1. **Цель работы**

Разработка автотестов для проверки функциональности программной системы на основе тест-кейсов.

1. **Описание реализованных автотестов**

Для реализации автотестов использованы инструменты, эмулирующие действия пользователя с элементами пользовательского интерфейса. В частности, применён Selenium WebDriver для автоматизации веб-приложений. Были разработаны модульные тесты, каждый из которых проверяет определенный функциональный блок системы.

1. **Пример кода автотестов:**

import unittest

from selenium import webdriver

from selenium.webdriver.common.keys import Keys

class TestRegistrationForm(unittest.TestCase):

def setUp(self):

self.driver = webdriver.Chrome()

self.driver.get("https://smstretching.ru/")

def test\_correct\_input(self):

# Ввод корректных значений в форму

self.\_fill\_form("Алексей", "Рязанцев")

saved\_name = self.driver.find\_element\_by\_id("saved-name").text

self.assertEqual(saved\_name, "Алексей")

def test\_invalid\_name\_input(self):

# Ввод некорректного значения в поле "Имя"

self.\_fill\_form("!@#$%", "Рязанцев")

error\_message = self.driver.find\_element\_by\_id("name-error").text

self.assertEqual(error\_message, "Некорректное имя")

def test\_invalid\_surname\_input(self):

# Ввод некорректного значения в поле "Фамилия"

self.\_fill\_form("Алексей", "!@#$%")

error\_message = self.driver.find\_element\_by\_id("surname-error").text

self.assertEqual(error\_message, "Некорректная фамилия")

def \_fill\_form(self, name, surname):

name\_input = self.driver.find\_element\_by\_id("name")

name\_input.clear()

name\_input.send\_keys(name)

surname\_input = self.driver.find\_element\_by\_id("surname")

surname\_input.clear()

surname\_input.send\_keys(surname)

save\_button = self.driver.find\_element\_by\_id("save-button")

save\_button.click()

def tearDown(self):

self.driver.quit()

if \_\_name\_\_ == "\_\_main\_\_":

unittest.main()

1. **Отчет о тестировании**

* **Тест-кейс 1**: Ввод корректных значений

**Результат**: Данные успешно сохраняются и остаются после обновления страницы. Изменение данных возможно

**Состояние**: Пройден

* **Тест-кейс 2**: Ввод некорректного значения в поле "Имя"

**Результат**: Появляется сообщение об ошибке "Некорректное имя".

Данные не сохраняются.

**Состояние**: Пройден

* **Тест-кейс 3:** Ввод некорректного значения в поле "Фамилия"

**Результат**: Появляется сообщение об ошибке "Некорректная фамилия". Данные не сохраняются.

**Состояние**: Пройден

* **Тест-кейс 4:** Ввод некорректного значения в поле "Отчество"

**Результат**: Некорректные символы сохраняются и отображаются в профиле.

**Состояние**: Не пройден

* **Тест-кейс 5:** Ввод некорректного значения в поле "Дата рождения"

**Результат**: Некорректные значения сохраняются без указания на ошибку

**Состояние**: Не пройден

* **Тест-кейс 6:** Ввод некорректного значения в поле "Телефон"

**Результат**: Номер телефона сохраняется без возможности изменения.

**Состояние**: Не пройден

* **Тест-кейс 7:** Ввод некорректного значения в поле "Email"

**Результат**: При вводе некорректных данных появляется сообщение об ошибке, но данные сохраняются.

**Состояние**: Не пройден

1. **Выводы по работе**

В ходе выполнения работы были разработаны автотесты для проверки функциональности программной системы на основе предоставленной тестовой документации. Большинство тест-кейсов не прошли из-за некорректного сохранения данных при вводе некорректных значений. Это свидетельствует о необходимости доработки системы для обеспечения корректной обработки некорректных данных пользовательского ввода.