

Classes

UDP_Client_Static

This is a static class which supports sending data to a remote ip and port using a Socket.

Methods

```
Send(IPAddress ip, int port, string message)
```

Creates a temporary socket and sends the message argument.

```
SendFile(IPAddress ip, int port, string filePath)
```

Creates a temporary socket and sends a header containing file information followed by the file data itself.

```
RecieveMessage(Socket soc, IPEndPoint endPoint)
```

Waits for an incoming message from the passed socket.

UDP_Server_Static

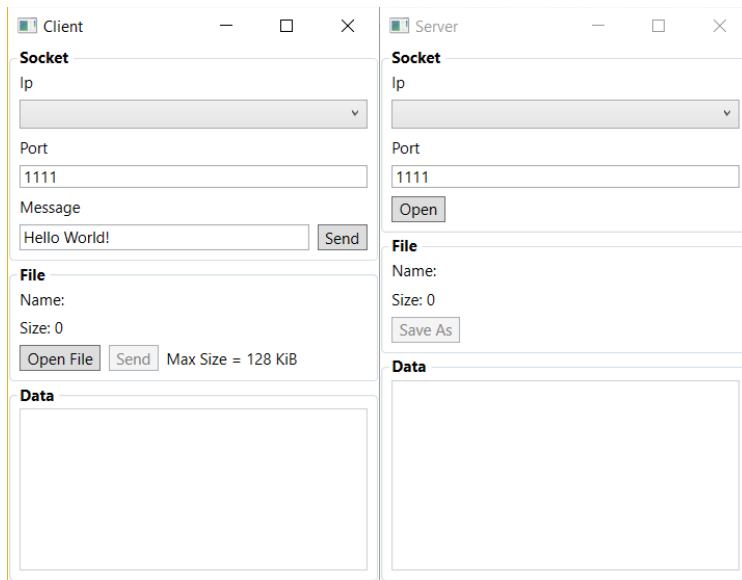
```
OpenSocket(IPAddress ip, int port)
```

Creates a semi temporary (lifetime of the method) socket, binds it to the given ip and port, and waits for incoming data.

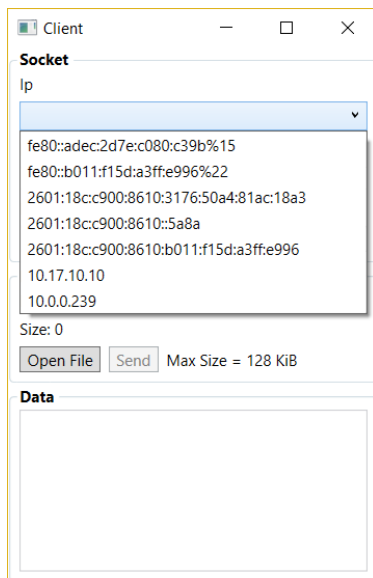
Note ** The rest of the files (MainWindow.xaml, MainWindow.xaml.cs) are UI exclusively and do not interact with the Sockets other than to call UDP_Client_Static and UDP_Server_Static. They do however server to load and save files, display information, and find all available hardware ips (both ipv4 and ipv6).

Run Procedure

1. Launch both Client.exe and Server.exe



2. Select an IP for the Client



3. Select the same IP for the server or leave the field blank to listen on all IPs.

- Click Open in the server to start the server socket.

The screenshot shows the 'Server' application window. It has three main sections: 'Socket', 'File', and 'Data'. The 'Socket' section contains a dropdown menu for 'Ip' and a text input for 'Port' with the value '1111'. Below these is an 'Open' button. The 'File' section has labels for 'Name:' and 'Size: 0', and a 'Save As' button. The 'Data' section is a text area containing the message 'Opened 0.0.0.0:1111'.

- To send a string, simply type your message into the Message field in the Client and click the Send button. You should see the message appear on the server side and the client should receive an echo response.

This block contains two side-by-side screenshots of the application windows. The left window is the 'Client' application, and the right window is the 'Server' application. The 'Client' window has 'Socket' (Ip: 10.17.10.10, Port: 1111), 'File' (Name, Size: 0, Open File, Send, Max Size = 128 KiB), and 'Data' (10.17.10.10:1111 - Echo: Hello World!) sections. The 'Server' window has 'Socket' (Ip, Port: 1111, Open), 'File' (Name, Size: 0, Save As), and 'Data' (Opened 0.0.0.0:1111, [::ffff:10.17.10.10]:65090 - Hello World!) sections.

- To send a file, click the Open File button on the Client and select one of the sample images in the Transfer Files folder (or any file you want to send). Then, if the file is within the size restriction, click the Send button. The server should log the incoming file and it should appear in the File section. You can then save the file if you so choose by clicking the Save As button.

| Client | Server |
|---|--|
| Socket Ip 10.17.10.10 Port 1111 Message Hello World! Send | Socket Ip Port 1111 Open |
| File Name: img1.bmp Size: 62826 Open File Send Max Size = 128 KiB | File Name: img1.bmp Size: 62826 Save As |
| Data 10.17.10.10:1111 - Successfully Recieved: img1.bmp | Data Opened 0.0.0.0:1111 [::ffff:10.17.10.10]:64480 - Incoming File: img1.bmp |