

МИНОБРНАУКИ РОССИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования

**«САРАТОВСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИМЕНИ Н. Г. ЧЕРНЫШЕВСКОГО»**

Кафедра математической кибернетики и компьютерных наук

**ПОСТРОЕНИЕ ГИБРИДНОГО ТЕСТОВОГО ФРЕЙМВОРКА ДЛЯ
АВТОМАТИЗИРОВАННОГО ВЕБ-ТЕСТИРОВАНИЯ**

АВТОРЕФЕРАТ МАГИСТЕРСКОЙ РАБОТЫ

Студентки 2 курса 273 группы
направления 02.04.03 — Математическое обеспечение и администрирование
информационных систем
факультета КНиИТ
Крючковой Александры Андреевны

Научный руководитель
зав. каф. техн. прогр,

к. ф.-м. н., доцент

И. А. Батраева

Заведующий кафедрой

к. ф.-м. н., доцент

С. В. Миронов

Саратов 2022

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	3
1 Теоретическая часть	6
1.1 Фундаментальные принципы построения тестового фреймворка ..	6
1.2 Архитектура гибридного фреймворка автоматизированного те- стирования	7
1.3 Технологический стек	7
2 Практическая часть	9
2.1 Гибридный фреймворк автоматизированного тестирования	9
2.2 Тестовый сценарий	9
2.3 Генерация отчетности о результатах тестирования	9
ЗАКЛЮЧЕНИЕ	10
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	11

ВВЕДЕНИЕ

Контроль качества программного обеспечения — планомерный процесс, целью которого является гарантировать, что система обладает требуемыми характеристиками. Тестирование программного обеспечения является важной частью процесса контроля качества, т. к. позволяет проверить и оценить соответствие разработанного пакета прикладных программ требованиям к его функциональности.

Цель автоматизации тестирования — автоматическое выполнение тестов вместо использования ручного труда специалистов по тестированию. Автоматизация тестирования является актуальной задачей для компаний в сфере разработки программного обеспечения, т. к. способствует сокращению как времени на проведение тестирования, так и количества необходимых для проведения тестирования специалистов, что, несомненно, способствует повышению экономической эффективности компаний, предоставляющих услуги в сфере информационных технологий.

Нынешние интерактивные веб-приложения стали ключевой платформой для предоставления услуг или продажи товаров. Именно этот тип приложений может работать на различных устройствах ввиду особенностей используемого технологического стека. В настоящее время существует огромное количество различного рода устройств и платформ, в связи с чем тестирование разрабатываемого веб-приложения становится ресурсоемкой задачей, поскольку для каждого типа устройств необходимо использовать соответствующие инструменты и строить требуемое тестовое окружение. Вследствие этой тенденции с каждой итерацией разработки требуется все больше материальных и кадровых ресурсов для обеспечения высокого уровня качества выпускаемого программного продукта. Однако зачастую проекты из-за ограничений используемых ресурсов не имеют возможности динамически масштабировать процесс тестирования. Ввиду чего разрабатываемый продукт подвергается высоким рискам сбоев и возникновения дефектов критического уровня, а электронные коммерческие системы теряют прибыль и клиентов. Однако внедрение автоматизированного тестирования в процесс поставки релизной сборки приложения способствует значительному сокращению вышеуказанных рисков. Так разработка универсальных гибридных тестовых фреймворков, сосредоточенных на поставленной задаче тестирования вне зависимости от используемой платформы

устройства и вида браузера и обеспечивающих автоматизацию этапов жизненного цикла тестирования программного обеспечения, имеет первостепенное значение для решения вышеуказанных проблем. Помимо этого использование универсальных гибридных тестовых фреймворков на проектах по разработке программного продукта обеспечивает высокий уровень автономности функционирования в условиях технологических ограничений, т. к. поддержка и модификация решения осуществляется внутри команды разработки инженерами по автоматизации тестирования или разработчиками посредством технологий с открытым исходным кодом и собственных разработок.

Практическая значимость гибридного тестового фреймворка заключается в том, что он может быть использован для широкого ряда задач веб-тестирования, обеспечивая проведение автоматизированного мобильного и десктопного тестирования без необходимости внедрения сторонних инструментов.

Целью научно-исследовательской работы является разработка универсального кроссплатформенного фреймворка для проведения автоматизированного тестирования программного обеспечения веб-приложений. Решение должно предоставлять функциональные возможности для исполнения автоматизированных тестов, верифицирующих корректность выполнения пользовательских сценариев использования веб-приложений в области электронной коммерции на мобильных и десктопных платформах.

Для выполнения поставленной цели необходимо выполнить следующие задачи:

1. Разработать архитектуру гибридного фреймворка автоматизированного веб-тестирования.
2. В соответствии с разработанной архитектурой реализовать гибридный тестовый фреймворк, удовлетворяющий следующим требованиям:
 - является универсальным — легко адаптируется под тестируемое программное обеспечение целевого бизнес-домена;
 - является гибридным — позволяет проводить автоматизированное тестирование десктопных и мобильных веб-приложений;
 - является кроссплатформенным — поддерживает несколько операционных систем, как мобильных, так и компьютерных;
 - является кроссбраузерным — поддерживает работу с несколькими браузерами;

- является легко масштабируемым;
- реализует методологии тестирования под управлением ключевых слов и данных;
- обеспечивает веб-тестирование;
- обеспечивает параллельный запуск тестов;
- предоставляет локальный и удаленный режим запуска автотестов;
- внедряется в процесс непрерывных интеграции и поставки;
- генерирует отчетность о проведении автотестирования.

В теоретической части работы необходимо описать все используемые технологии и необходимые предварительные сведения для практической реализации.

В практической части работы необходимо описать разработанный программный продукт и привести примеры функционирования разработанной системы.

1 Теоретическая часть

1.1 Фундаментальные принципы построения тестового фреймворка

Автоматизированное тестирование — это набор техник, подходов и инструментальных средств, позволяющий исключить человека из выполнения некоторых задач в процессе тестирования. Царев Ю.В., Сильянова Е. Ф. и Кисельников С. А. в статье [1] сравнивают ручное и автоматизированное тестирование веб-приложений по следующим критериям: ресурсоемкость, производительность и точность. Основываясь на полученных результатах исследования, авторы выделяют следующие преимущества автоматизированных тестов над мануальными: полнота и точность выполнения, высокая производительность, возможность проведения нефункционального тестирования. Таким образом, ключевым понятием автоматизированного тестирования является тестовый фреймворк.

Тестовый фреймворк — это совокупность инструментов и правил, способствующая ускорению разработки автотестов, а также решению поставленных задач функционального и нефункционального тестирования [2]. Первостепенной задачей тестового фреймворка является частичная или полная автоматизация тестовых сценариев функционального тестирования с целью сокращения человеко-часов, затрачиваемых на дымовое или регрессионное тестирования. Алексеева В. Э. и Баран В. И. в [3] описывают ключевые требования к автоматизированным системам тестирования: автоматический запуск тест-кейсов; поддержка концепции непрерывной интеграции; простота поддержки; гибкость; модульность. В публикации [4] к вышеперечисленным требованиям к фреймворку автотестирования добавляют модуль отчетности и модуль логирования. Авторы [5] указывают о необходимости выделения уровня бизнес-логики тестируемого приложения в качестве независимого модуля тестового фреймворка. Безруких А. Д. в [6] обосновывает актуальность проектирования автоматизированного тестирования программного обеспечения, также утверждает, что тестовый фреймворк не должен зависеть от тестируемого приложения, т. е. быть универсальным и гибким для масштабирования. Публикация [7] определяет CEL (Continuous, Evolutionary, Large-scale) принципы разработки фреймворка для автоматизированного мобильного тестирования, основанные на непрерывности, эволюции и масштабируемости.

Таким образом, фреймворк для автоматизированного тестирования дол-

жен:

- выполнять поставленные цели тестирования;
- иметь высокий уровень абстракции;
- быть универсальным;
- поддерживать концепцию непрерывной интеграции;
- быть финансово выгодным.

1.2 Архитектура гибридного фреймворка автоматизированного тестирования

Гибридный фреймворк автоматизированного веб-тестирования представляет собой тестовый фреймворк, позволяющий работать как с десктопной версией веб-приложения, так и его мобильной версией. Данный тип фреймворка сочетает в себе методы управления не только десктопным режимом браузера, но и мобильным, т. е. он располагает методами для широкого спектра электронных устройств, тем самым обеспечивая максимальное тестовое покрытие веб-приложения [8,9].

Абстрактная архитектура гибридного тестового фреймворка для веб-тестирования является модульной [10]: модуль «Тестовый фреймворк», модуль «Автотесты», модуль «Отчет». Уровень «Тестовый фреймворк» состоит из нескольких модулей: «Конфигурация» и «Ядро». Слой «Конфигурация» осуществляет работу с эмуляторами тестовых устройств, обработку исключительных ситуаций [11]. Модуль «Ядро» содержит реализации необходимых функциональных возможностей для работы с различными тестовыми окружениями, а также дополнительные вспомогательные утилиты [12]. Уровень «Автотесты» предоставляет все необходимые возможности для запуска тестов, конфигурации тестов в определенные наборы, создания и редактирования автотестов. Уровень «Отчет» реализует сбор информации о запуске автотестов, а именно ответственен за снятие скриншотов при возникновении ошибок во время прохождения тестов, пошаговую фиксацию выполнения тест-кейсов и составление отчета.

1.3 Технологический стек

Гибридный тестовый фреймворк базируется на семействе технологий Selenium [13]. Технологии Selenium используются на различных этапах разработки программного обеспечения и тестирования для поддержки создания

автоматизированного модульного, интеграционного и сквозного тестирования. В качестве инструментов автоматизации действий с мобильными устройствами был выбран кроссплатформенный инструмент Appium [14]. Данный инструмент можно использовать для автотестирования нативных, гибридных и веб-приложений, а также он ориентирован как на приложения для Android, так и для iOS.

Интеграция автотестов в процесс непрерывных интеграции и доставки осуществляется посредством Selenium, выполняющего задачи контейнеризации тестовых окружений, и сервера автоматизации Jenkins, обеспечивающий управление этапами автоматизации [15]. Основной сущностью, которая определяет задачи каждого этапа автоматизации, является Jenkins Pipeline. Также в процессе автоматизации жизненного цикла автотестов используются облачные технологии. Облачные провайдеры предоставляют виртуальные машины для запуска браузеров и эмуляторов [16]. Google Cloud Platform — облачный провайдер от корпорации Google, который предлагает набор инструментов, состоящий из готовой инфраструктуры, ресурсов и управленческих инструментов.

2 Практическая часть

2.1 Гибридный фреймворк автоматизированного тестирования

Гибридный фреймворк автоматизированного тестирования реализован в виде приложения «APERTURE», которое представляет собой консольную утилиту для автоматизации веб-тестирования сайтов электронной коммерции. Фреймворк поддерживает следующие варианты запуска: регрессионный, конкретный набор тестов и конкретный тест.

Тестовый фреймворк реализует концепцию управления под ключевыми словами. Банк ключевых слов включает в себя такие действия как: открыть браузер, обновить страницу, закрыть текущую вкладку, ввести адрес сайта в адресную строку, закрыть браузер, одинарный клик по элементу, двойной клик по элементу, ожидание видимости элемента, валидация текста элемента, валидация атрибута элемента, наведение курсора на элемент, ведение текста в элемент и другие.

2.2 Тестовый сценарий

Тестовый сценарий оформляется в виде книги с расширением .XLSX. Автотесты представляются как последовательность шагов. Каждый шаг определяет действие с элементом страницы или интерфейсом браузера. Тест-кейс описывается с помощью четырех столбцов: описание шага, веб-элемент, действие с элементом/браузером, значение, которое надо ввести или с которым сравнивают параметры элемента.

2.3 Генерация отчетности о результатах тестирования

Гибридный тестовый фреймворк предоставляет возможность проводить автоматизированное тестирования веб-приложений в рамках нескольких операционных систем и браузеров. Помимо этого фреймворк имеет несколько режимов запуска в зависимости от входных параметров. В связи с предъявляемыми требованиями к модулю отчетности фреймворка, было принято решение о реализации динамической генерации результатов тестирования в виде книги формата .xlxs. Данный формат предоставляет многофункциональный интерфейс для анализа результатов тестирования, т.к. колоночная запись позволяет информативно структурировать статистическую информацию, а также дает возможность использовать специальные представления результатов выполнения тестов в виде различного рода диаграмм и графиков [17].

ЗАКЛЮЧЕНИЕ

В ходе выполнения выпускной квалификационной работы были реализованы все поставленные задачи и цель. Разработанный гибридный тестовый фреймворк для проведения автоматизированного веб-тестирования сайтов электронной коммерции соответствует всем критериям многофункционального тестового фреймворка, поскольку он предоставляет не только функциональные возможности проведения сквозного тестирования веб-приложений, но и предоставляет функции для проведения мобильного автоматизированного тестирования как в режиме локального запуска, так и удаленного, также пригоден для внедрения в процессы непрерывных интеграции и поставки программного продукта.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

- 1 Царев, Ю. В. Особенности ручного и автоматизированного тестирования программного обеспечения / Ю. В. Царев, Е. Ф. Сильянова, С. А. Кисельников // *Вестник науки*. — 2021. — Т. 4, № 6. — С. 117–121.
- 2 Шамсутдинов, А. З. Разработка автоматизированной системы тестирования веб-приложений с применением библиотек Selenium и Allure / А. З. Шамсутдинов, Р. Ф. Гибадуллин // *Молодой исследователь: вызовы и перспективы*. — 2018. — Т. 16, № 69. — С. 422–424.
- 3 Алексеева, В. Э. Разработка системы автоматизации регрессионного тестирования / В. Э. Алексеева, В. И. Баран // *Редакционная коллегия*. — 2021. — С. 17–19.
- 4 Тобин, Д. С. Организация испытаний программного обеспечения для различных моделей его жизненного цикла / Д. С. Тобин, А. В. Богомолов, М.С. Голосовский // *Математические методы в технологиях и технике*. — 2021. — № 7. — С. 132–135.
- 5 Иванников, Д. В. Критерии внедрения автоматизированного тестирования пользовательского интерфейса / Д. В. Иванников, А. Ю. Толчинский, А. А. Солодко // *Студенческая наука: современные реалии*. — 2019. — Т. 7, № 1. — С. 50–52.
- 6 Безруких, А. Д. Автоматизированного тестирования веб-сайта / А. Д. Безруких, М. Д. Черепанов, Д. В. Валбу // *Современные проблемы и тенденции развития экономики и управления бизнес-процессами*. — 2021. — С. 122–126.
- 7 Linares-Vasquez, M. Continuous, Evolutionary and Large-Scale: A New Perspective for Automated Mobile App Testing / M. Linares-Vasquez, K. Moran, D. Poshyvanyk // *International Conference on Software Maintenance and Evolution*. — 2017. — Pp. 399–410.
- 8 Sun, Z. A web testing platform based on hybrid automated testing framework // *Proceedings of 2019 IEEE 4th Advanced Information Technology, Electronic and Automation Control Conference*. — Institute of Electrical and Electronics Engineers, Inc, 2019. — Pp. 689–692.

- 9 Абушахманова, А. А. Анализ инструментов автоматизации тестирования web-приложений / А. А. Абушахманова // *Актуальные вопросы развития современной цифровой среды*. — 2021. — С. 31–35.
- 10 Батраева, И. А. Принципы построения фреймворка для автоматизированного тестирования веб-приложений / И. А. Батраева, А. А. Крючкова // *Компьютерные науки и информационные технологии*. — 2021. — С. 17–20.
- 11 Брагин, К. А. Разработка оптимальной методики тестирования программного обеспечения / К. А. Брагин, В. О. Кушев // *XXIII Всероссийская студенческая научно-практическая конференция Нижневартковского государственного университета*. — 2021. — С. 86–90.
- 12 Горшков, А. Д. Анализ систем автоматизированного тестирования / А. Д. Горшков // *Перспективы развития науки и образования*. — 2021. — С. 84–92.
- 13 Лимонцев, Д. С. Сравнительный анализ автоматизированных средств тестирования программного обеспечения / Д. С. Лимонцев // *Приоритетные направления инновационной деятельности в промышленности*. — 2020. — С. 54–56.
- 14 Пожидаев, Ю. К. Автоматизация тестирования мобильных приложений с помощью Appium / Ю. К. Пожидаев, М. А. Миткевич // *Интернаука*. — 2020. — № 19(148). — С. 47–49.
- 15 Бабаев, В. С. Исследование современных ci/cd инструментов и внедрение их в крупные web-проекты на примере jenkins / В. С. Бабаев // *Научное сообщество студентов XXI столетия. Технические науки*. — 2019. — № 32. — С. 147–150.
- 16 Андреева, А. Ю. Облачная инфраструктура для современных веб-приложений. saas, paas, iaas платформы: Amazon web services, microsoft azure, google cloud / А. Ю. Андреева, П. А. Розанов // *Развитие современной науки и образования: актуальные вопросы, достижения и инновации*. — 2019. — № 32. — С. 15–17.
- 17 Семченко, Р. В. Работа с Excel на Java с помощью Apache POI / Р. В. Семченко, П. А. Еровлев // *Постулат*. — 2021. — № 1. — С. 9–11.