Федеральное государственное образовательное бюджетное

учреждение высшего образования

**«Финансовый университет при Правительстве Российской Федерации»**

**(Финансовый университет)**

**Департамент анализа данных и машинного обучения**

**Отчёт**

по дисциплине «Управление качеством программных систем»

**Лабораторная работа №6**

Выполнила:

студентка учебной группы ПИ21-1 Агапенкова Д.А.

Научный руководитель:

*Клочков Евгений Юрьевич*

  Москва 2024

**Цель работы:**

Автоматизированное тестирование является способом повысить скорость контроля качества при разработке программных систем. В данной лабораторной работе необходимо приобрести навыки создания автотестов для контроля функциональности. Работа основывается на разработанной ранее тестовой документации, а именно тест-кейсах, тест-сьютах, планах тестирования.

**1. Описание выбранного инструмента**

Selenium WebDriver является одним из возможных инструментов для проведения функционального тестирования веб-приложений. Его особенностью является использование установленного в операционной системе браузера для выполнения манипуляций с элементами пользовательского интерфейса веб-приложений. Selenium WebDriver является набором абстракций над веб-страницами, позволяющими искать и выбирать необходимые элементы (по имени тега, классу, xpath, …), взаимодействовать с ними (клик, ввод текста, …), осуществлять навигацию по страницами веб-приложения.

**2. Использование тестов**

Тестируется веб-приложение для стилизации изображений «Ден ь». Оно представляет из себя сервис, в который загружаются два изображения: требуемое стилизовать и стиль. Разработан сценарий тестирования. Его можно увидеть на ресурсе: <https://github.com/DashaAga/development-management/blob/main/%D0%9B%D0%B0%D1%8E%D0%BE%D0%BB%D0%B0%D1%82%D0%BE%D1%80%D0%BD%D0%B0%D1%8F%20%D1%80%D0%B0%D0%B1%D0%BE%D1%82%D0%B0%205.pdf>

В целях оптимизации процессов было принято решение использовать автоматическое тестирование и для него был выбранSelenium WebDriver для Python. Для начала было необходимо установить Selenium WebDriver для Python. Затем инструмент использовался для написания автотестов, которые будут эмулировать поведение пользователя веб-приложения, согласно ручным тестам. WebDriver позволяет искать элементы на странице и взаимодействовать с ними (кликать на кнопки, вводить текст и проверять значения).

Контроль результатов выполненного сценария можно осуществить путём проверки ожидаемых значений после выполнения действий на странице.

**3. Тестирование:**

Код тестируемой программы и тестов находится по ссылке:

https://github.com/den-corp

Тесты были пройдены успешно.

**4. Выводы и ответы на контрольные вопросы:**

Автоматическое тестирование значительно ускоряет работу и позволяет оптимизировать процессы.

1. Каковы цели использования автоматизированного тестирования?

2. В каких случаях можно использовать автоматизированное тестирование?

3. Каковы недостатки автоматизированного тестирования?

4. Что такое локатор?

5. Каковы цели нагрузочного тестирования?

1. Цели использования автоматизированного тестирования включают повышение эффективности тестирования, сокращение времени выполнения тестов, повышение точности и надежности результатов, а также обеспечение более широкого покрытия функциональности приложения.

2. Автоматизированное тестирование можно использовать в случаях, когда необходимо повторять одни и те же тесты многократно, обеспечить быстрое выполнение тестов, улучшить качество тестовых результатов, особенно при длительном или сложном тестировании, а также при необходимости интеграции тестирования в процесс непрерывной интеграции и развертывания (CI/CD).

3. Недостатки автоматизированного тестирования могут включать высокие затраты на начальную настройку и обслуживание тестов, сложность разработки и поддержки автотестов, а также ограничения в области тестирования, такие как невозможность эмуляции человеческого опыта и интуиции.

4. Локатор в автоматизированном тестировании - это механизм, используемый для поиска и идентификации элементов на веб-странице или в приложении. Он может быть основан на различных атрибутах элемента, таких как id, класс, имя, XPath и т. д.

5. Цели нагрузочного тестирования включают оценку производительности системы под нагрузкой, определение максимальной нагрузки, которую система может выдержать, выявление узких мест в архитектуре приложения и оценку поведения системы в условиях реальной или предполагаемой нагрузки.