

Министерство образования Республики Беларусь
Учреждение образования
БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Факультет компьютерных систем и сетей
Кафедра программного обеспечения информационных технологий
Дисциплина: Тестирование веб-ориентированных приложений (ТВОП)

ОТЧЕТ

По лабораторной работе №8

Тема работы: «Автоматизация тестирования»

Выполнили:
студенты: гр. 051006

Дранкевич А. А.
Хотин А. А.
Шуляк А. В.

Проверил:

Русина Н. В.

Минск 2023

1 ЦЕЛЬ

С использованием (на выбор) Selenium IDE, Selenium WebDriver или любого иного средства автоматизировать тестирование веб-ориентированного приложения <http://svyatoslav.biz/testlab/wt/>

2 ЧЕК-ЛИСТ

- a. Главная страница содержит слова «menu» и «banners».
- b. В нижней ячейке таблицы присутствует текст «CoolSoft by Somebody».
- c. По умолчанию все текстовые поля формы пусты, а значение поля «Пол» не выбрано.
- d. После заполнения поля «Рост» значением «50» и вес значением «3» и отправки формы, форма исчезает, а вместо неё появляется надпись «Слишком большая масса тела».
- e. Главная страница приложения сразу после открытия содержит форму с тремя текстовыми полями, одной группой из двух радио-баттонов и одной кнопкой.
- f. При неверном вводе значений веса и/или роста появляются сообщения о том, что рост может быть в диапазоне «50-300 см», а вес – в диапазоне «3-500 кг».
- g. Главная страница содержит текущую дату в формате «DD.MM.YYYY».

3 ТЕСТ-КЕЙСЫ

Идентификатор	Приоритет	Связанное с тест-кейсом	Модуль	Подмодуль	Описание (заглавие + предусловия + шаги)	Ожидаемые результаты
1	A	a	Пользовательский интерфейс	Содержание пользовательского интерфейса	Наличие слов «menu» и «banners» Приготовление: открыть в браузере сайт http://svyatoslav.biz/testlab/wt/	1. Открывается главная страница ПС без ошибок и аварийных завершений. 2. Главная страница содержит слова «menu» и «banners».
2	A	b	Пользовательский интерфейс	Содержание пользовательского интерфейса	Наличие текста «CoolSoft by Somebody» Приготовление: открыть в браузере сайт http://svyatoslav.biz/testlab/wt/	1. Открывается главная страница ПС без ошибок и аварийных завершений. 2. Главная страница содержит таблицу. 3. Таблица в своём последнем элементе верхнего уровня содержит

						текст «CoolSoft by Somebody»
3	A	c	Пользовательский интерфейс	Начальное состояние пользовательского интерфейса	Начальное состояние пользовательского интерфейса Приготовление: открыть в браузере сайт http://svyatoslav.biz/testlab/wt/	1. Открывается главная страница ПС без ошибок и аварийных завершений. 2. Поля «Имя», «Рост», «Вес» пусты 3. Значение поля «Пол» не выбранно
4	A	d	Результаты расчётов	Корректность расчётов	Расчёт при минимальных значениях Приготовление: открыть в браузере сайт http://svyatoslav.biz/testlab/wt/ 1. Ввести в поле «Имя» значение «TestUser» 2. Ввести в поле «Рост» значение «50» 3. Ввести в поле «Вес» значение «3» 4. Выбрать пол «М» 5. Нажать кнопку «Расчитать»	1. После нажатия на кнопку «Расчитать» форма исчезает и появляется форма с результатом 2. В форме с результатом содержится надпись «Слишком большая масса тела»
5	A	e	Пользовательский интерфейс	Начальное состояние пользовательского интерфейса	Содержание пользовательского интерфейса при старте приложения Приготовление: открыть в браузере сайт http://svyatoslav.biz/testlab/wt/	1. Открывается главная страница ПС без ошибок и аварийных завершений. 2. Главная страница содержит форму, с тремя текстовыми полями, одной группой из двух радиобаттонов и одной кнопкой
6	A	f	Область исходных данных	Неверные исходные данные	Неверное значение роста Приготовление: открыть в браузере сайт http://svyatoslav.biz/testlab/wt/ 1. Ввести в поле «Имя» значение «TestUser» 2. Ввести в поле «Рост» значение «0» 3. Ввести в поле «Вес» значение «3» 4. Выбрать пол «М» 5. Нажать кнопку «Расчитать»	1. После нажатия на кнопку «Расчитать» появляется надпись, содержащая текст «Рост должен быть в диапазоне 50-300 см.»
7	A	f	Область исходных данных	Неверные исходные данные	Неверное значение веса Приготовление: открыть в браузере сайт http://svyatoslav.biz/testlab/wt/	1. После нажатия на кнопку «Расчитать» появляется надпись, содержащая текст

				е данные	1. 1. Ввести в поле «Имя» значение «TestUser» 2. Ввести в поле «Рост» значение «50» 3. Ввести в поле «Вес» значение «0» 4. Выбрать пол «М» 5. Нажать кнопку «Рассчитать»	«Вес должен быть в диапазоне 3-500 кг..»
8	A	f	Область исходных данных	Неверные исходные данные	Неверное значение роста и веса Приготовления: открыть в браузере сайт http://svyatoslav.biz/testlab/wt/ 1. 1. Ввести в поле «Имя» значение «TestUser» 2. Ввести в поле «Рост» значение «0» 3. Ввести в поле «Вес» значение «0» 4. Выбрать пол «М» 5. Нажать кнопку «Рассчитать»	1. После нажатия на кнопку «Рассчитать» появляется надпись, содержащая текст «Рост должен быть в диапазоне 50-300 см.», и надпись, содержащая текст «Вес должен быть в диапазоне 3-500 кг.»
9	A	g	Пользовательские данные	Содержание пользовательского интерфейса	Значение и формат даты Приготовления: открыть в браузере сайт http://svyatoslav.biz/testlab/wt/	1. Открывается главная страница ПС без ошибок и аварийных завершений 2. Главная страница содержит текущую дату в формате «DD.MM.YYYY»

4 ОТЧЁТЫ О ДЕФЕКТАХ

Идентификатор	Кратное описание	Шаги по воспроизведению	Воспроизводимость	Важность	Срочность	Симптомы	Возможность обойти	Комментарии	Приложения
4	Отображается неправильный текст	<p>Приготовления: открыть в браузере сайт http://svyatoslav.biz/testlab/wt/</p> <ol style="list-style-type: none"> 1. Ввести в поле «Имя» значение «TestUser» 2. Ввести в поле «Рост» значение «50» 3. Ввести в поле «Вес» значение «3» 4. Выбрать пол «М» 5. Нажать кнопку «Рассчитать» 	Всегда	Средняя	Обычная	Некорректная операция	Нет	Ожидаемый результат – текст «Слишком большая масса тела». Полученный результат – текст «Слишком малая масса тела»	<div> <div>Расчёт веса Сегодня: 01.03.2023</div> <div> <div>menu</div> <div>Слишком малая масса тела</div> <div>banners</div> </div> <div>© CoolSoft by Somebody fhlrhwelrwerhwerh</div> </div>

9	Отображается не GMT +3 время	Приготовления: открыть в браузере сайт http://svyatoslav.biz/testlab/wt/	Всегда	Средняя	Обычная	Проблема в документации	Нет	Не указано какое именно должно отображаться время	 <p>The screenshot shows two web pages side-by-side. The left page is a weight calculation tool with fields for Name, Height, Weight, and Gender (Male/Female), and a 'Calculate' button. The right page is a time and date utility showing a clock, the current time (00:00:28 MSK), the date (Friday, 24 February 2023), and the country (Belarus). Both pages are displayed in a browser window with a dark theme.</p>
---	------------------------------	--	--------	---------	---------	-------------------------	-----	---	--

Приложение 1

Исходный код автоматизированных тестов

```
using Newtonsoft.Json.Bson;
using OpenQA.Selenium;
using OpenQA.Selenium.DevTools.V108.Network;
using OpenQA.Selenium.Firefox;
using System.Globalization;
using System.Security.Cryptography.X509Certificates;
using System.Text;

namespace _08
{
    public class Tests
    {
        // SetUp
        private FirefoxDriverService _service;
        private FirefoxOptions _options;

        // Test A
        private By _txtMenu = By.XPath("//*[@contains(text(),'menu')]");
        private By _txtBanners = By.XPath("//*[@contains(text(),'banners')]");

        // Test B
        private By _lastTableElement =
By.XPath("//*[@table//tr[last()]/td[last()][contains(text(),'CoolSoft by
Somebody')]]");

        // Test C
        private By _txtFieldName = By.XPath("//input[@name='name']");
        private By _txtFieldHeight = By.XPath("//input[@name='height']");
        private By _txtFieldWeight = By.XPath("//input[@name='weight']");
        private By _rbMale = By.XPath("//input[@name='gender' and
@value='m']");
        private By _rbFemale = By.XPath("//input[@name='gender' and
@value='f']");

        // Test D
        // (using Test C fields)
        private By _btnCalculate = By.XPath("//input[@type='submit' and
@value='Рассчитать']");
        private By _txtTooBigWeight = By.XPath("//*[@text()='Слишком большая
масса тела']");

        // Test E
        private By _allTxtFields = By.XPath("//input[@type='text']");
        private By _allRbs = By.XPath("//input[@type='radio']");
        private By _allBtns = By.XPath("//input[@type='submit']");

        // Test F
        // (using Text C fields)
        private By _txtErrorHeight = By.XPath("//*[@text()='Рост должен быть в
диапазоне 50-300 см.']").);
        private By _txtErrorWeight = By.XPath("//*[@text()='Вес должен быть в
диапазоне 3-500 кг.']").);

        // Test G
        private By _txtDate = By.XPath("//*[@table//*[contains(text(),'Расчёт
веса')]]");

        [SetUp]
        public void Setup ()
        {
```

```

        _service = FirefoxDriverService.CreateDefaultService
("d:\\BSUIR\\6_Sem\\WAT\\Projects\\_08\\_08\\bin\\Debug\\net6.0",
"geckodriver.exe");
        _service.FirefoxBinaryPath = @"C:\Program Files\Mozilla
Firefox\firefox.exe";
        _options = new FirefoxOptions ();
    }

    [Test]
    public void TestA()
    {
        var options = _options;
        options.AddArgument("-headless");
        var webDriver = new FirefoxDriver(_service, options);
        webDriver.Navigate().GoToUrl
("http://svyatoslav.biz/testlab/wt/");
        try
        {
            var txtMenu = webDriver.FindElement(_txtMenu);
            var txtBanners = webDriver.FindElement(_txtBanners);

            Assert.Multiple( () => {
                Assert.That(txtMenu, Is.Not.Null);
                Assert.That(txtBanners, Is.Not.Null);
            });
        }
        catch (OpenQA.Selenium.NoSuchElementException ex)
        {
            Assert.Fail();
        }
        finally
        {
            webDriver.Quit();
        }
    }

    [Test]
    public void TestB()
    {
        var options = _options;
        options.AddArgument("-headless");
        var webDriver = new FirefoxDriver(_service, options);
        webDriver.Navigate().GoToUrl
("http://svyatoslav.biz/testlab/wt/");
        try
        {
            var lastTableElement =
webDriver.FindElement(_lastTableElement);

            Assert.That(lastTableElement, Is.Not.Null);
        }
        catch (OpenQA.Selenium.NoSuchElementException ex)
        {
            Assert.Fail();
        }
        finally
        {
            webDriver.Quit();
        }
    }

    [Test]
    public void TestC()
    {

```



```

        var options = _options;
        options.AddArgument("-headless");
        var webDriver = new FirefoxDriver(_service, options);
        webDriver.Navigate().GoToUrl
("http://svyatoslav.biz/testlab/wt/");
        try
        {
            IWebElement txtFieldName =
(IWebElement)webDriver.FindElement(_txtFieldName);
            IWebElement txtFieldHeight =
(IWebElement)webDriver.FindElement(_txtFieldHeight);
            IWebElement txtFieldWeight =
(IWebElement)webDriver.FindElement(_txtFieldWeight);

            Assert.Multiple(() => {
                Assert.That(txtFieldName.GetAttribute("value"),
Is.EqualTo(""));
                Assert.That(txtFieldHeight.GetAttribute("value"),
Is.EqualTo(""));
                Assert.That(txtFieldWeight.GetAttribute("value"),
Is.EqualTo(""));
            });

            IWebElement rbMale =
(IWebElement)webDriver.FindElement(_rbMale);
            IWebElement rbFemale =
(IWebElement)webDriver.FindElement(_rbFemale);

            Assert.Multiple(() => {
                Assert.That(rbMale.Selected, Is.EqualTo(false));
                Assert.That(rbFemale.Selected, Is.EqualTo(false));
            });
        }
        catch (OpenQA.Selenium.NoSuchElementException ex)
        {
            Assert.Fail();
        }
        finally
        {
            webDriver.Quit();
        }
    }

    [Test]
    public void TestD()
    {
        var options = _options;
        options.AddArgument("-headless");
        var webDriver = new FirefoxDriver(_service, options);
        webDriver.Navigate().GoToUrl
("http://svyatoslav.biz/testlab/wt/");
        try
        {
            IWebElement txtFieldName =
(IWebElement)webDriver.FindElement(_txtFieldName);
            IWebElement txtFieldHeight =
(IWebElement)webDriver.FindElement(_txtFieldHeight);
            IWebElement txtFieldWeight =
(IWebElement)webDriver.FindElement(_txtFieldWeight);

            txtFieldName.SendKeys("Test User");
            txtFieldHeight.SendKeys("50");
            txtFieldWeight.SendKeys("3");

```

```

        IWebElement rbMale =
(IWebElement)webDriver.FindElement(_rbMale);

        rbMale.Click();

        IWebElement btnCalculate =
(IWebElement)webDriver.FindElement(_btnCalculate);

        btnCalculate.Click();

        IWebElement txtTooBigWeight =
(IWebElement)webDriver.FindElement(_txtTooBigWeight);

    }
    catch (OpenQA.Selenium.NoSuchElementException ex)
    {
        Assert.Fail();
    }
    finally
    {
        webDriver.Quit();
    }
}

[Test]
public void TestE()
{
    var options = _options;
    options.AddArgument("-headless");
    var webDriver = new FirefoxDriver(_service, options);
    webDriver.Navigate().GoToUrl
("http://svyatoslav.biz/testlab/wt/");
    try
    {
        var allTxtFields = webDriver.FindElements(_allTxtFields);

        Assert.That(allTxtFields.Count, Is.EqualTo(3));

        var allRbs = webDriver.FindElements(_allRbs);

        Assert.Multiple( () => {
            Assert.That(allRbs.Count, Is.EqualTo(2));
            Assert.That(allRbs[0].GetAttribute("name"),
Is.EqualTo(allRbs[1].GetAttribute("name")));
        });

        var allBtns = webDriver.FindElements(_allBtns);

        Assert.That(allBtns.Count, Is.EqualTo(1));
    }
    catch (OpenQA.Selenium.NoSuchElementException ex)
    {
        Assert.Fail();
    }
    finally
    {
        webDriver.Quit();
    }
}

[Test]
public void TestF()
{
    var options = _options;

```

```

        options.AddArgument("-headless");
        var webDriver = new FirefoxDriver(_service, options);
        webDriver.Navigate().GoToUrl
("http://svyatoslav.biz/testlab/wt/");
        try
        {
            // Assert Error Height and Error Weight
            IWebElement txtFieldHeight =
(IWebElement)webDriver.FindElement(_txtFieldHeight);
            IWebElement txtFieldWeight =
(IWebElement)webDriver.FindElement(_txtFieldWeight);

            txtFieldHeight.SendKeys("0");
            txtFieldWeight.SendKeys("0");

            IWebElement btnCalculate =
(IWebElement)webDriver.FindElement(_btnCalculate);

            btnCalculate.Click();

            var txtErrorHeight =
webDriver.FindElement(_txtErrorHeight);
            var txtErrorWeight =
webDriver.FindElement(_txtErrorWeight);

            Assert.Multiple( () => {
                Assert.That(txtErrorHeight, Is.Not.Null);
                Assert.That(txtErrorWeight, Is.Not.Null);
            });

            // Assert Error Height
            txtFieldHeight =
(IWebElement)webDriver.FindElement(_txtFieldHeight);
            txtFieldWeight =
(IWebElement)webDriver.FindElement(_txtFieldWeight);

            txtFieldHeight.SendKeys("0");
            txtFieldWeight.SendKeys("50");

            txtErrorHeight = webDriver.FindElement(_txtErrorHeight);

            Assert.That(txtErrorHeight, Is.Not.Null);

            // Assert Error Weight
            txtFieldHeight =
(IWebElement)webDriver.FindElement(_txtFieldHeight);
            txtFieldWeight =
(IWebElement)webDriver.FindElement(_txtFieldWeight);

            txtFieldHeight.SendKeys("50");
            txtFieldWeight.SendKeys("0");

            txtErrorWeight = webDriver.FindElement(_txtErrorWeight);

            Assert.That(txtErrorWeight, Is.Not.Null);
        }
        catch (OpenQA.Selenium.NoSuchElementException ex)
        {
            Assert.Fail();
        }
        finally
        {
            webDriver.Quit();
        }

```

```

    }

[Test]
public void TestG()
{
    var options = _options;
    options.AddArgument("-headless");
    var webDriver = new FirefoxDriver(_service, options);
    webDriver.Navigate().GoToUrl
("http://svyatoslav.biz/testlab/wt/");
    try
    {
        IWebElement txtDate =
(IWebElement)webDriver.FindElement(_txtDate);
        int startDateId = 0, i = 0;
        while (i < txtDate.Text.Length && startDateId == 0)
        {
            if (txtDate.Text[i] >= '0' && txtDate.Text[i] <=
'9')
                startDateId = i;

            i++;
        }
        string date = txtDate.Text.Substring(startDateId);

        DateTime resultDateTime;
        string format = "dd.MM.yyyy";
        if (DateTime.TryParseExact(date, format,
CultureInfo.InvariantCulture, DateTimeStyles.None, out resultDateTime))
        {
            DateTime currDateTime = DateTime.Now;
            Assert.Multiple( () => {
                Assert.That(currDateTime.Day,
Is.EqualTo(resultDateTime.Day));
                Assert.That(currDateTime.Month,
Is.EqualTo(resultDateTime.Month));
                Assert.That(currDateTime.Year,
Is.EqualTo(resultDateTime.Year));
            });
        }
        else
            Assert.Fail();
    }
    catch (OpenQA.Selenium.NoSuchElementException ex)
    {
        Assert.Fail();
    }
    finally
    {
        webDriver.Quit();
    }
}
}
}

```