

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«САМАРСКИЙ УНИВЕРСИТЕТ»

Отчёт по лабораторной работе №4
«Отправка электронной почты»
по курсу «Сети ЭВМ и телекоммуникации»

Выполнил:
Власов Г.В.
гр.6303

Проверил:
Пигусов А.С.

Самара 2019

Цель работы:

Приобретение навыков работы с простым протоколом отправки электронной почты SMTP.

Задание:

Разработать программу (почтовый клиент), позволяющую пользователю отправлять электронную почту по протоколу SMTP, используя существующие в сети Internet почтовые сервера. Программа должна поддерживать возможность аутентификации пользователя на сервере при отправке электронной почты в случае, если она требуется на выбранном пользователем сервере исходящей электронной почты. Программа должна поддерживать возможность отправки одного и того же письма нескольким пользователям и приложение к письму как минимум одного файла. Адреса получателей, тема и текст письма, месторасположение прикладываемого к письму файла и адрес почтового сервера указываются пользователем. Приложение должно включать контроль ошибок при вводе и обработке запросов.

Листинг программы:

```
using System;
using System.Collections.Generic;
using System.Net;
using System.Net.Mail;
using System.Windows.Forms;
namespace MailClient
{
    public partial class Form1 : Form
    {
        List<string> attachments = new List<string>();
        List<string> recievers = new List<string>();
        public Form1()
        {
            InitializeComponent();
            listBox_Recievers.Visible = false;
            label17.Visible = false;
        }
        private void button_Send_Click(object sender, EventArgs e)
        {
            try
            {
                if (int.TryParse(textBox_Port.Text, out int n))
                {
                    SmtplibClient client = new SmtplibClient(textBox_SMTP.Text,
int.Parse(textBox_Port.Text));
                    client.EnableSsl = true;
                    client.Timeout = 20000;
                    client.DeliveryMethod = SmtplibDeliveryMethod.Network;
                    client.UseDefaultCredentials = false;
                    client.Credentials = new NetworkCredential(textBox_Login.Text,
textBox_Password.Text);
                    MailMessage msg = new MailMessage();
                    if (attachments.Count != 0)
                    {
                        for (int i = 0; i < attachments.Count; i++)
```

```

        {
            msg.Attachments.Add(new Attachment(attachments[i]));
        }
    }
    if(recievers.Count !=0 )
    {
        for (int i = 0; i < recievers.Count; i++)
        {
            msg.To.Add(recievers[i]);
        }
    }
    else
    {
        msg.To.Add(textBox_To.Text);
    }
    msg.From = new MailAddress(textBox_Login.Text);
    msg.Subject = textBox_Sub.Text;
    msg.Body = textBox_Msg.Text;
    client.Send(msg);
    MessageBox.Show("Сообщение успешно доставлено!");
    attachments.Clear();
    listBox_Attachments.SelectedItems.Clear();
    listBox_Recievers.SelectedItems.Clear();
    recievers.Clear();
    listBox_Recievers.Visible = false;
    label7.Visible = false;
}
else
{
    MessageBox.Show("Проверьте правильность ввода данных");
}
}
catch (Exception ex)
{
    MessageBox.Show(ex.Message);
}
}
private void button_Atch_Click(object sender, EventArgs e)
{
    OpenFileDialog ofd = new OpenFileDialog();
    if (ofd.ShowDialog() == DialogResult.OK)
    {
        attachments.Add(ofd.FileName);
        listBox_Attachments.Items.Add(attachments.Count + ") " + ofd.FileName);
    }
}
private void button1_Click(object sender, EventArgs e)
{
    listBox_Recievers.Visible = true;
    label7.Visible = true;
    recievers.Add(textBox_To.Text);
    listBox_Recievers.Items.Add(recievers.Count + ") " + textBox_To.Text);
    textBox_To.Clear();
}
private void Form1_Load(object sender, EventArgs e)
{
}
}
}
}

```

Результаты выполнения программы:

