МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное

образовательное учреждение высшего образования

«НОВОСИБИРСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

ФАКУЛЬТЕТ АВТОМАТИКИ И ВЫЧИСЛИТЕЛЬНОЙ ТЕХНИКИ

Кафедра Автоматизированных систем управления

**ОТЧЕТ О ВЫПОЛНЕНИИ**

**Лабораторной работы № 4**

Дисциплина «*Технологии и методы программирования*»

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Выполнили:** |  | **Проверил:** |  | |
| Студенты | *Бузмаков А.И.,*  *Шестаков К.Д.* | Преподаватель | *Астапчук В. А., Пустовских Д. А.* | |
| Факультет | *АВТФ* |  |  | |
| Направление (специальность) подготовки | *09.03.01 – Информатика и вычислительная техника* | Балл: \_\_\_\_\_\_\_\_\_ | |  |
| Группа | *АП-227* | Оценка *\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_* | | |
| Шифр |  |  |  | |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | | |
| подпись | | подпись | | |
| Дата сдачи: | «17» апреля 2024г. | Дата защиты: | «\_\_»\_\_\_\_\_\_\_\_\_\_20\_\_г. | |

Новосибирск 2024

**Текст задания**

**Задание 1:** Разработать программу-клиент, которая должна:

* запрашивать у пользователя адрес программы-сервера;
* устанавливать соединение с сервером;
* передавать на сервер имя выбранного каталога;
* принимать ответ от сервера и выводить его на экран;
* закрывать соединение с сервером.

**Задание 2:** Разработать программу-сервер, которая должна:

* ожидать запрос от программ-клиентов на соединение;
* устанавливать соединение с клиентами, при установлении соединения передавать список логических устройств клиенту;
* принимать от клиента данные:

если это имя каталога, передавать клиенту структуру каталога (имена файлов и подкаталогов);

если это имя текстового файла, передавать клиенту содержимое файла.

* сообщения, полученные от клиента, выводить на экран;
* закрывать соединение с клиентом при получении уведомления.

**Тексты разработанных программ**

Таблица 1 – Код программы-клиента

|  |
| --- |
| ClientForm.cs |
| using System;  using System.Net.Sockets;  using System.Text;  using System.Windows.Forms;  namespace ClientApp  {  public partial class ClientForm : Form  {  private const int port = 8888;  private TcpClient client;  private NetworkStream stream;  private byte[] data;  public ClientForm()  {  InitializeComponent();  }  private void connectButton\_Click(object sender, EventArgs e)  {  try  {  if (addressBox.Text == "" || dirBox.Text == "")  {  MessageBox.Show("Все поля должны быть заполнены!");  }  else  {  client = new TcpClient();  client.Connect(addressBox.Text, port);  stream = client.GetStream();  string directoryName = dirBox.Text;  data = Encoding.UTF8.GetBytes(directoryName);  stream.Write(data, 0, data.Length);  data = new byte[256];  int bytes = stream.Read(data, 0, data.Length);  string response = Encoding.UTF8.GetString(data, 0, bytes);  MessageBox.Show(response);  stream.Close();  client.Close();  }  }  catch (Exception ex)  {  MessageBox.Show(ex.Message);  }  }  }  } |

Таблица 2 – Код программы-сервера

|  |
| --- |
| ServerForm.cs |
| using System;  using System.IO;  using System.Net;  using System.Net.Sockets;  using System.Text;  using System.Threading;  using System.Windows.Forms;  namespace ServerApp  {  public partial class ServerForm : Form  {  private TcpListener server;  private const int port = 8888;  private bool isRunning = true;  public ServerForm()  {  InitializeComponent();  }  private void LogMessage(string message)  {  if (logBox.InvokeRequired)  {  logBox.Invoke(new MethodInvoker(delegate { LogMessage(message); }));  }  else  {  logBox.AppendText(message + "\n");  }  }  private void StartServer()  {  try  {  server = new TcpListener(IPAddress.Any, port);  server.Start();  LogMessage("Сервер запущен.");  while (isRunning)  {  TcpClient client = server.AcceptTcpClient();  Thread clientThread = new Thread(HandleClient);  clientThread.Start(client);  }  }  catch (Exception ex)  {  MessageBox.Show(ex.Message);  }  }  private void HandleClient(object obj)  {  TcpClient client = (TcpClient)obj;  NetworkStream stream = client.GetStream();  byte[] data = new byte[256];  int bytes = stream.Read(data, 0, data.Length);  string request = Encoding.UTF8.GetString(data, 0, bytes);  // Лог подключения клиента  LogMessage($"Клиент подключен: {((IPEndPoint)client.Client.RemoteEndPoint).Address}");  if (File.Exists(request) && Path.GetExtension(request).ToLower() == ".txt")  {  // Обработка запроса на случай, если был передан текстовый файл  string fileContent = File.ReadAllText(request);  byte[] fileData = Encoding.UTF8.GetBytes(fileContent);  stream.Write(fileData, 0, fileData.Length);  // Сообщение о том, что запрос с текстовым файлом выполнен  LogMessage($"Содержимое текстового файла отправлено клиенту. Запрос: {request}");  }  else if (File.Exists(request) && Path.GetExtension(request).ToLower() != ".txt")  {  // На случай, если переданный файл не является текстовым  LogMessage($"Переданный файл не является текстовым. Запрос: {request}");  }  else if (Directory.Exists(request))  {  // Обработка запроса клиента  string[] files = Directory.GetFiles(request);  string[] directories = Directory.GetDirectories(request);  StringBuilder responseBuilder = new StringBuilder();  responseBuilder.AppendLine($"Файлы в каталоге '{request}':");  foreach (string file in files)  {  responseBuilder.AppendLine(Path.GetFileName(file));  }  responseBuilder.AppendLine($"Директории в каталоге '{request}':");  foreach (string directory in directories)  {  responseBuilder.AppendLine(Path.GetFileName(directory));  }  byte[] responseData = Encoding.UTF8.GetBytes(responseBuilder.ToString());  stream.Write(responseData, 0, responseData.Length);  // Сообщение о том, что запрос каталогом выполнен  LogMessage($"Содержимое каталога передано клиенту. Запрос: {request}");  }  else  {  // Некорректный запрос  LogMessage("Некорректный запрос: " + request);  }  stream.Close();  client.Close();  }  private void startButton\_Click(object sender, EventArgs e)  {  Thread serverThread = new Thread(StartServer);  serverThread.Start();  }  }  } |

**Результаты разработанных программ**

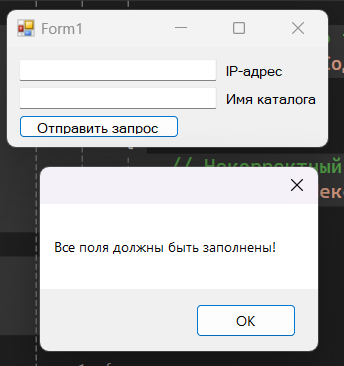


Рисунок 1 - В случае, если поля в приложении-клиенте не были заполнены

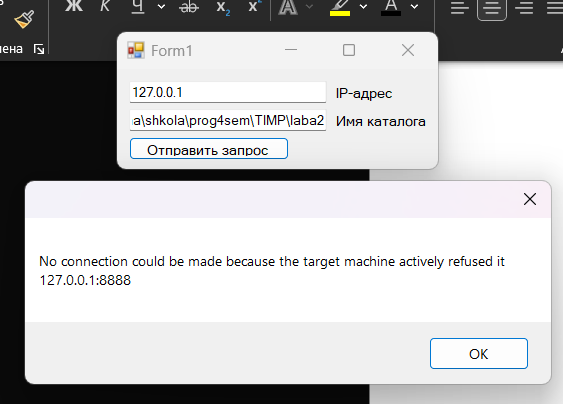


Рисунок 2 - В случае, если сервер не был запущен

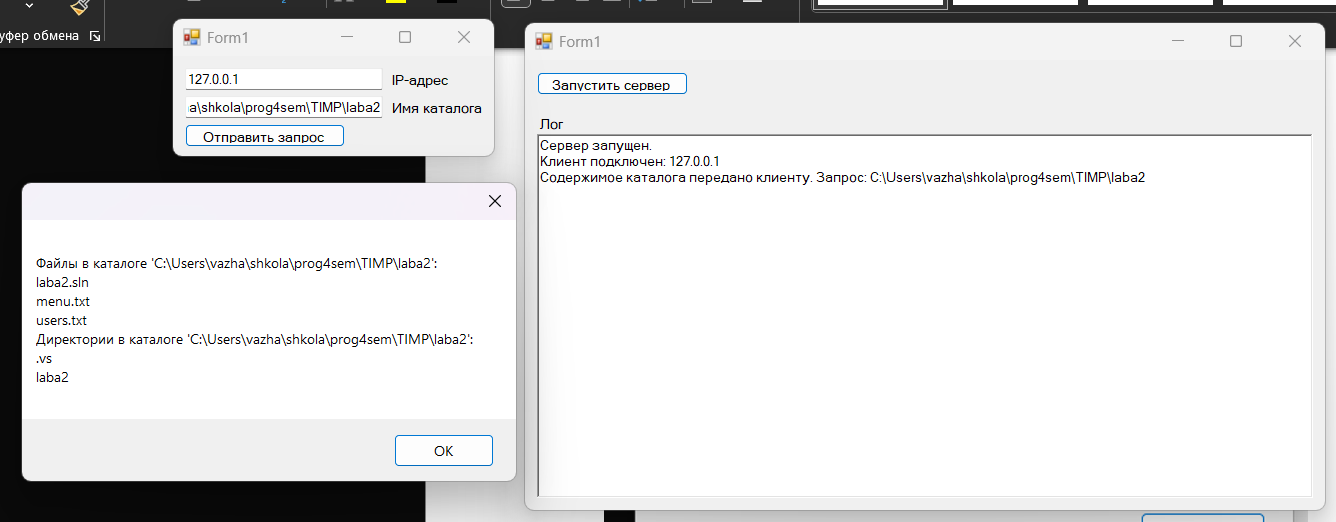


Рисунок 3 - Обработка запроса на просмотр содержимого каталога

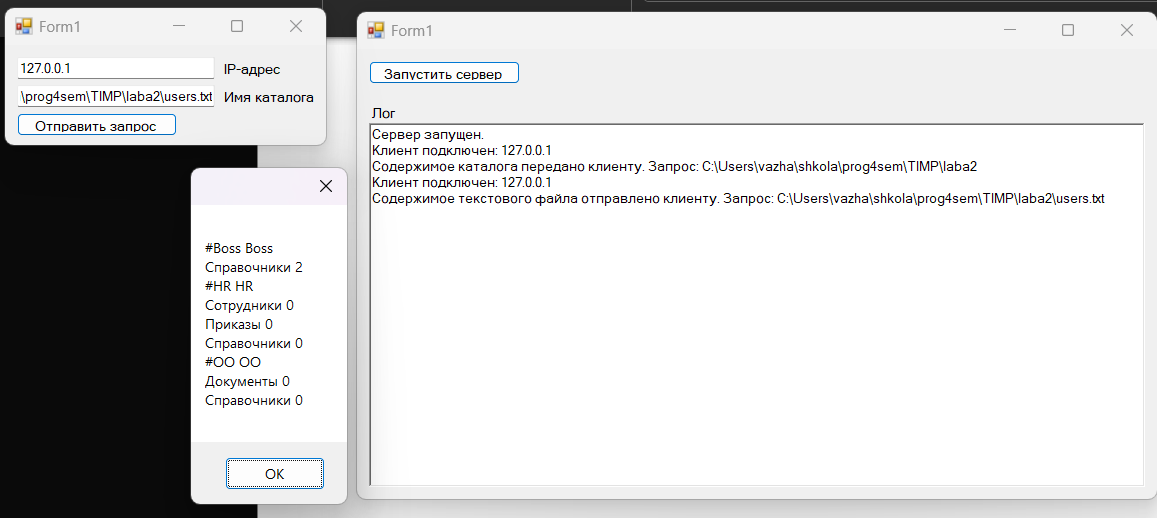


Рисунок 4 - Обработка запроса на просмотр содержимого текстового файла

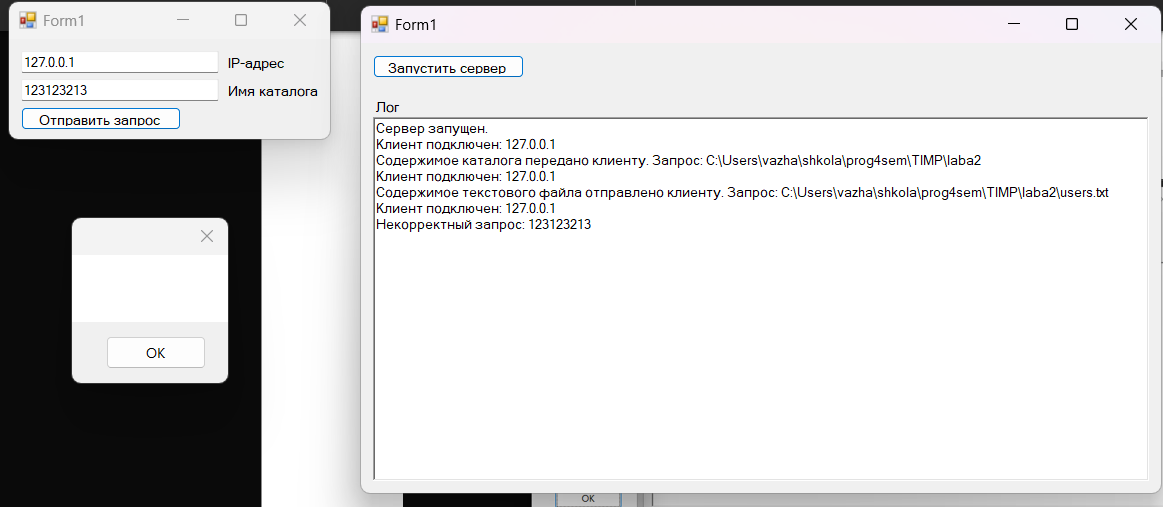


Рисунок 5 - Обработка некорректного запроса