

В целом сайт интуитивно понятен и удобен, легко и быстро находится нужный товар. Каталог товаров и поисковая панель расположены удобно и сразу «бросаются в глаза».

3.3 Выводы по итогам испытаний

В результате работы удалось провести 55 тестов, 4 из которых были автоматизированы, и найти 2 багов со значительной серьёзностью и высоким приоритетом.

В рамках работы не удалось протестировать функционал сравнения товаров, а также было проведено минимальное количество тестов для каждой категории и всего на двух браузерах – Firefox и Chrome – и одной операционной системе: Windows 10, а также частично Windows 7 и Android. Также в результате работы не получилось соблюсти временные затраты, что отражено в таблице 2. По итогу несколько больше времени ушло на разработку тест-кейсов и подготовку отчёта, что компенсировалось меньшим временем, затраченным на написание автотестов и выполнение тестов.

Таблица 2 – Расчётное и фактически затраченное время на проект

|  |  |  |  |
| --- | --- | --- | --- |
| Номер | Задача | Планируемое время, дни | Фактическое время, дни |
| 1 | Планирование | 1 | 1 |
| 2 | Тест-анализ и тест-дизайн | 2 | 2 |
| 3 | Разработка тест-кейсов | 2 | 3 |
| 4 | Написание автотестов | 3 | 2 |
| 5 | Выполнение тестов | 3 | 2 |
| 6 | Анализ результатов и подведение итогов | 1 | 1 |
| 7 | Подготовка отчёта | 2 | 3 |

# Заключение

Тестирование программного обеспечения – комплексная задача, состоящая из множества подзадач, заключающихся в проверке качества не только функционала, но ещё и ряда других особенностей, таких как адаптивность вёрстки, кроссбраузерность, кроссплатформенность, обладать высокой производительностью и скоростью загрузки, а также поддерживать версии для слабовидящих и людей, страдающих дальтонизмом. Современные приложения должны быть приспособлены и адаптированы под различные браузеры, разрешения экрана, мобильные и настольные устройства, поскольку веб-приложения должны одинаково эффективно запускаться на любых устройствах.

В рамках итогового проекта по программе профессиональной переподготовки: «Тестирование и контроль качества программного обеспечения» удалось провести комплексное тестирование веб-приложения сайт «Стройпарк» – <https://stroyparkdiy.ru/>. В сжатые сроки удалось разработать стратегию тестирования, подготовить тест-кейсы, провести непосредственно тестирование и определить, что даже у таких крупных проектов могут быть недочёты и баги.

# Список используемых источников

1. Инструмент для создания интеллект-карт [Электронный ресурс]. Режим доступа: <https://coggle.it/>

2. Куликов С. Тестирование программного обеспечения. Базовый курс. – 3-е издание, 2021. – 298 с.

3. Инструмент для тестовой комбинаторики, применяемой в технике попарного тестирования [Электронный ресурс]. Режим доступа: <https://pairwise.teremokgames.com/>

4. Среда для разработки Eclipse IDE [Электронный ресурс]. Режим доступа: <https://eclipseide.org/>

5. Инструмент для автоматизированного тестирования Selenium [Электронный ресурс]. Режим доступа: <https://www.selenium.dev/>

6. Инструмент для автоматизированного тестирования Selenium [Электронный ресурс]. Режим доступа: <https://www.selenium.dev/>

7. Irish P., Garsiel T. How browsers work. – 2011 [Электронный ресурс]. Режим доступа: <https://web.dev/howbrowserswork/>

8. Инструмент для оценки соответствия веб-приложений стандартам разработки и применения стилей [Электронный ресурс]. Режим доступа: <https://validator.w3.org/nu/>

9. Инструмент-плагин для разработчиков и тестировщиков DevTools [Электронный ресурс]. Режим доступа: <https://firefox-source-docs.mozilla.org/devtools-user/index.html>

10. Инструмент для кроссбраузерного тестирования Browsering [Электронный ресурс]. Режим доступа: <https://www.browserling.com/>

11. Инструмент для проведения нагрузочного тестирования Loaddy [Электронный ресурс]. Режим доступа: <https://loaddy.com/>

12. Инструмент для тестирования производительности и скорости загрузки сайтов [Электронный ресурс]. Режим доступа: <https://developers.google.com/speed/pagespeed/insights>

13. Инструмент для проверки корректности грамматики и орфографии [Электронный ресурс]. Режим доступа: <https://languagetool.org/>

14. Баг-трекинговая система, применяемая для заведения багов в проекте [Электронный ресурс]. Режим доступа: <http://mantis.fdo.tusur.ru/>

# Приложение А

Результат комбинаторики попарных значений.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № | **Цена** | Мощность | **Источник питания** | **Цвет** | Производитель |
| **1** | менее 500 | 0,5 | USB | белый | APEYRON ELECTRICS |
| **2** | менее 500 | 1 | батарейки | желтый | ARTE LAMP |
| **3** | менее 500 | 3 | сеть | розовый | RISALUX |
| **4** | менее 500 | 0,5 | USB | синий | APEYRON ELECTRICS |
| **5** | 501-1000 | 1 | сеть | синий | APEYRON ELECTRICS |
| **6** | 501-1000 | 3 | USB | белый | ARTE LAMP |
| **7** | 501-1000 | 0,5 | USB | желтый | RISALUX |
| **8** | 501-1000 | 0,5 | батарейки | розовый | APEYRON ELECTRICS |
| **9** | 1001-1500 | 3 | USB | розовый | APEYRON ELECTRICS |
| **10** | 1001-1500 | 0,5 | батарейки | синий | ARTE LAMP |
| **11** | 1001-1500 | 0,5 | сеть | белый | RISALUX |
| **12** | 1001-1500 | 1 | USB | желтый | APEYRON ELECTRICS |
| **13** | 1501 и более | 0,5 | сеть | желтый | APEYRON ELECTRICS |
| **14** | 1501 и более | 0,5 | USB | розовый | ARTE LAMP |
| **15** | 1501 и более | 1 | USB | синий | RISALUX |
| **16** | 1501 и более | 3 | батарейки | белый | APEYRON ELECTRICS |

# приложение Б

Листинг кода с автоматизированными тестами

import org.junit.jupiter.api.AfterAll;

import org.junit.jupiter.api.AfterEach;

import org.junit.jupiter.api.BeforeAll;

import org.junit.jupiter.api.Test;

import org.openqa.selenium.By;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.chrome.ChromeDriver;

import org.openqa.selenium.Dimension;

import org.openqa.selenium.OutputType;

import org.openqa.selenium.WebElement;

import org.openqa.selenium.interactions.Actions;

import org.openqa.selenium.support.ui.ExpectedConditions;

import org.openqa.selenium.support.ui.WebDriverWait;

import static org.junit.jupiter.api.Assertions.\*;

import org.junit.jupiter.api.TestInstance;

import org.junit.jupiter.api.TestInstance.Lifecycle;

public class stroypark {

private WebDriver driver;

@BeforeAll

public void setUp() {

System.setProperty("webdriver.chrome.driver", "D:\\Driver\\chromedriver.exe");

driver = new ChromeDriver();

}

@AfterAll

public void tearDown() {

driver.quit();

}

@Test

public void test1() {

driver.get("https://stroyparkdiy.ru/");

driver.manage().window().setSize(new Dimension(1280, 969));

driver.findElement(By.linkText("Как заказать")).click();

assertEquals(driver.findElement(By.cssSelector("h1")).getText(), ("Как сделать заказ"));

}

@Test

public void test2() {

driver.get("https://stroyparkdiy.ru/");

driver.manage().window().setSize(new Dimension(1280, 969));

driver.findElement(By.linkText("Как оплатить")).click();

assertEquals(driver.findElement(By.cssSelector("h1")).getText(), ("Оплата товара"));

}

@Test

public void test3() {

driver.get("https://stroyparkdiy.ru/");

driver.manage().window().setSize(new Dimension(1280, 969));

driver.findElement(By.linkText("Самовывоз")).click();

assertEquals(driver.findElement(By.cssSelector("h1")).getText(), ("Как сделать заказ на самовывоз из магазина"));

}

@Test

public void test4() {

driver.get("https://stroyparkdiy.ru/");

driver.manage().window().setSize(new Dimension(1280, 969));

js.executeScript("window.scrollTo(0,1007)");

driver.findElement(By.linkText("Возврат товара")).click();

assertEquals(driver.findElement(By.cssSelector("h1")).getText(), ("Возврат/обмен товара"));

}

}