# ****Разработка почтового клиента на языке С#.****

# **Отправка электронной почты. Протокол SMTP. Классы SmtpClient, MailMessage.**

**Цель работы:** приобретение навыков работы с простым протоколом отправки электронной почты SMTP.

**Почтовый клиент** – программное обеспечение, устанавливаемое на компьютере пользователя и предназначенное для получения, написания, отправки и хранения сообщений электронной почты.

**Протокол SMTP** (*англ. Simple Mail Transfer Protocol* – простой протокол передачи почты) – это широко используемый сетевой протокол, предназначенный для передачи электронной почты в сетях TCP/IP.

Класс **SmtpClient**

Для работы с протоколом SMTP и отправки электронной почты в .NET предназначен класс **SmtpClient** из пространства имен **System.Net.Mail**.

Этот класс определяет ряд свойств, которые *позволяют настроить отправку*:

|  |  |
| --- | --- |
| *Host* | SMTP-сервер, с которого производится отправление почты. Например, smtp.gmail.com |
| *Port* | Порт, используемый SMTP-сервером. По умолчанию используется 25 порт, что соответствует незашифрованному каналу. |
| *EnableSsl* | Указывает, будет ли использоваться протокол SSL при отправке. |
| *Credentials* | Аутентификационные данные отправителя. |

Класс **MailMessage**

Еще одним ключевым классом, который используется при отправке, является – **MailMessage**. Данный класс представляет собой *отправляемое сообщение*. Среди его свойств можно выделить следующие:

|  |  |
| --- | --- |
| *From* | Адрес отправителя в виде экземпляра класса MailAddress. |
| *To* | Адрес получателя. Также представляет объект MailAddress. |
| *Subject* | Тема или заголовок письма типа String. |
| *Body* | Тело письма, т.е. само сообщение. |
| *Attachments* | Коллекция присоединенных к письму файлов. |
| *IsBodyHtml* | указывает, представляет ли письмо содержимое с кодом html |
| *DeliveryMethod* | Метод обработки исходящих сообщений электронной почты. Установите *SmtpDeliveryMethod.Network,* что означает передачу email по сети на SMTP-сервер. |

Класс **Attachment**

Класс **Attachment** реализует присоединение файла к письму. Если хотите переслать файл, нужно добавить **Attachment** вашего файла в список MailMessage.Attachments.

При отправке письма с присоединенным файлом желательно сжимать его в архив. Это увеличит скорость передачи.

**ЗАДАНИЕ**

С помощью классов пространства имен System.Net.Mailразработать 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 адресатам 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.

В случае ошибки при отправке письма выводить соответствующее сообщение.

**ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ**

Создайте пустое консольное приложение и подключите следующие библиотеки:

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

Добавьте в проект следующий код:

int port = 587; // порт smtp-сервера (в случае mail.ru это 587)  
bool enableSSL = true;

string emailFrom = "xxxxxx @ mail.ru"; // адрес почты отправителя письма  
string password = "xxxxxx"; // пароль почты отправителя письма  
string emailTo = "xxxxxx @ xxxx.xx"; // адрес почты получателя письма  
string subject = "Тест"; // тема письма  
string body = "Письмо-тест работы smtp-клиента!"; // текст письма  
string smtpAddress = "smtp.mail.ru"; // адрес smtp-сервера с которого будем отправлять письмо

MailMessage mail = new MailMessage();

mail.From = new MailAddress(emailFrom);  
mail.To.Add(emailTo);  
mail.Subject = subject;  
mail.Body = body;  
mail.IsBodyHtml = true; // можно поставить false, если отправляется только текст

mail.Attachments.Add(new Attachment("C:\\File.txt")); // если нужно прикрепить текстовый файл   
mail.Attachments.Add(new Attachment("C:\\Zip.zip")); // если нужно прикрепить архив

using (SmtpClient smtp = new SmtpClient(smtpAddress, port))  
{

smtp.Credentials = new NetworkCredential(emailFrom, password);  
smtp.EnableSsl = enableSSL;  
try  
{

smtp.Send(mail); // отправка сообщения  
}  
catch (Exception ex)  
{

Console.WriteLine(ex.Message.ToString()); // для вывода ошибки в консоль  
}

}

Сначала объявляем и инициализируем все необходимые переменные, затем создаем объект MailMessage и выполняем соответствующие настройки. Далее отправляем письмо.

В случае ошибки при отправке письма, выводим текст ошибки в консоль.