

Санкт-Петербургский Национальный Исследовательский
Университет Информационных Технологий, Механики и Оптики

Факультет «Инфокоммуникационных Технологий»
Направление подготовки «Программирование в
инфокоммуникационных системах»

Лабораторная работа №2

Выполнил:

Крылов Дан Станиславович

Группа №3322

Проверил:

Кочубеев Николай Сергеевич

Санкт-Петербург
2024

Цель работы.

протестировать работу открытого кода из GitHub с помощью интеграционных-тестов.

Задачи.

1. Выбор репозитория с GitHub
2. Анализ тестируемых функциональностей
3. Написание интеграционных тестов

Ход работы.

1. Выбор репозитория.

Был выбран репозиторий - <https://github.com/jecfish/coffee-cart>

Он представляет собой сайт, на котором происходит выбор кофе, добавление его в корзину, подсчет общей стоимости напитков.

2. Анализ тестируемых функциональностей

Особенности системы:

1. Дважды щелкните по названию кофе, чтобы перевести его на китайский язык.
2. Щелкните правой кнопкой мыши по значку кофе, чтобы открыть добавить в корзину.
3. На кнопку «Нажмите курсор на «Оплатить»» отобразится быстрый предварительный просмотр корзины, нажмите, чтобы добавить или удалить товары.
4. Случайное всплывающее окно с промо-кофе появляется при добавлении каждых 3-х предметов в корзину.

Важные случаи использования (Use Cases):

- Добавление товара
- Смена языка (для названия напитка)
- Акционный товар (каждый 4 напиток по акции)
- Введение имени и почты для оплаты заказа
- Удаление элемента из общего заказа

3. Написание тестов.

На рисунках 1-3 приведены тесты. Первый тест добавляет в корзину напиток и проверяет его наличие. Второй тест применяет смену языка названия напитка (английский/китайский) при нажатии. В третьем тесте проверяется присутствие 4 акционного напитка. Четвертый тест рассматривает отправку почты и имени пользователя. А в пятом тесте используется удаление элемента корзины. Все тесты успешно пройдены.

```
1 import pytest
2 from selenium import webdriver
3 from selenium.webdriver.common.by import By
4 from selenium.webdriver.support.ui import WebDriverWait
5 from selenium.webdriver.support import expected_conditions as EC
6
7 @pytest.fixture(scope="module")
8 def browser():
9     options = webdriver.ChromeOptions()
10    options.add_argument("--start-maximized")
11    driver = webdriver.Chrome(options=options)
12    yield driver
13    driver.quit()
14
15 def test_add_to_cart(browser):
16     # Открываем главную страницу
17     browser.get("https://coffee-cart.app/")
18     # Находим элемент кофе и добавляем его в корзину
19     coffee_element = browser.find_element(By.CSS_SELECTOR, "div.cup")
20     coffee_element.click()
21     # Проверяем, что товар был успешно добавлен в корзину
22     cart_icon = browser.find_element(By.CSS_SELECTOR, '[aria-label="Cart page"]')
23     cart_icon.click()
24     # Ждем появления элементов корзины
25     WebDriverWait(browser, 10).until(EC.presence_of_all_elements_located((By.CLASS_NAME, "list-item")))
26     cart_items = browser.find_elements(By.CLASS_NAME, "list-item")
27     assert len(cart_items) == 2
```

Рисунок 1 – тест 1.

```

def test_language_switch(browser):
    browser.get("https://coffee-cart.app/")

    wait = WebDriverWait(browser, 10)
    coffee_element = wait.until(EC.element_to_be_clickable((By.XPATH, '//h4[contains(text(), "Espresso")]')))

    ActionChains(browser).double_click(coffee_element).perform()

    assert coffee_element.text == "特浓咖啡\n$10.00"

def test_popup_offer_appears_every_third_item(browser):

    browser.get("https://coffee-cart.app/")

    coffee_elements = browser.find_elements(By.CSS_SELECTOR, "div.cup")

    for i in range(3):
        coffee_elements[i].click()

    popup = WebDriverWait(browser, 10).until(
        EC.visibility_of_element_located((By.CSS_SELECTOR, ".promo"))
    )

    assert popup.is_displayed() is True

```

Рисунок 2 – Тесты 2-3

```

def test_checkout_process(browser):

    browser.get("https://coffee-cart.app/")

    coffee_element = browser.find_element(By.CSS_SELECTOR, "div.cup")
    coffee_element.click()

    checkout_button = browser.find_element(By.CSS_SELECTOR, ".pay")
    checkout_button.click()

    name_field = browser.find_element(By.ID, "name")
    email_field = browser.find_element(By.ID, "email")

    name_field.send_keys("John Doe")
    email_field.send_keys("johndoe@example.com")

    submit_button = browser.find_element(By.XPATH, "//button[@type='submit']")
    submit_button.click()

def test_total_price_update_on_quantity_change(browser):

    browser.get("https://coffee-cart.app/")

    coffee_element = browser.find_element(By.CSS_SELECTOR, "div.cup")
    coffee_element.click()
    coffee_element.click()

    cart_icon = browser.find_element(By.CSS_SELECTOR, '[aria-label="Cart page"]')
    cart_icon.click()

    quantity_input = browser.find_element(By.CSS_SELECTOR, '[aria-label="Remove one Espresso"]')
    quantity_input.click()

    total_price = browser.find_element(By.CSS_SELECTOR, ".pay")
    total_price_value = float(total_price.text.split(' ')[1])
    expected_total_price = '$10.00'
    assert total_price_value == expected_total_price

```

Рисунок 3 – Тесты 4-5

Вывод:

Достигнута цель работы - протестирована работа кода из GitHub с помощью интеграционных-тестов.