Министерство образования Российской Федерации

Московский государственный институт электронной техники

(технический университет)

утверждаю

Зав. Кафедрой СПИНТех,

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

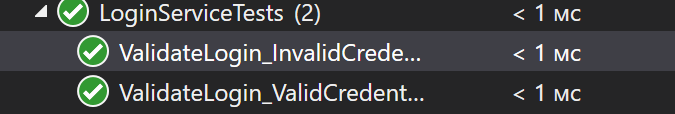
«\_\_»\_\_\_\_\_\_\_\_\_2023 г.

ЛАБОраторная работа 6

Разработать программу в соответствии со своим вариантом и подготовить к ней набор модульных тестов. Провести Unit-тестирование и анализ покрытия кода в модульных тестах.

**Тест класса LoginService:**

|  |
| --- |
| [TestClass]  public class LoginServiceTests  {  [TestMethod]  public void ValidateLogin\_ValidCredentials\_ReturnsTrue() //позитивный тест логина  {  // Arrange  LoginService loginService = new LoginService();  // Act  bool result = loginService.ValidateLogin("1", "1");  // Assert  Assert.IsTrue(result);  }  [TestMethod]  public void ValidateLogin\_InvalidCredentials\_ReturnsFalse() //негативный тест логина  {  // Arrange  LoginService loginService = new LoginService();  // Act  bool result = loginService.ValidateLogin("11111111", "22222");  // Assert  Assert.IsFalse(result);  } |



**Тест класса FileHelper:**

|  |
| --- |
| [TestClass]  public class FileHelperTests  {  [TestMethod]  public void TestReadAllLines()  {  // Arrange  string path = @"C:\Users\ant\_daddy\OneDrive\Рабочий стол\save\_me\Kassa\Kassa\db.txt";  // Act  string[] result = FileHelper.ReadAllLines(path);  // Assert  Assert.IsNotNull(result);  Assert.IsTrue(result.Length > 0);  }  } |



**Тест класса DataProcessor:**

|  |
| --- |
| [TestClass]  public class DataProcessorTests  {  [TestMethod]  public void TestFindDataByBarcode() // тест поиска  {  // Arrange  string[] lines = new string[] { "12345,Product1,10.00", "67890,Product2,20.00" };  string barcode = "12345";  // Act  string[] result = DataProcessor.FindDataByBarcode(lines, barcode);  // Assert  Assert.IsNotNull(result);  Assert.AreEqual("12345", result[0]);  Assert.AreEqual("Product1", result[1]);  Assert.AreEqual("10.00", result[2]);  }  [TestMethod]  public void TestCalculateTotalPrice()  {  // Arrange  decimal price = 10.00m;  int quantity = 3;  // Act  decimal result = DataProcessor.CalculateTotalPrice(price, quantity);  // Assert  Assert.AreEqual(30.00m, result);  }  } |

