

BÁO CÁO THỰC HÀNH

Môn học: Nhập môn mạng máy tính

Buổi báo cáo: Lab 03

Tên chủ đề: Lập trình Socket trong C#

GVHD: Nguyễn Xuân Hà

Ngày thực hiện: 02/05/2023

THÔNG TIN CHUNG:

Lớp: NT106.O23

STT	Họ và tên	MSSV	Email
1	Phạm Nhật Tân	22521311	22521311@gm.uit.edu.vn

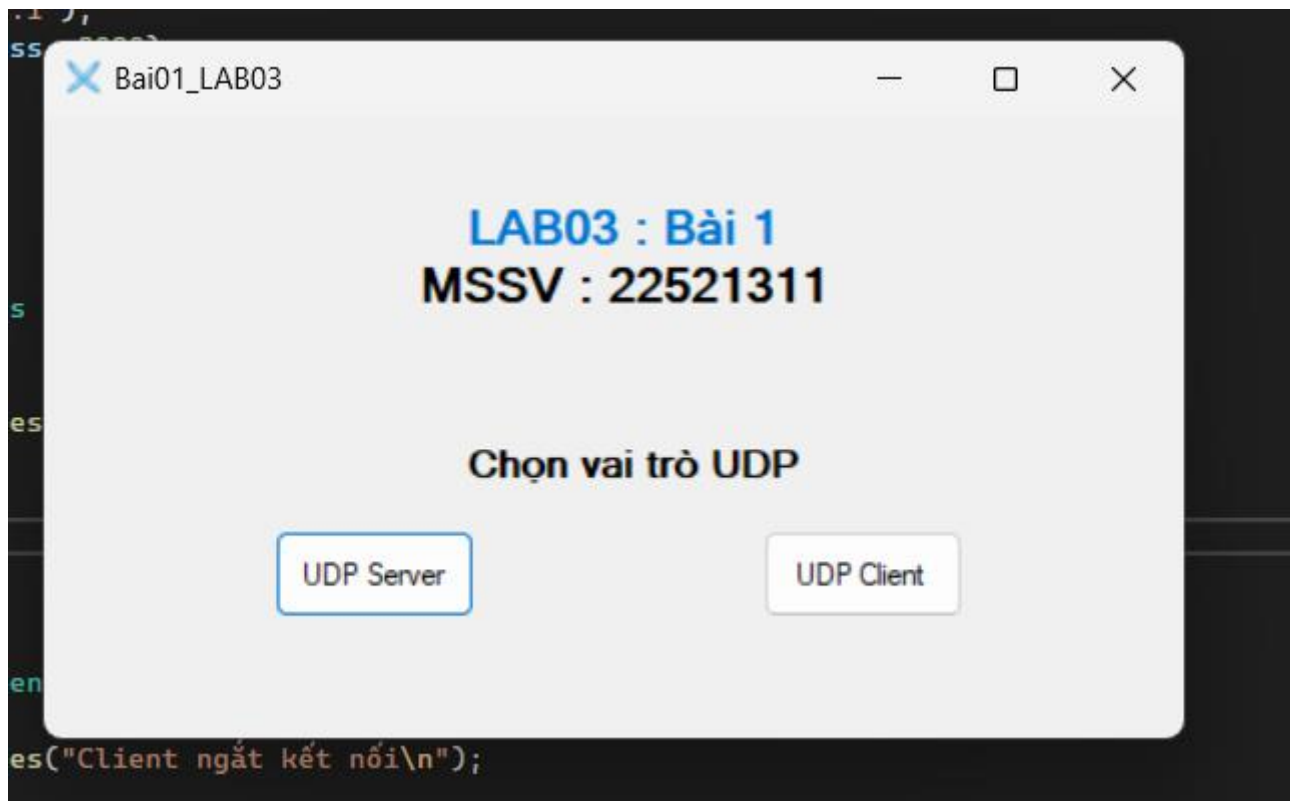
1. ĐÁNH GIÁ KHÁC:

Nội dung	Kết quả
Tổng thời gian thực hiện bài thực hành trung bình	14 ngày
Link Video thực hiện (nếu có)	https://drive.google.com/drive/folders/1qN66YwnMSr7-Rqn3Fgx9fXYzgX0xBfB7?usp=sharing
Ý kiến (nếu có) + Khó khăn + Đề xuất ...	<ul style="list-style-type: none">- Khó khăn : phần 1 Client – Multiserver khá phức tạp, chưa thể thực hiện- Đề xuất : thêm thời gian tìm hiểu và cập nhật phần đây sau
Điểm tự đánh giá	7 điểm

Phần bên dưới của báo cáo này là báo cáo chi tiết của nhóm thực hiện.

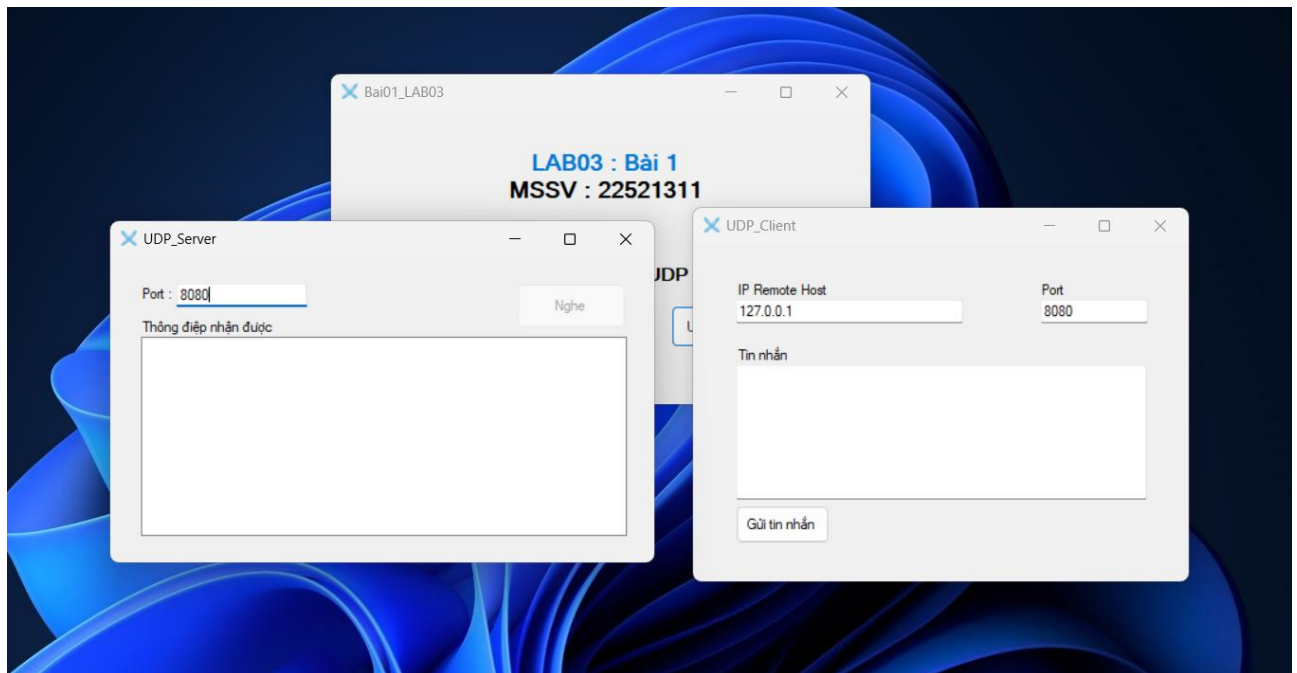
BÁO CÁO CHI TIẾT

1) Bài 01 – Ứng dụng thực hiện gửi và nhận dữ liệu dùng UDP



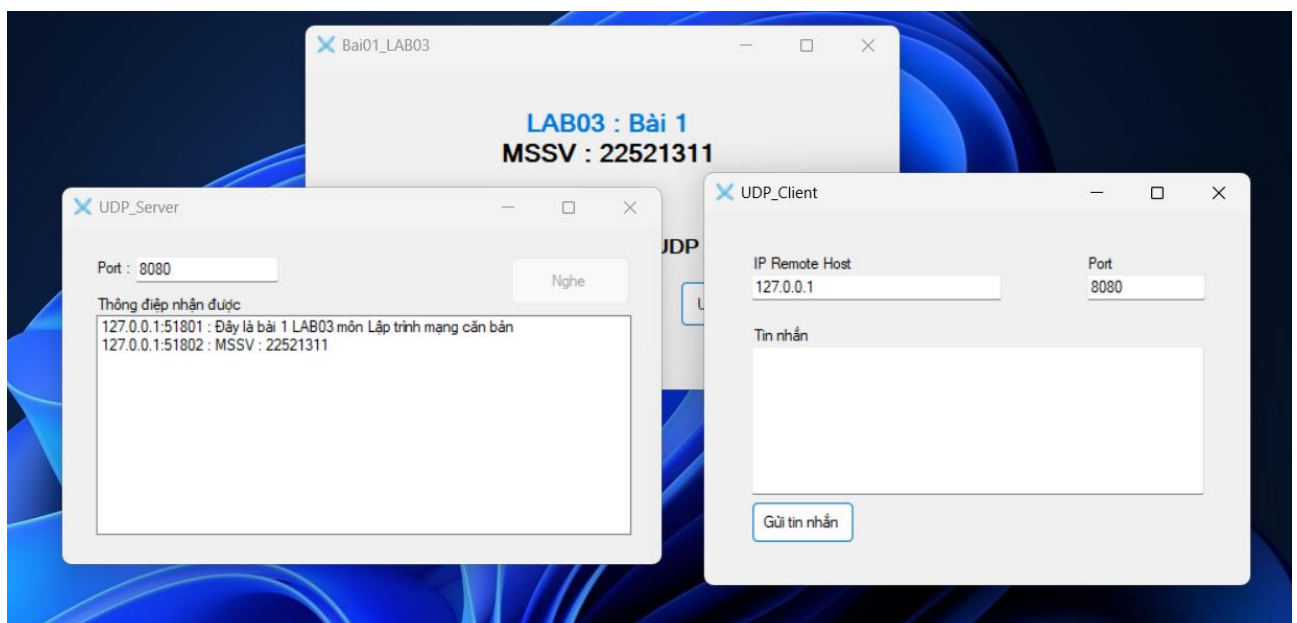
(giao diện bài 1)

Ứng dụng có 2 chức năng : **trở thành UDP Server** và **trở thành UDP Client** để **gửi và nhận dữ liệu** giữa hai bên **sử dụng giao thức UDP**. Khi muốn thực hiện trao đổi thông tin, chạy ứng dụng 2 lần và chọn 2 role khác nhau, điền đầy đủ thông tin yêu cầu.



(chuẩn bị thực hiện trao đổi dữ liệu bằng UDP)

Giao thức UDP là giao thức phi kết nối, không cần thiết lập kết nối giữa hai bên khi tiến hành trao đổi thông tin. Ở bài tập này, ta sẽ sử dụng địa chỉ loopback 127.0.0.1 với port 8080 để thực hiện trao đổi bằng giao thức UDP. Tại bên nhận, khai báo kết nối UDP với số hiệu cổng là 8080; Tại bên gửi khai báo kết nối UDP với số hiệu cổng giống nhau 8080, cùng với địa chỉ thực hiện (127.0.0.1).



(truyền thông tin từ Client đến Server dùng UDP)

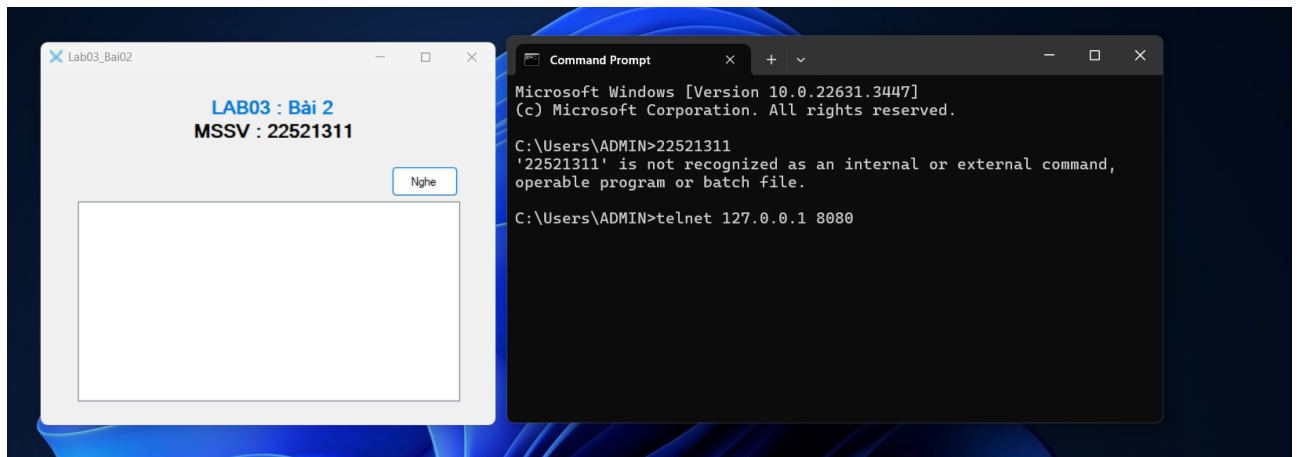
Khi Server kích hoạt sự kiện nghe, nó sẽ listen bất kì địa chỉ IP nào với số cổng đã chỉ định rõ (ở đây là cổng 8080). Khi client nhấn gửi tin nhắn, với số cổng trùng với cổng Server đang lắng nghe, thông tin gửi đi từ Client được Server bắt nhận và hiển thị bên Server.

2) **Bài 02 – Lắng nghe dữ liệu từ dịch vụ Telnet sử dụng kết nối TCP**



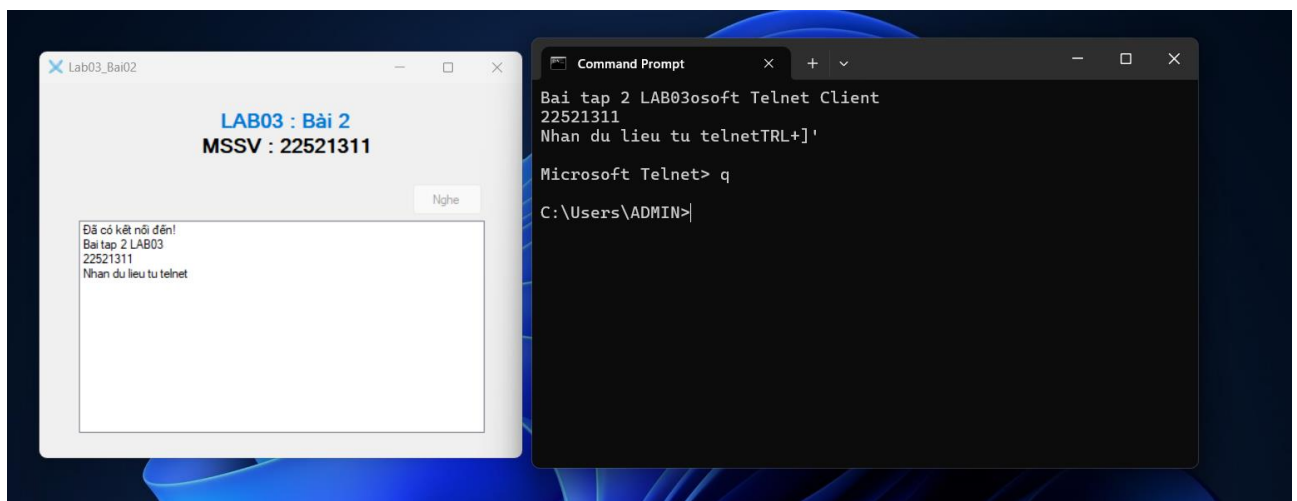
(giao diện bài 2)

Ứng dụng được cài đặt sử dụng kết nối TCP, với **cổng được đặt định là 8080** để lắng nghe dữ liệu từ dịch vụ telnet có sẵn trên Window.



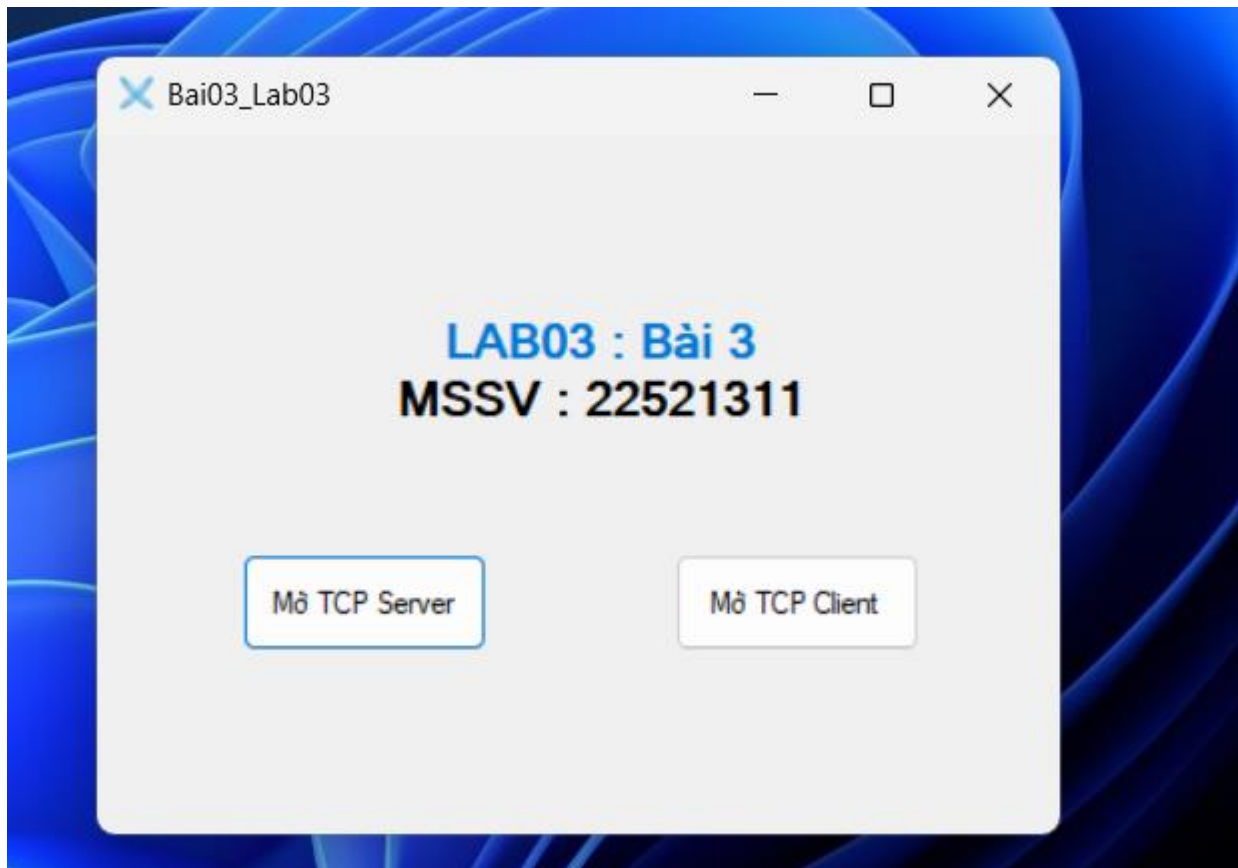
(thực hiện kết nối TCP từ dịch vụ telnet)

Để có thể telnet đến địa chỉ IP của máy tại cổng 8080, ta phải mở port 8080 lắng nghe kết nối TCP đến port 8080, do đó khi nhấn nút Listen có nghĩa là ta đang thực hiện lắng nghe kết nối tại địa chỉ IP của máy và cổng là 8080.



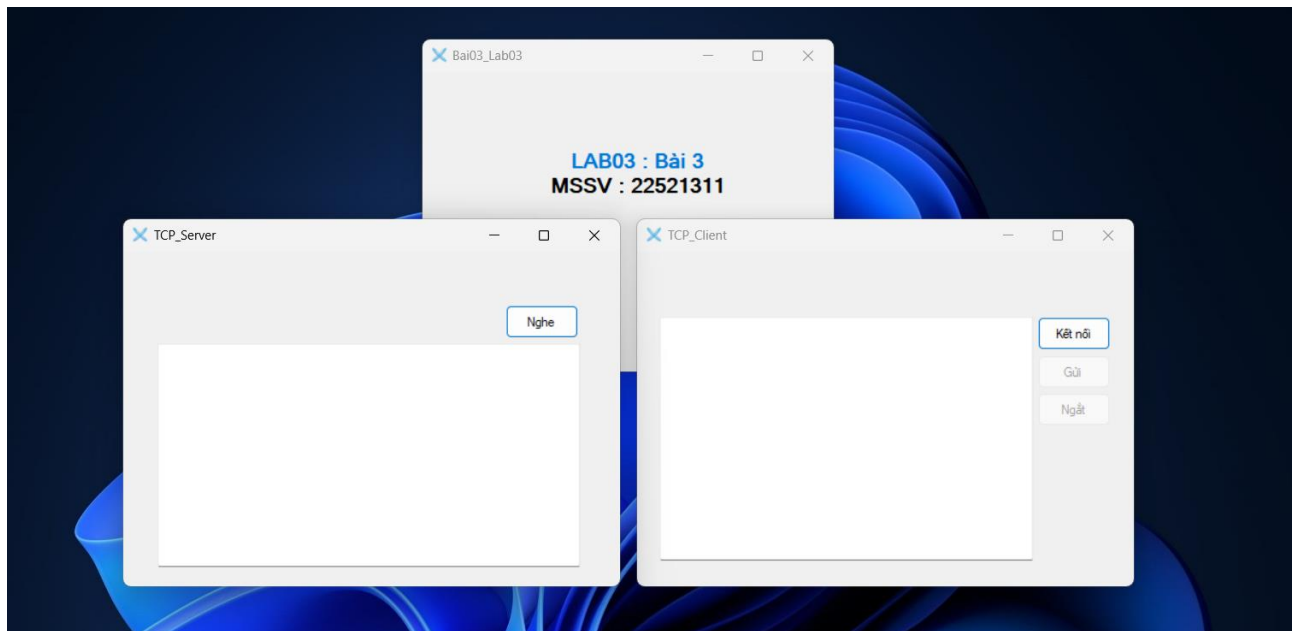
(gửi thông điệp từ telnet và nhận ở ứng dụng thông qua kết nối TCP)

3) **Bài 03 - Ứng dụng thực hiện gửi và nhận dữ liệu dùng TCP**



(giao diện bài 3)

Ứng dụng có 2 chức năng : **trở thành TCP Server** và **trở thành TCP Client** để **gửi và nhận dữ liệu** giữa hai bên **sử dụng giao thức TDP**. Khi muốn thực hiện trao đổi thông tin, chạy ứng dụng 2 lần và chọn 2 role khác nhau, điền đầy đủ thông tin yêu cầu.



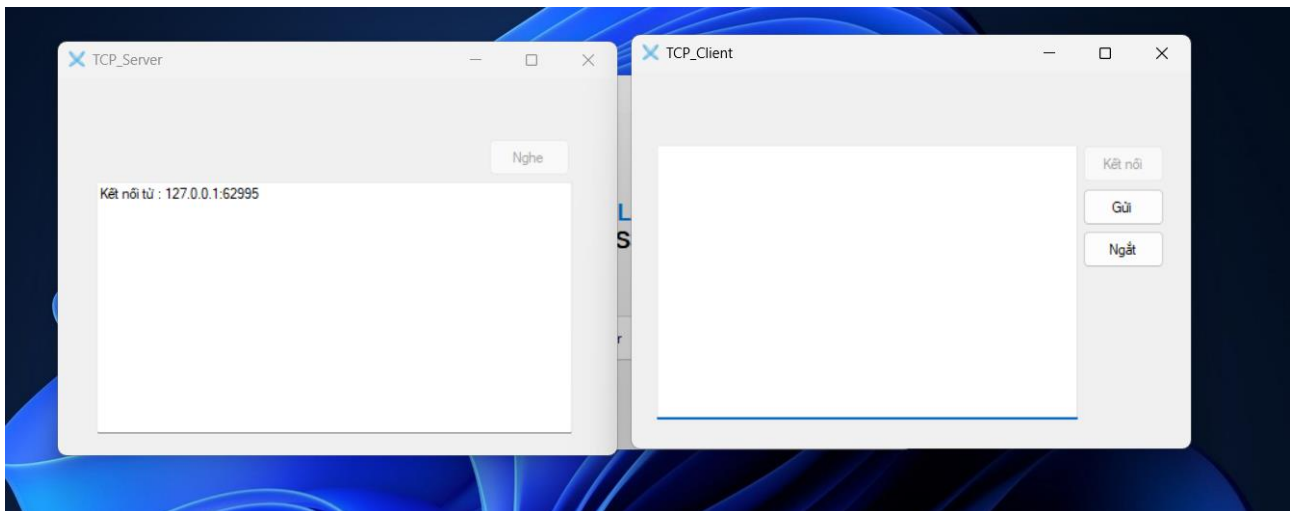
(chuẩn bị thực hiện trao đổi dữ liệu bằng UDP)

Mô tả Server :

- Tạo một đối tượng `System.Net.Sockets.TcpListener` để bắt đầu “lắng nghe” trên một cổng cục bộ.
- Đợi và chấp nhận kết nối từ client với phương thức `AcceptSocket()`. Phương thức này trả về một đối tượng `System.Net.Sockets.Socket` dùng để gửi và nhận dữ liệu.
- Thực hiện giao tiếp với client.
- Đóng Socket.

Mô tả Client :

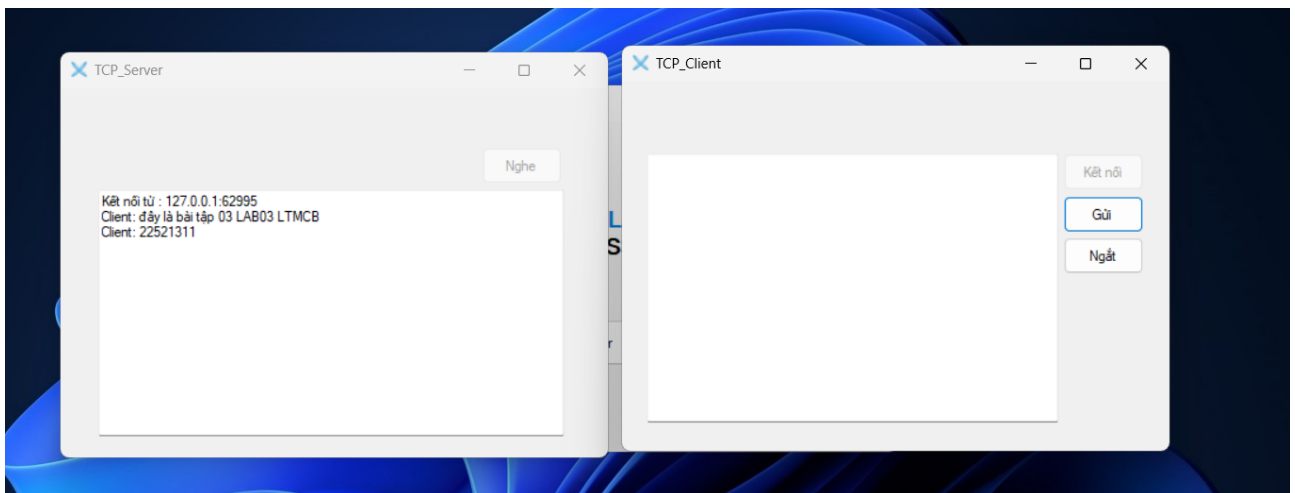
- Tạo một đối tượng `System.Net.Sockets.TcpClient`
- Kết nối đến server với địa chỉ và port xác định với phương thức `TcpClient.Connect()`.
- Lấy luồng (stream) giao tiếp bằng phương thức `TcpClient.GetStream()`.
- Thực hiện giao tiếp với server.
- Đóng luồng và socket.



(thực hiện kết nối thành công từ Client đến Server)

Ở Server : Tạo socket bên nhận, socket này là socket lắng nghe các kết nối tới nó tại địa chỉ IP của máy và port 8080. Sau khi khởi tạo, bắt đầu lắng nghe và đợi tín hiệu để đồng ý kết nối

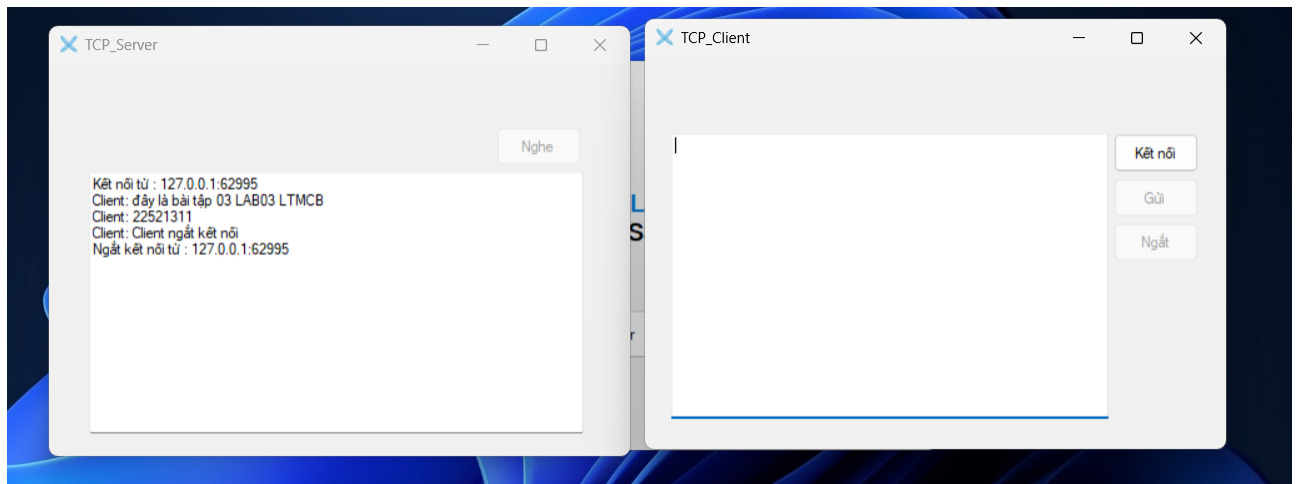
Ở Client : Tạo đối tượng TcpClient và kết nối đến Server với 1 địa chỉ Ip và Port xác định. Nếu Server đang trong trạng thái lắng nghe, nó sẽ đồng ý và thiết lập kết nối.



(thực hiện kết gửi thông tin từ Client đến Server)

Ở Client : Tạo luồng để đọc và ghi dữ liệu dựa trên NetworkStream. Lấy luồng (stream) giao tiếp bằng phương thức TcpClient.GetStream(). Dùng phương thức Write để gửi dữ liệu đến Server.

Ở Server : Dùng một vòng lặp để thực hiện việc nhận dữ liệu, với điều kiện vòng lặp là trạng thái Socket vẫn còn đang là kết nối.



(thực hiện kết gửi thông tin từ Client đến Server)

Ở Client : Khi muốn dừng kết nối, dùng phương thức Write để gửi dữ liệu mang dấu hiệu kết thúc cho Server biết và đóng kết nối. Sau đó thực hiện đó NetworkStream và TcpClient.

Ở Server : Nhận được thông điệp đóng kết nối của Client, chờ 1 khoảng thời gian rồi sau đó mới thực hiện việc đóng kết nối.

YÊU CẦU CHUNG

1) Đánh giá

- Chuẩn bị tốt các yêu cầu đặt ra trong bài thực hành.
- Sinh viên hiểu và tự thực hiện được bài thực hành, trả lời đầy đủ các yêu cầu đặt ra.
- Nộp báo cáo kết quả chi tiết những đã thực hiện, quan sát thấy và kèm ảnh chụp màn hình kết quả (nếu có); giải thích cho quan sát (nếu có).
- Sinh viên báo cáo kết quả thực hiện và nộp bài.

2) Báo cáo

- File .PDF hoặc .docx. Tập trung vào nội dung, giải thích.
- Nội dung trình bày bằng Font chữ Times New Romans/ hoặc font chữ của mẫu báo cáo này (UTM Avo)– cỡ chữ 13. Canh đều (Justify) cho văn bản. Canh giữa (Center) cho ảnh chụp.
- Đặt tên theo định dạng: LabX_MSSV1_MSSV2. (trong đó X là Thứ tự buổi Thực hành).
Ví dụ: Lab01_21520001_21520002
- Nộp file báo cáo trên theo thời gian đã thống nhất tại courses.uit.edu.vn.

Bài sao chép, trễ, ... sẽ được xử lý tùy mức độ vi phạm.

HẾT