МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования

НИЖЕГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ

УНИВЕРСИТЕТ им. Р.Е.АЛЕКСЕЕВА

Институт радиоэлектроники и информационных технологий

Кафедра «Вычислительные системы и технологии»

ОТЧЁТ

по лабораторной работе №2

по дисциплине «Тестирование программного обеспечения»

РУКОВОДИТЕЛЬ:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Скорынин С.С.

СТУДЕНТ:

Китов А.А.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

21-ПО

Работа защищена «\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_

С оценкой \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Нижний Новгород

2025

**Цель работы:** изучить основы разработки модульных тестов, получить навыки работы со средствами тестирования.

**Задание:**

Выбрать web-сайт для тестирования. Согласовать задание с преподавателем. Составить план функционального тестирования для выбранного сайта.

• Разработать набор unit-тестов для тестирования web- сайта с использованием Selenium Web-driver (не менее 10).

План тестирования должен включать в себя как минимум:

1. Проверка заголовка страницы.

2. Проверка видимости объектов

3. Переход по ссылке.

4. Заполнение текстового поля.

5. Эмуляция нажатия на кнопку.

• Реализовать набор unit-тестов на языке С#, поддерживаемом платформой .NET. Использование других языков необходимо заранее согласовать с преподавателем.

• Продемонстрировать: Выполнение unit- тестов с использованием Selenium Web-driver.

• Сделать отчет по лабораторной работе.

**Программа:**

import unittest

from selenium import webdriver

from selenium.webdriver.edge.options import Options

from selenium.webdriver.common.keys import Keys

from selenium.webdriver.common.by import By

from selenium.webdriver.support.ui import WebDriverWait

from selenium.webdriver.support import expected\_conditions as EC

import logging

from selenium.webdriver.remote.remote\_connection import LOGGER

LOGGER.setLevel(logging.WARNING)

class TestSSTUWebsite(unittest.TestCase):

    def setUp(self):

        options = Options()

        options.use\_chromium = True

        options.add\_argument("--headless")

        options.add\_argument('--log-level=3')

        self.driver = webdriver.Edge(options=options)

        self.addCleanup(self.driver.quit)

#

    def test\_page\_title(self):

        self.driver.get('https://www.sstu.ru/')

        self.assertIn('Саратовский государственный технический университет имени Гагарина Ю.А.', self.driver.title)

#

    def test\_clicked\_links(self):

        self.driver.get('https://www.sstu.ru/')

        wait = WebDriverWait(self.driver, 10)

        link = wait.until(EC.element\_to\_be\_clickable((By.CSS\_SELECTOR, 'a[href\*="/abiturientu/"]')))

        link.click()

        self.assertIn('/abiturientu/', self.driver.current\_url)

#

    def test\_visibility(self):

        self.driver.get('https://www.sstu.ru/')

        element = self.driver.find\_element(By.CLASS\_NAME, 'logo')

        self.assertTrue(element.is\_displayed())

#

    def test\_social\_media\_link(self):

        self.driver.get('https://www.sstu.ru/')

        vk\_link\_button = self.driver.find\_element(By.CSS\_SELECTOR, 'a.vk')

        vk\_link\_button.click()

        WebDriverWait(self.driver, 10).until(EC.number\_of\_windows\_to\_be(2))

        self.driver.switch\_to.window(self.driver.window\_handles[1])

        self.assertIn('vk.com/sstusaratov', self.driver.current\_url.lower())

#

    def test\_search\_query(self):

        self.driver.get('https://www.sstu.ru/')

        wait = WebDriverWait(self.driver, 10)

        search\_button = wait.until(EC.element\_to\_be\_clickable((By.ID, 'header-search')))

        search\_button.click()

        search\_input = wait.until(EC.visibility\_of\_element\_located((By.ID, 'inline-search\_\_input')))

        search\_input.send\_keys('Институт прикладных информационных технологий и коммуникаций' + Keys.RETURN)

        wait.until(EC.presence\_of\_element\_located((By.CLASS\_NAME, 'search-page')))

        self.assertIn('search', self.driver.current\_url)

#

    def test\_menu\_elements(self):

        self.driver.get('https://www.sstu.ru/')

        wait = WebDriverWait(self.driver, 10)

        menu\_elements = wait.until(EC.presence\_of\_all\_elements\_located((By.CSS\_SELECTOR, 'ul.menu > li')))

        self.assertGreater(len(menu\_elements), 0)

#

    def test\_footer\_links(self):

        self.driver.get('https://www.sstu.ru/')

        footer\_links = self.driver.find\_elements(By.CSS\_SELECTOR, 'footer')

        for link in footer\_links:

            self.assertTrue(link.is\_displayed())

#

    def test\_institute\_page(self):

        self.driver.get('https://www.sstu.ru/obrazovanie/instituty/inpit/')

        title\_element = WebDriverWait(self.driver, 10).until(

            EC.presence\_of\_element\_located((By.CLASS\_NAME, 'slide-text'))

        )

        self.assertTrue(title\_element.is\_displayed())

        self.assertIn('ИНСТИТУТ ПРИКЛАДНЫХ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ И КОММУНИКАЦИЙ', title\_element.text)

        contacts\_element = WebDriverWait(self.driver, 10).until(

            EC.presence\_of\_element\_located((By.CLASS\_NAME, 'social-list'))

        )

        self.assertTrue(contacts\_element.is\_displayed())

#

    def test\_admission\_page(self):

        self.driver.get('https://www.sstu.ru/abiturientu/v-o/2025/')

        admission\_title = WebDriverWait(self.driver, 10).until(

            EC.presence\_of\_element\_located((By.TAG\_NAME, 'h1'))

        )

        self.assertIn('Прием 2025 года', admission\_title.text)

        admission\_links = self.driver.find\_elements(By.CSS\_SELECTOR, '.external-link')

        self.assertGreater(len(admission\_links), 0)

#

    def test\_news\_section(self):

        self.driver.get('https://www.sstu.ru/')

        main\_news\_link = WebDriverWait(self.driver, 10).until(

            EC.element\_to\_be\_clickable((By.CSS\_SELECTOR, 'div.title-add a[href\*="/main-news/"]'))

        )

        main\_news\_link.click()

        self.driver.switch\_to.window(self.driver.window\_handles[-1])

        news\_items = WebDriverWait(self.driver, 10).until(

            EC.presence\_of\_all\_elements\_located((By.CSS\_SELECTOR, '.news-item'))

        )

        self.assertGreater(len(news\_items), 0, msg='Не найдено новостных элементов.')

        first\_news\_date = WebDriverWait(self.driver, 10).until(

            EC.presence\_of\_element\_located((By.CSS\_SELECTOR, '.news-item\_\_date'))

        )

        self.assertTrue(first\_news\_date.is\_displayed(), msg='Дата первой новости не отображается.')

if \_\_name\_\_ == '\_\_main\_\_':

    unittest.main()

**Проверка работы программы:**

