**5 ЛАБОРАТОРНАЯ РАБОТА №5**

**«Исследование способов модульного тестирования программного**

**обеспечения в среде NUnit»**

**5.1 Цель работы**

Исследовать эффективность использования методологии TDD при разработке программного обеспечения. Получить практические навыки использования фреймворка NUnit для модульного тестирования программного обеспечения.

**5.2 Вариант задания**

Реализовать на языке C# один из классов, спроектированных в лабораторной работе № 1. Методы класса при этом не реализовывать. Разработать для созданного класса набор модульных тестов, включающий тесты для каждого метода. Запустить набор тестов, проанализировать и сохранить результаты. Поочередно реализовать методы класса, выполняя тестирование при каждом изменении программного кода. После того, как весь набор тестов будет выполняться успешно, реализацию классов можно считать завершенной.

**5.3 Ход выполнения работы**

5.3.1 В начале выполнения работы для класса, изменяющего амперсанты в строке на знаки вопроса была выделена функция, которая выполняет эту операцию. Код тестового класса представлен в листинге 5.1.

Листинг 5.1 – Код тестового класса

using Lab3;

namespace TestLab3;

public class ChangerUnitTest

{

[Fact]

public void Test1()

{

// Arrange

var stringChanger = new StringChanger();

var input = "a";

var expected = "a";

// Act

var result = stringChanger.ChangeSymbols(input);

// Assert

Assert.Equal(expected, result);

}

[Fact]

public void Test2()

{

// Arrange

var stringChanger = new StringChanger();

var input = "?";

var expected = "&";

// Act

var result = stringChanger.ChangeSymbols(input);

// Assert

Assert.Equal(expected, result);

}

[Fact]

public void Test3()

{

// Arrange

var stringChanger = new StringChanger();

var input = "abcde";

var expected = "abcde";

// Act

var result = stringChanger.ChangeSymbols(input);

// Assert

Assert.Equal(expected, result);

}

[Fact]

public void Test4()

{

// Arrange

var stringChanger = new StringChanger();

var input = "?ab23ef";

var expected = "&ab23ef";

// Act

var result = stringChanger.ChangeSymbols(input);

// Assert

Assert.Equal(expected, result);

}

[Fact]

public void Test5()

{

// Arrange

var stringChanger = new StringChanger();

var input = "1bc?defg";

var expected = "1bc&defg";

// Act

var result = stringChanger.ChangeSymbols(input);

// Assert

Assert.Equal(expected, result);

}

[Fact]

public void Test6()

{

// Arrange

var stringChanger = new StringChanger();

var input = "123bcdefg?";

var expected = "123bcdefg&";

// Act

var result = stringChanger.ChangeSymbols(input);

// Assert

Assert.Equal(expected, result);

}

[Fact]

public void Test7()

{

// Arrange

var stringChanger = new StringChanger();

var input = "??ab???c1d???";

var expected = "&&ab&&&c1d&&&";

// Act

var result = stringChanger.ChangeSymbols(input);

// Assert

Assert.Equal(expected, result);

}

}

Далее был реализован тестируемый класс StringChanger с упомянутым ранее методом, он показан в листинге 5.2.

Листинг 5.2 – Код класса StringChanger

namespace Lab3;

public class StringChanger

{

public string ChangeSymbols(string str)

{

if (str == null) throw new ArgumentNullException(nameof(str));

return str.Replace('?', '&');

}

}

На рисунке 5.1 показан результат работы тестов, написанных ранее для класса StringChanger.

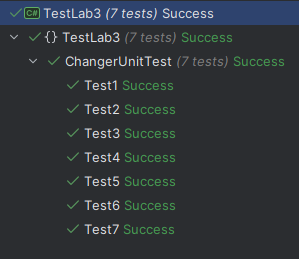


Рисунок 5.1 – Результаты тестов

**Выводы**

В ходе лабораторной работы была исследована эффективность использования методологии TDD при разработке программного обеспечения. Также были получены практические навыки использования фреймворка NUnit для модульного тестирования программного обеспечения. В конце выполнения лабораторной работы был написан отчет.