**Министерство науки и высшего образования Российской Федерации**

**ФГАОУ ВО «УрФУ имени первого Президента России Б.Н. Ельцина»**

Кафедра «школа бакалавриата (школа)»

Оценка работы \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Руководитель от УрФУ Спиричева Н.Р.

Тема задания на практику

**Разработка и интеграция автоматизированной системы тестирования для веб-приложения**

ОТЧЕТ

Вид практики Производственная практика

Тип практики Производственная практика, преддипломная

Руководитель практики от предприятия (организации) Соловьева Н.В. И

**ФИО руководителя Подпись**

Студент Коновалов Т.А. \_\_\_\_\_\_

**ФИО студента Подпись**

Специальность (направление подготовки) 09.03.01 Информатика и вычислительная техника

Группа РИ-410910

Екатеринбург 2025

# ВВЕДЕНИЕ

В современном мире цифровые технологии развиваются стремительными темпами, а веб-приложения становятся все более сложными и многофункциональными. Они включают в себя множество компонентов: динамические интерфейсы, распределенные серверные системы, интеграции со сторонними API, сложную бизнес-логику и высокие требования к безопасности и производительности. В таких условиях традиционные методы ручного тестирования оказываются недостаточно эффективными: они требуют значительных временных и трудовых затрат, а также не всегда способны обеспечить достаточное покрытие тестами.

Рост сложности веб-приложений приводит к увеличению числа потенциальных уязвимостей и ошибок, которые могут негативно сказаться на пользовательском опыте, безопасности и стабильности системы. Особенно критично это для проектов с частыми обновлениями, где изменения в одном модуле могут неожиданно повлиять на работу других компонентов. В таких условиях автоматизация тестирования становится не просто полезным инструментом, а необходимостью, позволяющей: сократить время проверки функциональности, повысить точность обнаружения ошибок, минимизировать человеческий фактор и снизить затраты на поддержку качества продукта.

Цель работы **-** разработать и интегрировать автоматизированную систему тестирования для веб-приложения, способной эффективно проверять его функциональность на разных уровнях.

Для достижения данной цели планируется выполнение следующих шагов:

1. Изучить существующие решения и подходы для автоматизации тестирования

2. Спроектировать сценарии для тестирования пользовательского графического интерфейса.

3. Разработать программу для проверки работоспособности приложения с формированием отчета

4. Настроить и развернуть систему автоматизированного тестирования

5. Применить систему автоматизированного тестирования для тестирования веб-приложения на практике

Объект исследования – процесс тестирования веб-приложений.

Предметом исследования являются методы и инструменты для автоматизированного тестирования.

В рамках данного отчёта будут рассмотрены основные моменты разработки данной системы, проведен анализ существующих решений, рассмотрены основные подходы и методы для автоматизации тестирования, а также определена архитектура системы и разработана большая часть её компонентов.

1. Описание работы.
   1. Используемые технологии

Для выполнения поставленных задач были использованы следующие средства разработки:

* язык программирования Python, так как он позволяет легко автоматизировать необходимые процессы;
* для автоматизации действий браузера и верхнеуровневого тестирования фреймворк Selenium, засчёт простой интеграции с Python;
* для формирования отчёта в формате docx была использована библиотека Python-docx;
* для написания модульных тестов был использован фреймворк BOOST TEST;
* для автоматического запуска запуска тестов была использована система Gitea Actions.

1.2. Календарный план

В таблице 1 представлен календарный план прохождения практики.

Таблица 1 – Календарный план

|  |  |  |  |
| --- | --- | --- | --- |
| ***Этапы практики*** | ***Наименование  работ студента*** | ***Срок*** | ***Примечание*** |
| организационный | Усвоение правил внутреннего трудового распорядка Профильной организации; правил по охране труда и технике безопасности, санитарно-эпидемиологических правил, режима конфиденциальности, принятого в Профильной организации. Обсуждение с руководителем практики от Профильной организации темы индивидуального задания на практическую подготовку в организации, уточнение ее формулировки, составление конкретного последовательного перечня работ, необходимых для выполнения задания. | 10.02.2025– 21.02.2025 | выполнено |

|  |  |  |  |
| --- | --- | --- | --- |
| основной | Выполнение индивидуального задания по следующему перечню работ:   1. Изучение аналогов, а также подходов для автоматизированного тестирования. 2. Написание основных сценариев для тестирования. 3. Написание верхнеуровневых и модульных тестов. 4. Разработка системы для автоматического запуска тестов и формирования отчёта. | 21.02.2025 - 30.03.2025 | выполнено |
| заключительный | Составление структуры и написание полного отчета о выполнении индивидуального задания на практическую подготовку в Профильной организации, в соответствии с требованиями шаблонов УрФУ. Согласование отчета с руководителем практики. Получение Отзыва на прохождение практической подготовки и документа к Отзыву от Профильной организации. Нормоконтроль отчета о практике. Подготовка презентации к защите практики. | 31.03.2025- 06.04.2025 | выполнено |

1.3 Анализ решений для тестирования

В настоящее время существует множество инструментов для автоматизированного тестирования веб-приложений. Рассмотрим наиболее популярные из них, их ключевые особенности, преимущества и недостатки.

1. **Selenium WebDriver**

Selenium WebDriver является одним из самых распространенных инструментов для автоматизированного тестирования веб-приложений. Разработанный в 2004 году, он поддерживает множество языков программирования, включая Java, Python, C# и JavaScript.

**Основные преимущества:**

* Кроссплатформенность - поддерживает все основные браузеры (Chrome, Firefox, Safari, Edge)
* Гибкость интеграции с различными фреймворками (TestNG, JUnit, pytest)
* Возможность масштабирования для сложных проектов
* Большое сообщество пользователей и обширная документация

**Недостатки:**

* Тесты могут быть хрупкими при изменениях в интерфейсе
* Отсутствие встроенных инструментов для отладки

1. **Cypress**

Cypress - современный фреймворк для end-to-end тестирования, появившийся в 2017 году. Работает на JavaScript/TypeScript.

**Основные преимущества:**

* Простота настройки и использования
* Встроенные инструменты отладки
* Автоматические ожидания элементов
* Хорошая интеграция с современными фронтенд-фреймворками

**Недостатки:**

* Ограниченная поддержка браузеров (только Chromium-движки)
* Не поддерживает работу с несколькими вкладками/доменами
* Меньшая гибкость по сравнению с Selenium

**3. Playwright**

Playwright - относительно новый фреймворк (2020 год), разработанный Microsoft. Поддерживает несколько языков программирования.

**Основные преимущества:**

* Полноценная кросс-браузерная поддержка
* Стабильность тестов благодаря встроенным механизмам retry
* Возможность эмуляции мобильных устройств
* Поддержка параллельного выполнения тестов

**Недостатки:**

* Меньшее сообщество по сравнению с Selenium
* Не поддерживает устаревшие версии браузеров

Таблица 2 – Сравнительная таблица

| **Фреймворк** | **Языки** | **Браузеры** | **Преимущества** | **Недостатки** |
| --- | --- | --- | --- | --- |
| **Selenium** | Javа, Python, C#, JS | Все | Гибкость, активное сообщество | Сложность, хрупкость |
| **Cypress** | JS/TS | Chrome, Edge | Простота использования, отладка | Не поддерживает Firefox/Safari |
| **Playwright** | JS, Python, Java, C# | Chrome, Firefox, WebKit | Стабильность, открытый API | Молодой проект |

Выбор конкретного инструмента должен основываться на требованиях проекта, технологическом стеке команды и масштабах тестирования. Для сложных enterprise-решений по-прежнему оптимальным выбором остается Selenium, в то время как для более специализированных задач могут быть предпочтительнее Cypress или Playwright.