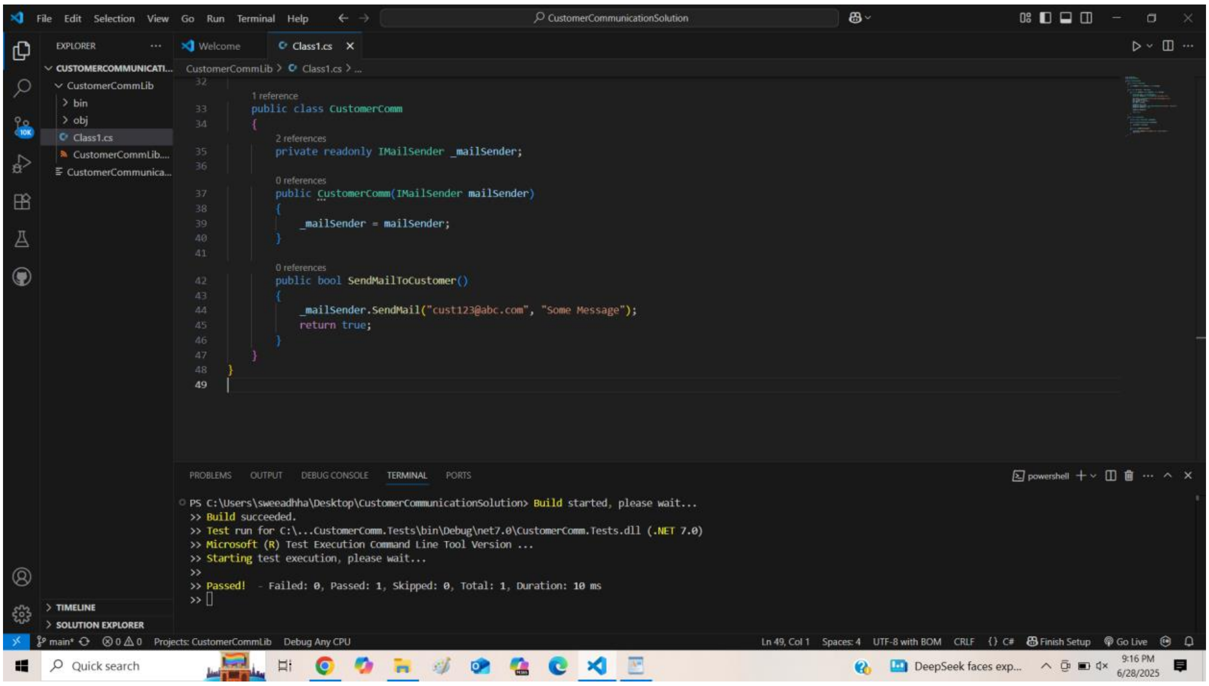
MOQ unit Testing

Code:

```
using System.Net;
using System.Net.Mail;

namespace CustomerCommLib
{
    public interface IMailSender
    {
        bool SendMail(string toAddress, string message);
    }

    public class MailSender : IMailSender
    {
        public bool SendMail(string toAddress, string message)
        {
            MailMessage mail = new MailMessage();
            SmtpClient SmtpServer = new SmtpClient("smtp.gmail.com");

            mail.From = new MailAddress("your_email_address@gmail.com");
            mail.To.Add(toAddress);
            mail.Subject = "Test Mail";
            mail.Body = message;

            SmtpServer.Port = 587;
            SmtpServer.Credentials = new NetworkCredential("username",
"password");
            SmtpServer.EnableSsl = true;

            SmtpServer.Send(mail);

            return true;
        }
    }

    public class CustomerComm
    {
        private readonly IMailSender _mailSender;

        public CustomerComm(IMailSender mailSender)
        {
            _mailSender = mailSender;
        }
        public bool SendMailToCustomer()
        {
            _mailSender.SendMail("cust123@abc.com", "Some Message");
            return true;
        }
    }
}

using Moq;
using NUnit.Framework;
using CustomerCommLib;

namespace CustomerComm.Tests
{
    [TestFixture]
    public class CustomerCommTests
    {
        private Mock<IMailSender> _mockMailSender;
        private CustomerComm _customerComm;

        [OneTimeSetUp]
        public void Init()
        {
            _mockMailSender = new Mock<IMailSender>();
            _mockMailSender
                .Setup(x => x.SendMail(It.IsAny<string>(),
It.IsAny<string>()))
                .Returns(true);

            _customerComm = new CustomerComm(_mockMailSender.Object);
        }

        [Test]
        public void SendMailToCustomer_ShouldReturnTrue()
        {
            bool result = _customerComm.SendMailToCustomer();

            Assert.IsTrue(result);
            _mockMailSender.Verify(
                x => x.SendMail(It.IsAny<string>(), It.IsAny<string>()),
                Times.Once
            );
        }
    }
}
```

[redacted]

Output :



File Edit Selection View Go Run Terminal Help
CustomerCommunicationSolution
EXPLORER
CUSTOMERCOMMUNICATI...
CustomerCommLib
bin
obj
Class1.cs
CustomerCommLib....
CustomerCommunica...
TIMELINE
SOLUTION EXPLORER
Welcome
Class1.cs
CustomerCommLib > Class1.cs > ...
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
powershell
main* ⊗ 0 ⚠ 0 Projects: CustomerCommLib Debug Any CPU
Ln 49, Col 1 Spaces: 4 UTF-8 with BOM CRLF {} C# Finish Setup Go Live
Quick search
DeepSeek faces exp...
9:16 PM
6/28/2025

```
    1 reference
    public class CustomerComm
    {
        2 references
        private readonly IMailSender _mailSender;

        0 references
        public CustomerComm(IMailSender mailSender)
        {
            _mailSender = mailSender;
        }

        0 references
        public bool SendMailToCustomer()
        {
            _mailSender.SendMail("cust123@abc.com", "Some Message");
            return true;
        }
    }
}
```

```
PS C:\Users\sweeadhha\Desktop\CustomerCommunicationSolution> Build started, please wait...
>> Build succeeded.
>> Test run for C:\...CustomerComm.Tests\bin\Debug\net7.0\CustomerComm.Tests.dll (.NET 7.0)
>> Microsoft (R) Test Execution Command Line Tool Version ...
>> Starting test execution, please wait...
>>
>> Passed!  - Failed: 0, Passed: 1, Skipped: 0, Total: 1, Duration: 10 ms
>> 
```