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 ПО»

РУКОВОДИТЕЛЬ:

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

СТУДЕНТ:

  Смирнова С.В.

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

        21-ПО

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

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

Нижний Новгород

2025

**Задание:**

1. Выбрать web-site (например, sibsutis.ru).

2. Составить план функционального тестирования для выбранного сайта.

3. Реализовать набор тестов при помощи NUnit и Selenium WebDriver.

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.

**Ход работы:**

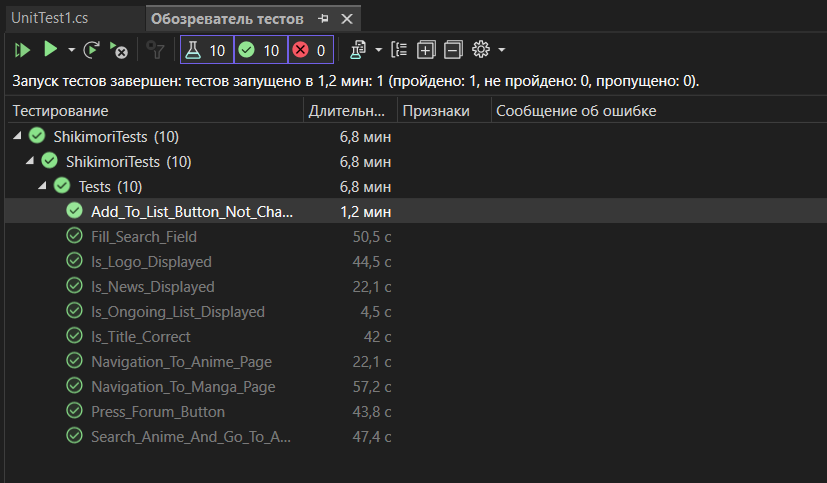
Для тестирования был выбран сайт «shikimori».

Это сайт для оценки аниме и добавления их в различные списки.

План тестирования:

1. Проверка заголовка страницы
2. Проверка отображается ли логотип
3. Отображается ли объект «Новости»
4. Отображается ли блок с аниме выходящими в данный момент (онгоингами)
5. Проверка перехода по ссылке к url «/animes»
6. Проверка перехода по ссылке к url «/mangas»
7. Заполнение текстового поля «Поиск»
8. Эмуляция нажатия на кнопку «Форум»
9. Поиск конкретного аниме и переход по ссылке на его страничку
10. Проверка, что если пользователь не авторизован кнопка «Добавить в список» не изменяет своего значения

Обозреватель тестов:

****

**Код программы:**

using NUnit.Framework;

using OpenQA.Selenium;

using OpenQA.Selenium.Chrome;

using OpenQA.Selenium.Support.UI;

using System;

using WebDriverManager;

using WebDriverManager.DriverConfigs.Impl;

new DriverManager().SetUpDriver(new ChromeConfig()); //Скачивает драйвер

namespace ShikimoriTests

{

public class Tests

{

private IWebDriver driver;

private WebDriverWait wait;

[SetUp]

public void Setup()

{

driver = new ChromeDriver();

wait = new WebDriverWait(driver, TimeSpan.FromSeconds(10));

driver.Manage().Window.Maximize();

driver.Navigate().GoToUrl("https://shikimori.one");

}

[TearDown]

public void TearDown()

{

driver.Quit();

}

//Проверка заголовка страницы

[Test]

public void Is\_Title\_Correct()

{

string expected = "Шикимори - энциклопедия аниме и манги";

Assert.AreEqual(expected, driver.Title);

}

//Проверка видимости объектов

[Test]

public void Is\_Logo\_Displayed()

{

var logo = driver.FindElement(By.CssSelector("#dashboards\_show > header > div.menu-logo > a > div.logo"));

Assert.IsTrue(logo.Displayed);

}

[Test]

public void Is\_News\_Displayed()

{

var news = driver.FindElement(By.XPath("//\*[@id=\"dashboards\_show\"]/section/div[2]/div/div[4]/div[2]"));

Assert.IsTrue(news.Displayed);

}

[Test]

public void Is\_Ongoing\_List\_Displayed()

{

var ongoingList = driver.FindElement(By.CssSelector("#dashboards\_show > section > div > div > div:nth-child(1) > div.subheadline.linkheadline > a"));

Assert.IsTrue(ongoingList.Displayed);

}

//Переход по ссылке

[Test]

public void Navigation\_To\_Anime\_Page()

{

var menuToggle = wait.Until(d =>

d.FindElement(By.CssSelector("#dashboards\_show > header > div.menu-logo > div > span.submenu-triangle.icon-home")));

menuToggle.Click();

var animeLink = driver.FindElement(By.CssSelector("a.icon-anime[title='Аниме']"));

((IJavaScriptExecutor)driver).ExecuteScript("arguments[0].click();", animeLink);

wait.Until(d => d.Url.Contains("/animes"));

Assert.That(driver.Url, Does.Contain("/animes"));

}

[Test]

public void Navigation\_To\_Manga\_Page()

{

var mangaLink = wait.Until(d =>d.FindElement(By.XPath("//\*[@id=\"dashboards\_show\"]/section/div[2]/div/div[3]/div/div[2]/div[1]/div[2]/div")));

mangaLink.Click();

wait.Until(d => d.Url.Contains("/mangas"));

Assert.That(driver.Url, Does.Contain("/mangas"));

}

//Заполнение текстового поля

[Test]

public void Fill\_Search\_Field()

{

var search = wait.Until(d => d.FindElement(By.CssSelector("#dashboards\_show > header > div.menu-icon.search.mobile")));

((IJavaScriptExecutor)driver).ExecuteScript("arguments[0].click();", search);

string searchText = "Наруто";

var searchInput = wait.Until(d=>d.FindElement(By.XPath("//\*[@id=\"dashboards\_show\"]/header/div[3]/label/input")));

searchInput.SendKeys(searchText);

Assert.AreEqual(searchText, searchInput.GetAttribute("value"));

}

//Эмуляция нажатия на кнопку

[Test]

public void Press\_Forum\_Button()

{

var forumButton = wait.Until(d=>d.FindElement(By.XPath("//\*[@id=\"dashboards\_show\"]/section/div/div/div[3]/div/div[2]/div[2]/div[3]/div[2]/a")));

forumButton.Click();

wait.Until(d => d.Url.Contains("/forum"));

Assert.That(driver.Url, Does.Contain("/forum"));

}

[Test]

public void Search\_Anime\_And\_Go\_To\_Animes\_Url()

{

var search = wait.Until(d => d.FindElement(By.CssSelector("#dashboards\_show > header > div.menu-icon.search.mobile")));

((IJavaScriptExecutor)driver).ExecuteScript("arguments[0].click();", search);

string searchText = "Наруто";

var searchInput = wait.Until(d => d.FindElement(By.XPath("//\*[@id=\"dashboards\_show\"]/header/div[3]/label/input")));

searchInput.SendKeys(searchText);

searchInput.SendKeys(Keys.Enter);

wait.Until(d=>d.Url.Contains("/animes?search"));

var anime = wait.Until(d =>d.FindElement(By.CssSelector("a[href\*='/animes/z20-naruto']")));

anime.Click();

wait.Until(d => d.Url.Contains("/z20-naruto"));

Assert.That(driver.Url, Does.Contain("/z20-naruto"));

}

[Test]

public void Add\_To\_List\_Button\_Not\_Changing\_If\_User\_Not\_Authorized()

{

driver.Navigate().GoToUrl("https://shikimori.one/animes/z20-naruto");

try

{

var signInButton = wait.Until(d => d.FindElement(By.CssSelector("header a[href\*='/users/sign\_in']")));

Assert.IsTrue(signInButton.Displayed);

}

catch (WebDriverTimeoutException)

{

Assert.Fail("Нет кнопки входа");

}

var addButton = wait.Until(d => d.FindElement(By.XPath("//\*[@id=\"animes\_show\"]/section/div/div[2]/div/div/div[1]/div[1]/div[1]/div[2]/div/div/form/div[1]")));

string addButtonText = addButton.Text;

addButton.Click();

var currentAddButton = addButton.Text;

//Текст кнопки не должен меняться, так как неавторизированному пользователю нельзя добалять аниме в списки

Assert.That(currentAddButton, Is.EqualTo(addButtonText));

}

}

}