ChatClient

ChatGUI

- thread : Thread
- tcpClient : TcpClient
- stream : NetworkStream
- reader : BinaryReader
- HOSTNAME : STRING
- PORT : INT
- writer : BinaryWriter

+ ChatGui() : void
+ Connect(hostname : string, port : int) : bool
+ Run() : void
+ ProcessServerResponse() : void
+ addMessage(value : string
+ Send(data : Packet) : void
+ nicknameForm() : void

Nickname

- writer : BinaryWriter

+ Nickname(writer : BinaryWriter) : void
+ Send(data : Packet) : void

SimpleServer

Client

+ socket : Socket
+ nickname : String
+ thread : Thread
+ stream : NetworkStream
+ reader : BinaryReader
+ writer : BinaryWriter

+ Start() : void
+ Stop() : void
+ SocketMethod() : void
+ SendText(client : Client, receivedMessage : String) : void
+ Send(data : Packet) : void

SimpleServer

+ semaphoreObject : Semaphore
+ tcpListener : TcpListener
+ clients : Client[0...*]

+ SimpleServer(ipAddress : string, port : int) : void
+ Start() : void
+ Stop() : void
+ SocketMethod(client : Client) : void

Packets

Packets

+ type : PacketType
- memberName

<<enum>> PacketType

+ PacketType : enum

NicknamePacket

+ nickname : String

+ NicknamePacket(nickname : String) : void

SellProductPacket

+ productType : String

+ SellProductPacket(productType : String) : void

ChatMessagePacket

+ message : String

+ ChatMessagePacket(message : String) : void

BuyProductPacket

+ productType : String

+ BuyProductPacket(productType : String) : void

Simple Server

GUI	Client	Server
		◯
		Listen for client(s)
	◯	Create new Client object from connected client(s)
	Create thread for client	Start client(s) thread
◯		Determine packet type
Wait for data from server		Perform correct action for packet
Process the server response (Store bytes and deserialise packet)	Send client nickname and message	◇ [Infinite loop]
Display packet message		[Exit] Stop client(s)
		Stop listening for client(s)
		◎

Phase