



BÀI TẬP LỚN HỌC PHẦN: CÁC HỆ PHÂN TÁN

Đề tài 01:
MySSH

Giảng viên hướng dẫn: TS. Nguyễn Bình Minh

Nhóm sinh viên thực hiện:

Vũ Mạnh Kiểm	CNTT-TT 2.2 – K56	2011 1731
Đỗ Đăng Hiền	CNTT-TT 2.3 – K56	2011 1581
Nguyễn Huy Hùng	CNTT-TT 2.4 – K56	2011 1520
Thlok Pisey	CNTT 2 – K55	2010 2793

Mã lớp: 79247 – Mã học phần: IT4371

TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI
VIỆN CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG

BÀI TẬP LỚN
HỌC PHẦN: CÁC HỆ PHÂN TÁN

Đề tài 01:
MySSH

Sinh viên thực hiện:	Vũ Mạnh Kiêm	2011 1731
	Đỗ Đăng Hiên	2011 1581
	Nguyễn Huy Hùng	2011 1520
	Thlok Pisey	2010 2793

Giảng viên hướng dẫn: TS. Nguyễn Bình Minh

LỜI NÓI ĐẦU

Có thể nói công nghệ thông tin là một ngành còn non trẻ so với nhiều lĩnh vực khác của đời sống xã hội loài người. Công nghệ thông tin bắt đầu phát triển mạnh mẽ từ năm 1996 nhưng sự phát triển của nó là theo cấp số nhân. Thực sự sức mạnh của công nghệ thông tin là không gì có thể ngăn cản. Nó đi sâu và trợ giúp các lĩnh vực khác trong đời sống xã hội. Và theo như tác giả của cuốn sách “Thế giới phẳng” - Thomas Friedman thì công nghệ thông tin là nhân tố tạo nên thế giới phẳng 3.0. Vì vậy việc học tập, nghiên cứu và phát triển trong lĩnh vực công nghệ thông tin là một nhu cầu, một nhiệm vụ và một hướng đi tất yếu của giới trẻ. Nó không chỉ giúp cho các bạn phát triển bản thân mà còn giúp cho đất nước tiến bộ và phát triển hơn.

Các hệ thống phân tán là một nội dung quan trọng trong lĩnh vực công nghệ thông tin, ngày nay các hệ thống luôn cần phục vụ một số lượng người dùng đông đảo và trải rộng ở các khu vực địa lý khác nhau. Với những hệ thống phục vụ cộng đồng (ví dụ Google, Facebook ...) thì người dùng luôn yêu cầu hệ thống phải đủ nhanh để không phải chờ đợi nhiều. Mỗi hệ thống càng lớn, càng mở rộng thì yêu cầu phân tán càng quan trọng. Làm chủ được các công nghệ cốt lõi của hệ thống phân tán sẽ giúp cho sinh viên có nền tảng tư duy chỉ động hơn trong việc áp dụng với các hệ thống thực tế.

Môn học các hệ phân tán của viện công nghệ thông tin và truyền thông đã cung cấp cho sinh viên những kiến thức cơ bản nhưng cũng rất cập nhật về các công nghệ phân tán, giúp sinh viên phần nào có được nền tảng cơ bản của lĩnh vực nghiên cứu thú vị này. Rất khó để có thể nắm vững được toàn bộ các kỹ thuật cũng như công nghệ phân tán hiện tại nhưng qua nội dung môn học, qua bài tập lớn, nhóm chúng em cũng đã có được nhiều trải nghiệm cũng như kiến thức quý báu để làm tăng thêm vốn hiểu biết và kinh nghiệm của mình. Trong các bài tập lớn của môn học, nhóm chúng em đã chọn đề tài số 01: MySSH với nền tảng là lập trình Socket. Là nhóm duy nhất lựa chọn đề tài này, nhóm em đã cùng nhau là việc và cố gắng hoàn thành ở mức độ tốt nhất có thể. Hy vọng những gì nhóm làm được đã đáp ứng được các yêu cầu và hy vọng nhận được sự đánh giá tích cực cũng như những lời góp ý chân thành từ thầy Bình Minh và các bạn trong lớp.

Hà Nội, tháng 05 năm 2015

Nhóm sinh viên thực hiện đề tài

BẢNG PHÂN CÔNG NHIỆM VỤ

Họ và tên	Công việc thực hiện
Vũ Mạnh Kiểm (nhóm trưởng)	<ul style="list-style-type: none">- Lên ý tưởng.- Code Server là Client.- Chỉnh sửa nội dung báo cáo.
Đỗ Đăng Hiển	<ul style="list-style-type: none">- Lên ý tưởng.- Hỗ trợ code Server.- Làm Slide.
Nguyễn Huy Hùng	<ul style="list-style-type: none">- Lên ý tưởng.- Hỗ trợ code Client.- Làm báo cáo.
Thlok Pisey	<ul style="list-style-type: none">- Test hệ thống và phát hiện lỗi.

MỤC LỤC

LỜI NÓI ĐẦU	1
BẢNG PHÂN CÔNG NHIỆM VỤ	2
MỤC LỤC	3
PHẦN 1 – MÔ TẢ BÀI TOÁN	5
1.1 BÀI TOÁN	5
a. Giới thiệu bài toán	5
b. Phân tích các khái niệm.....	5
1.2 PHƯƠNG HƯỚNG GIẢI QUYẾT VẤN ĐỀ.....	7
a. Môi trường hoạt động.....	7
b. Ngôn ngữ lập trình.....	7
c. Đáp ứng về yêu cầu chức năng.....	7
PHẦN 2 – CÀI ĐẶT	9
2.1 Chi tiết giải pháp và kỹ thuật Code để xử lý các yêu cầu.....	9
a. Config đối với server.....	9
b. Lập trình Socket.	9
c. Xử lý đa luồng của server.....	9
d. Giới hạn số lượng đăng nhập và phát hiện 2 user đăng nhập cùng một tài khoản.	10
e. Server nhận biết các yêu cầu khác nhau thế nào?	11
f. Xem các file và thư mục trong thư mục hiện hành.	11
g. Di chuyển file và thư mục.	11
h. Xóa file và thư mục.	11
i. Xem thời gian của hệ thống.....	12
j. Tải file về từ server.....	12
k. Upload file lên server.	12
2.2 Hướng dẫn mở project với visual studio 2008.....	13
2.3 Hướng dẫn cài đặt và thực thi chương trình đã được build ra file.....	14

KẾT LUẬN VÀ LỜI CẢM ƠN.....	21
TÀI LIỆU THAM KHẢO.....	22

PHẦN 1

MÔ TẢ BÀI TOÁN

1.1 BÀI TOÁN

a. Giới thiệu bài toán

Viết ứng dụng myssh mô phỏng chương trình “ssh” để Client kết nối từ xa đến Server thực hiện các lệnh hệ thống trên Server.

Yêu cầu:

- Dùng socket, giao thức TCP.
- Giao diện dòng lệnh
- Multiclients: cho phép nhiều clients kết nối đồng thời (giới hạn N clients, vượt quá N thì không cho phép kết nối)
- Khi client kết nối tới phải đăng nhập (username / pass). CSDL người dùng là của chương trình, không phải của hệ thống, cơ chế lưu tùy ý (CSDL, text, v.v)
 - + Server thông báo kết nối thành công hay không
- Khi kết nối thành công thì client có thể thực hiện các lệnh trên hệ thống của server
 - + Các lệnh có thể có tham số hoặc không
 - + Một số lệnh cơ bản: hiển thị thư mục hiện thời, in danh sách các tệp, thư mục trong thư mục hiện thời, xóa, di chuyển tệp, thư mục, hiển thị ngày giờ hệ thống, v.v
- Server phân tích lệnh nhận được từ phía client, thực hiện và trả kết quả cho client
 - + Nếu các lệnh thực hiện không được phép chạy thì phải thông báo lỗi.
 - + Lệnh chạy trên server phụ thuộc vào hệ điều hành mà server được cài đặt lên (Windows, Linux)
- Kết thúc phiên làm việc client gửi lệnh ngắt kết nối tới server và chấm dứt kết nối.
- Yêu cầu đặc biệt: Có lệnh get cho phép tải về 1 tệp dạng text từ Server về Client (nếu tệp lớn, có thể ngắt nhỏ ra để gửi).

b. Phân tích các khái niệm

- **Giới thiệu về chương trình SSH:**
SSH là gì?

SSH là một chương trình tương tác giữa máy chủ và máy khách có sử dụng cơ chế mã hoá đủ mạnh nhằm ngăn chặn các hiện tượng nghe trộm, đánh cắp thông tin trên đường truyền. Các chương trình trước đây: telnet, rlogin không sử dụng phương pháp mã hoá. Vì thế bất cứ ai cũng có thể nghe trộm thậm chí đọc được

toàn bộ nội dung của phiên làm việc bằng cách sử dụng một số công cụ đơn giản. Sử dụng SSH là biện pháp hữu hiệu bảo mật dữ liệu trên đường truyền từ hệ thống này đến hệ thống khác.

Cách thức làm việc của SSH

SSH làm việc thông qua 3 bước đơn giản:

- Định danh host - xác định định danh của hệ thống tham gia phiên làm việc SSH.
- Mã hoá - thiết lập kênh làm việc mã hoá.
- Chứng thực - xác thực người sử dụng có quyền đăng nhập hệ thống.

- Lập trình Socket

Giới thiệu Socket

- Socket là một công logic mà một chương trình sử dụng để kết nối với một chương trình khác chạy trên một máy tính khác trên Internet. Chương trình mạng có thể sử dụng nhiều Socket cùng một lúc, nhờ đó nhiều chương trình có thể sử dụng Internet cùng một lúc.
- Hiểu đơn giản, Socket là tập hợp của IP + Port. Nếu coi một khu tập thể là một máy tính thì địa chỉ của khu tập thể là Ip của máy tính đó, số phòng của một căn hộ trong đó là Port. Nếu có địa chỉ + số phòng, ta sẽ tới được chính xác căn hộ đó để thực hiện được công việc mình muốn.

Có 2 loại Socket:

- Stream Socket: Dựa trên giao thức TCP(Transmission Control Protocol) việc truyền dữ liệu chỉ thực hiện giữa 2 quá trình đã thiết lập kết nối. Giao thức này đảm bảo dữ liệu được truyền đến nơi nhận một cách đáng tin cậy, đúng thứ tự nhờ vào cơ chế quản lý luồng lưu thông trên mạng và cơ chế chống tắc nghẽn.
- Datagram Socket: Dựa trên giao thức UDP(User Datagram Protocol) việc truyền dữ liệu không yêu cầu có sự thiết lập kết nối giữa 2 quá trình. Ngược lại với giao thức TCP thì dữ liệu được truyền theo giao thức UDP không được tin cậy, có thể không đúng trình tự và lặp lại. Tuy nhiên vì nó không yêu cầu thiết lập kết nối không phải có những cơ chế phức tạp nên tốc độ nhanh, ứng dụng cho các ứng dụng truyền dữ liệu nhanh như chat, game.

Port

- Port xác định duy nhất một tiến trình trên một máy trong mạng. Nói cách khác là cách mà phân biệt giữa các ứng dụng.

Ví dụ, khi máy tính chạy nhiều ứng dụng mạng như Yahoo, Firefox, game online, giả sử chương trình Yahoo sử dụng port 5150 thì khi ai đó gửi tin nhắn đến cho người dùng, lúc tin nhắn đến máy nó sẽ dựa vào port để nhận biết đó là

chương trình Yahoo (port 5150) chứ không phải là chương trình khác. Sau đó thông tin sẽ được xử lý và hiển thị tin nhắn lên.

Một TCP/IP Socket gồm một địa chỉ IP kết hợp với một port xác định duy nhất một tiến trình trên mạng. Hay nói cách khác luồng thông tin trên mạng dựa vào IP là để xác định máy một máy trên mạng còn port xác định 1 tiến trình trên 1 máy.

- **Ứng dụng Client – Server**

- Trước tới giờ, việc lập trình thường với mục đích là tạo ra được một ứng dụng. Nhưng ứng dụng đó chỉ hoạt động độc lập 1 mình riêng lẻ. Mục tiêu của bài tập lớn là sẽ đưa ra những ứng dụng dạng Client – Server. Tức là sẽ có 2 loại ứng dụng chính đó là Client và Server.
- Quy trình hoạt động của ứng dụng Server – Client như sau: Server có nhiệm vụ của là lắng nghe, chờ đợi kết nối từ Client trên địa chỉ IP của mình với PORT được quy định sẵn. Khi client gửi dữ liệu tới Server thì nó phải giải quyết một công việc là nhận dữ liệu đó → xử lý → trả kết quả lại cho Client.
- Client là ứng dụng được phục vụ, nó chỉ gửi truy vấn và chờ đợi kết quả từ Server.

1.2 PHƯƠNG HƯỚNG GIẢI QUYẾT VẤN ĐỀ

Cách giải quyết như sau:

a. Môi trường hoạt động

Nhóm sử dụng môi trường hoạt động của cả Client và Server đều là hệ điều hành Windows.

Lý do:

- Các thành viên trong nhóm có kỹ năng tốt trong việc lập trình với windows API trên môi trường windows.
- Đa số các thành viên đều không sử dụng thành thạo môi trường Linux hay các môi trường khác.

b. Ngôn ngữ lập trình

Nhóm sử dụng ngôn ngữ lập trình C++ với sự hỗ trợ của windows API.

Lý do:

- C++ là ngôn ngữ lập trình hệ thống chuẩn, các thành viên trong nhóm đều am hiểu và có thể sử dụng.

c. Đáp ứng về yêu cầu chức năng.

- Multiclients:

- Để thực hiện yêu cầu này, server cần sử dụng nhiều luồng khác nhau để phục vụ các Client khác nhau đồng thời cần có một luồng chính lắng nghe các kết nối tới.
- Phương án được sử dụng là tạo Thread riêng cho mỗi Client.
- Đăng nhập và cơ sở dữ liệu:
 - Nhóm sử dụng CSDL ở dạng file text, Client không thể thực hiện bất kỳ yêu cầu nào nếu chưa tiến hành đăng nhập.
- Các lệnh cơ bản trên server:
 - Nhóm đã cài đặt các yêu cầu cơ bản của bài toán:
 - ✓ Đăng nhập vào hệ thống.
 - ✓ Xem danh sách các file và thư mục trong thư mục hiện tại.
 - ✓ Di chuyển file, thư mục.
 - ✓ Xóa file, thư mục.
 - ✓ Hiện thị ngày giờ của hệ thống (theo 2 lựa chọn theo giờ địa phương hoặc theo giờ UTC).
 - ✓ Nếu có lỗi không thực hiện được, server sẽ báo lại cho Client.
- Yêu cầu đặc biệt:
 - Nhóm đã đáp ứng được yêu cầu cho phép client tải file về từ server.
- Làm thêm:
 - Nhóm đã mở rộng 2 chức năng cho bài toán:
 - ✓ Nếu có 2 người trở lên cùng đăng nhập 1 tài khoản, người đăng nhập sau sẽ được thông báo có người đã đang đăng nhập tài khoản đó.
 - ✓ Cho phép Client up file lên server.

PHẦN 2

CÀI ĐẶT

2.1 Chi tiết giải pháp và kỹ thuật Code để xử lý các yêu cầu.

a. Config đối với server.

Cấu hình config của server gồm có:

- Số lượng Client tối đa mà server phục vụ.
- Số lượng User của hệ thống.
- Danh sách User và password tương ứng.

Các thông tin trên được lưu trong file config.ini. Khi server được chạy, nó sẽ đọc file config này đầu tiên và thiết lập các thông số trên cho hệ thống.

Chú ý: có một vấn đề mà nhóm đã thảo luận nhưng chưa xử lý ngay đó là việc mã hóa password trên file config. Nhóm đã có ý định sử dụng bảng thế để mã hóa nhưng do không tập chung nhiều vào chức năng đăng nhập và server chưa cần yêu cầu bảo mật cao nên nhóm đã không tiến hành cài đặt nội dung này.

b. Lập trình Socket.

Trên môi trường windows, nhóm sử dụng thư viện winsoc.h để lập trình socket. Thư viện này khá dễ dàng sử dụng và có khá nhiều tài liệu hướng dẫn về nó.

Để tra chính thức thì có thể nên dùng [MSDN](#) nhưng với thư viện này có một bài hướng dẫn rất hay và chi tiết mà chúng ta cũng nên tham khảo của anh Kevin Hoang, người quản trị của diễn đàn Cộng đồng C Việt: [Link](#).

c. Xử lý đa luồng của server.

Server sử dụng một luồng chính để lắng nghe các kết nối tới từ Client. Các server sẽ sử dụng cổng 1080 để lắng nghe các kết nối tới.

Khi có một kết nối từ Client tới, server sẽ đồng ý vào tạo ra một socket khác để giao tiếp với Client đó. Chương trình sử dụng hàm [CrealThread\(\)](#) trong Windows API để tạo ra một luồng mới dành cho việc xử lý yêu cầu từ Client này.

Như vậy luồng chính là luồng luôn luôn lắng nghe kết nối. Có bao nhiêu Client đang kết nối thì sẽ có tương ứng bấy nhiêu luồng phục vụ.

Chú ý: Mỗi luồng tạo ra là độc lập về mặt dữ liệu và không thể can thiệp vào dữ liệu của nhau do đó không thể có sự nhầm lẫn về các lệnh của các client khác nhau.

d. Giới hạn số lượng đăng nhập và phát hiện 2 user đăng nhập cùng một tài khoản.

- Số lượng đăng nhập tối đa được qui định trong file config.ini. Khi số lượng đã đạt tới giới hạn này, khi client mới kết nối tới, server vẫn sẽ chấp nhận kết nối và tạo thêm một luồng nữa. Tuy nhiên luồng này chỉ có nhiệm vụ để thông báo cho Client biết rằng server vẫn đang hoạt động nhưng đã có nhiều người truy cập, client có thể chờ để truy cập sau. Sau khi gửi thông báo đến client, server sẽ ngắt kết nối và giải phóng luồng đã cấp phát đó.

Nếu server không chịu chấp nhận kết nối và báo cho client thông tin. Có thể Client sẽ thông báo không kết nối được và nghĩ rằng server đã ngừng hoạt động do một sự cố nào đó. Đó sẽ là một tình huống phát sinh nhiều vấn đề và suy nghĩ tiêu cực nơi client mà các hệ thống phân tán cần tránh.

- Hệ thống sử dụng một cấu trúc để lưu thông tin các user:
 - ✓ Tên user
 - ✓ Password
 - ✓ User đã có đang đăng nhập không.

Nếu một User thực hiện đăng nhập với tên user vào password, hệ thống sẽ kiểm tra tài khoản này có đang đăng nhập hay không dựa vào thông số thứ 3 ở trên. Khi phát hiện User này đang đăng nhập, server sẽ thông báo cho người đang muốn đăng nhập biết điều này và cho phép người thứ 2 này đăng nhập.

Chú ý: Do server sử dụng một số biến toàn cục cần khai báo kích thước trước nên chương trình sử dụng một define max_user là 10. Do đó để đảm bảo chương trình không bị crash hệ thống sẽ không đồng ý và báo lỗi nếu server config số lượng tối đa truy cập lớn hơn 10. Đây có thể là một nhược điểm cần phải khắc phục của chương trình trong hệ thống thực tế đối với tính khả mở.

e. Server nhận biết các yêu cầu khác nhau thế nào?

Server nhận biết các yêu cầu của client dựa vào cấu trúc lệnh qui định chung giữa client và server. Phương pháp được nhóm để nhận biết cấu trúc đó là tìm xâu. Ví dụ: client muốn đăng nhập thì dùng lệnh:

DangNhap “UserName” “password”

Bên phía server sẽ kiểm tra nội dung của Client gửi sang. Xâu đầu tiên là DangNhap, chuyển hướng tới chức năng đăng nhập. Tách được Username vào password thì sẽ đem đi so sánh với dữ liệu của server.

Tương tự các chức năng khác cũng được nhận biết và xử lý tương tự.

Chú ý: Đây là phương pháp đơn giản, hiệu quả và dễ cài đặt nhưng cũng ẩn chứa một số rủi ro, việc quản định dạng xâu mà client nhập vào là một vấn đề và xử lý được hết các trường hợp nhập nhầm người dùng là khó khăn. Hiện nhóm chưa test được ra lỗi nhưng không đảm bảo chắc chắn không thể có lỗi khi sử dụng.

f. Xem các file và thư mục trong thư mục hiện hành.

Việc này thực chất là đi duyệt thư mục hiện hành, lấy tên file và thư mục rồi trả về cho Client.

Windows API hỗ trợ các hàm để thực hiện công việc này là: [FindFirstFile\(\)](#) và [FindNextFile\(\)](#), với 2 hàm trên, ta có thể lấy được tất cả các file và thư mục.

Tính năng này có thể thực hiện đệ quy để lấy tiếp danh sách các file và thư mục trong thư mục con. Vì mục đích của bài tập lớn không phải là vấn đề này nên nhóm đã không tiến hành cài đặt đệ quy.

g. Di chuyển file và thư mục.

Windows API hỗ trợ hàm [MoveFile\(\)](#) có thể thực hiện di chuyển cả file và thư mục từ nơi này sang nơi khác.

Với yêu cầu bài toán cần tách di chuyển file và thư mục thành 2 lệnh riêng, nhóm đã đưa ra giải pháp là check xem lệnh chuyển file thì đường dẫn đưa sang là file hay thư mục. Chỉ thực hiện lệnh chuyển file với path là File và chuyển thư mục với path là thư mục.

h. Xóa file và thư mục.

Việc xóa file là tương đối đơn giản với lệnh được hỗ trợ là [DeleteFile\(\)](#).

Việc xóa thư mục được thực hiện phức tạp hơn. Môi trường windows không hỗ trợ API nào xóa thư mục có dữ liệu bên trong. Do đó ta sẽ phải đi vào trong và duyệt đệ quy tới từng thư mục để tìm từng file rồi xóa đi. Sau khi xóa hết file ở trong thư mục, ta dùng API [RemoveDirectory\(\)](#) để xóa thư mục.

i. Xem thời gian của hệ thống.

Windows hỗ trợ 2 API để xem thời gian (ngày và giờ) của hệ thống:

[GetLocalTime\(\)](#) – Xem thời gian địa phương của hệ thống.

[GetSystemTime\(\)](#) – Xem thời gian hệ thống theo giờ UTC.

Nhóm đã đưa cả 2 tính năng này vào trong server với 2 tùy chọn khác nhau cho người dùng lựa chọn.

j. Tải file về từ server.

Việc tải file được thực hiện như sau:

- Client yêu cầu tải file với đường dẫn a trên server.
- Server nhận đường dẫn này, kiểm tra xem có file hay không. Nếu không có thì báo cho Client biết không thể lấy file đó. Nếu có, server sẽ gửi về dấu hiệu “StartGetFile “Tên file” để client biết để client tạo ra một file trống với tên file tương ứng.
- Server mở file cần gửi và thực hiện vòng lặp: đọc một khối dữ liệu ra và gửi cho client cho tới khi gửi được hết file đó.
- Client nhận từng nội dung của client gửi và ghi vào file trống mà nó đã tạo ra.
- Khi hoàn thành, server báo cho client biết nó đã gửi hết file đó.
- Client nhận tín hiệu đã gửi xong file thì sẽ đó file đang ghi lại và hoàn tất quá trình. Thông báo quá trình gửi file thành công.

Trong các bước trên nếu có lỗi ở bước nào thì hệ thống sẽ thông báo không thành công.

Chú ý: Trên lý thuyết là phương pháp trên không giới hạn kích thước file sẽ chuyển nhưng thực tế cái đặt nhóm có giới hạn một kích thước nhất định. Lý do cho việc này là để thuận tiện cho cài đặt hơn trong việc sử dụng kiểu dữ liệu (kích thước file lớn nhất có thể chuyển là giới hạn của kiểu dữ liệu DWORD).

k. Upload file lên server.

Việc Upload file lên server được thực hiện ngược lại so với quá trình lấy file từ server xuống. Nó hoàn toàn và thật sự tương tự quá trình trên.

2.2 Hướng dẫn mở project với visual studio 2008

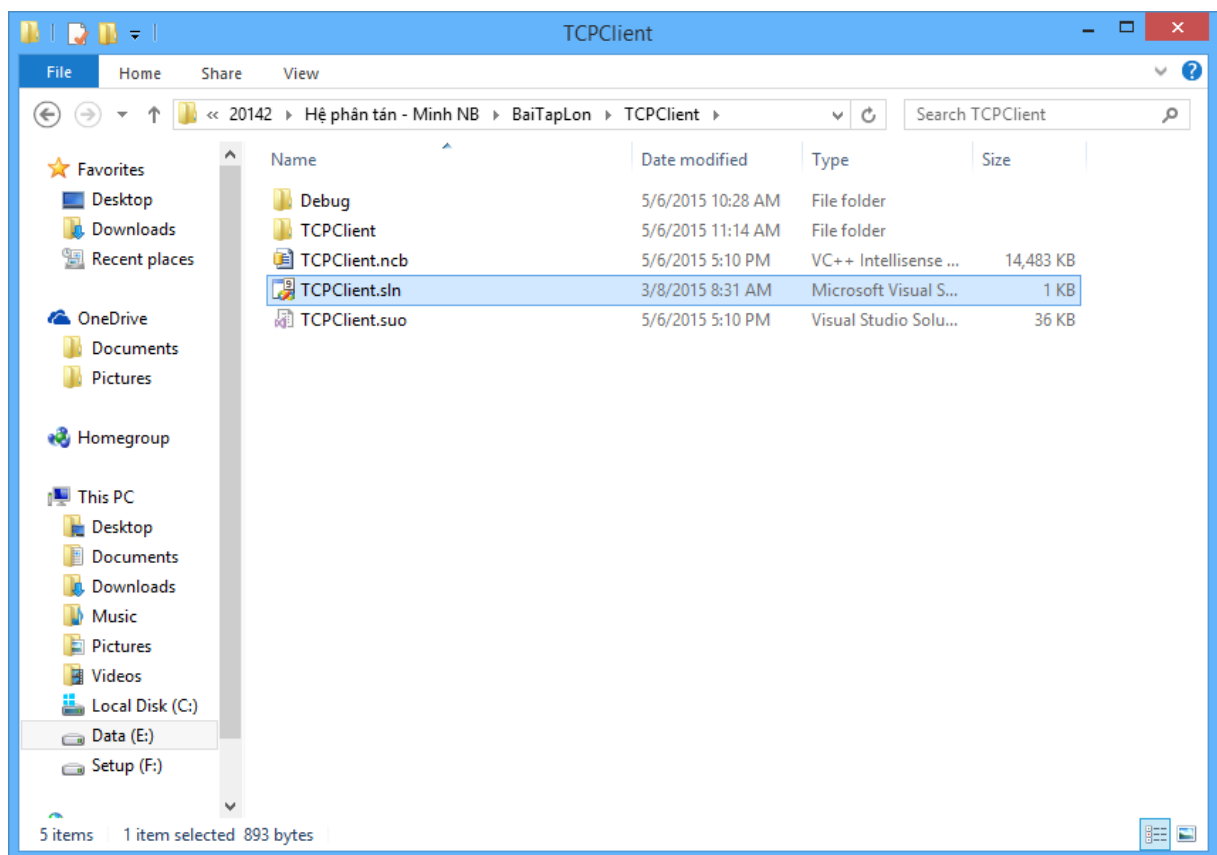
Cả Client và server đều được viết trên visual studio 2008 nên nếu sử dụng các bản visual cao hơn có thể sẽ không tương thích để có thể mở được Project.

Nếu chỉ để xem riêng lẻ mã nguồn thì Notepad++ là lựa chọn chấp nhận được nhưng để hiểu được project thì nhất định phải dùng visual studio.

Nguồn download:

<https://www.microsoft.com/en-us/download/details.aspx?id=41638>

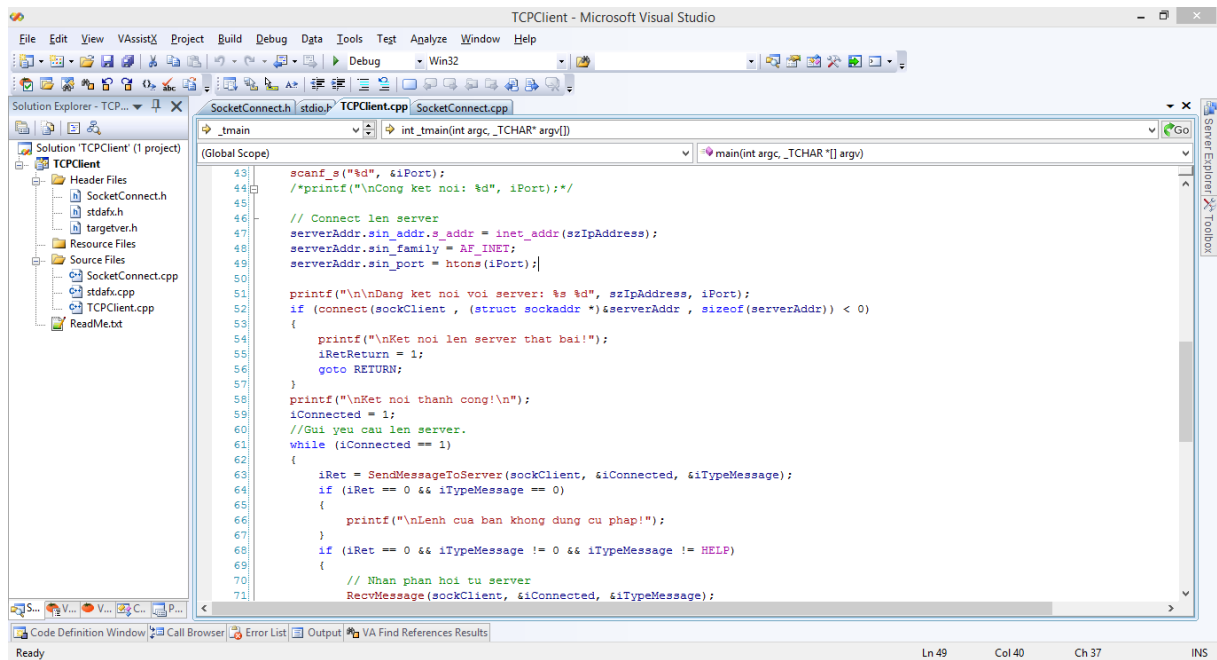
Dung lượng download có thể lên tới 3GB và quá trình cài đặt có thể mất khoảng 1h đồng hồ.



Hình 1: Thư mục của Project

Để mở project, ta click đúp vào file có TCPClient.sln với client và TCPServer.sln với server.

Sau khi mở được project, chúng ta có thể nhìn thấy như sau:



Hình 2: Giao diện Visual studio 2008

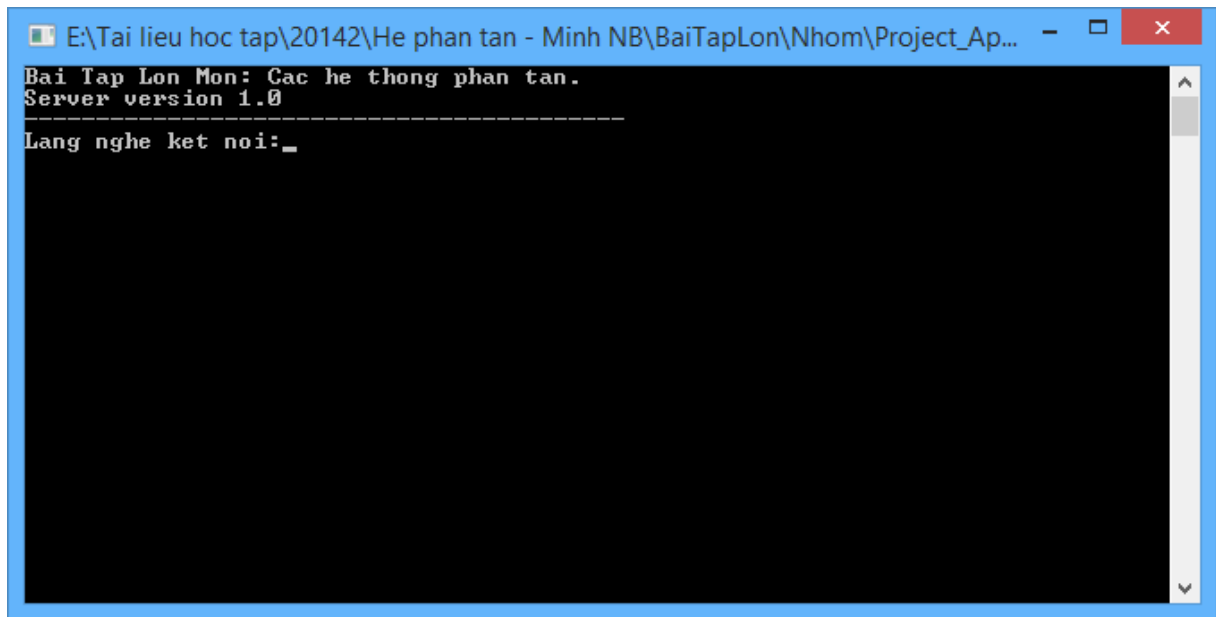
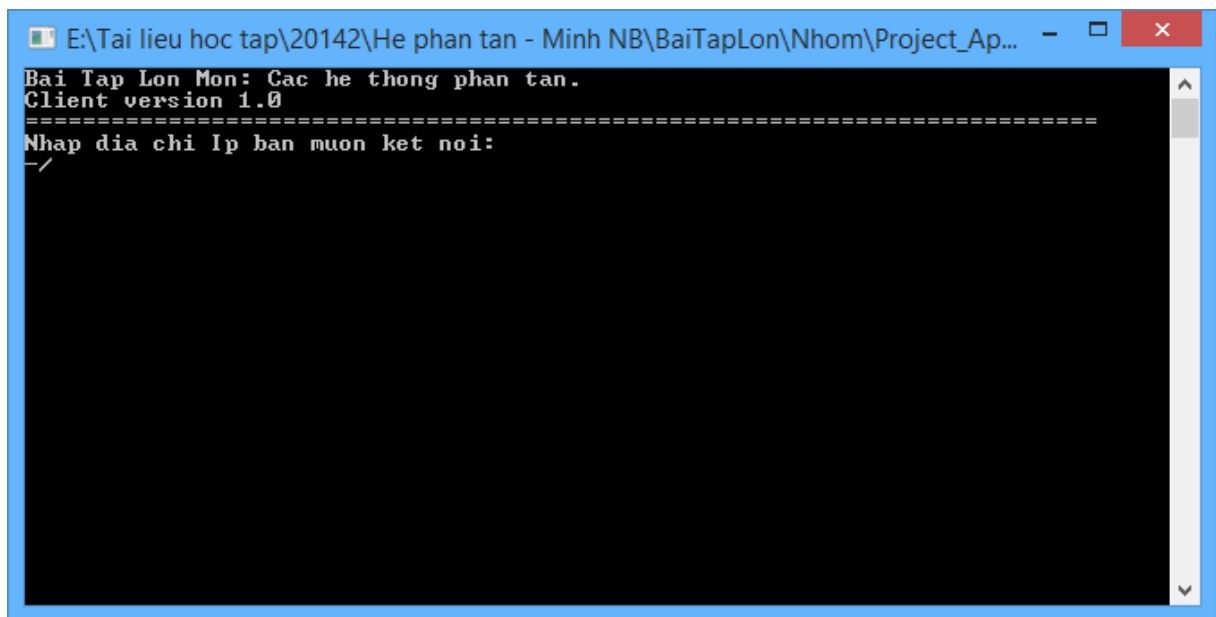
Click vào từng file mã nguồn (.cpp) và file thư viện (.h) để xem mã nguồn của chương trình.

2.3 Hướng dẫn cài đặt và thực thi chương trình đã được build ra file.

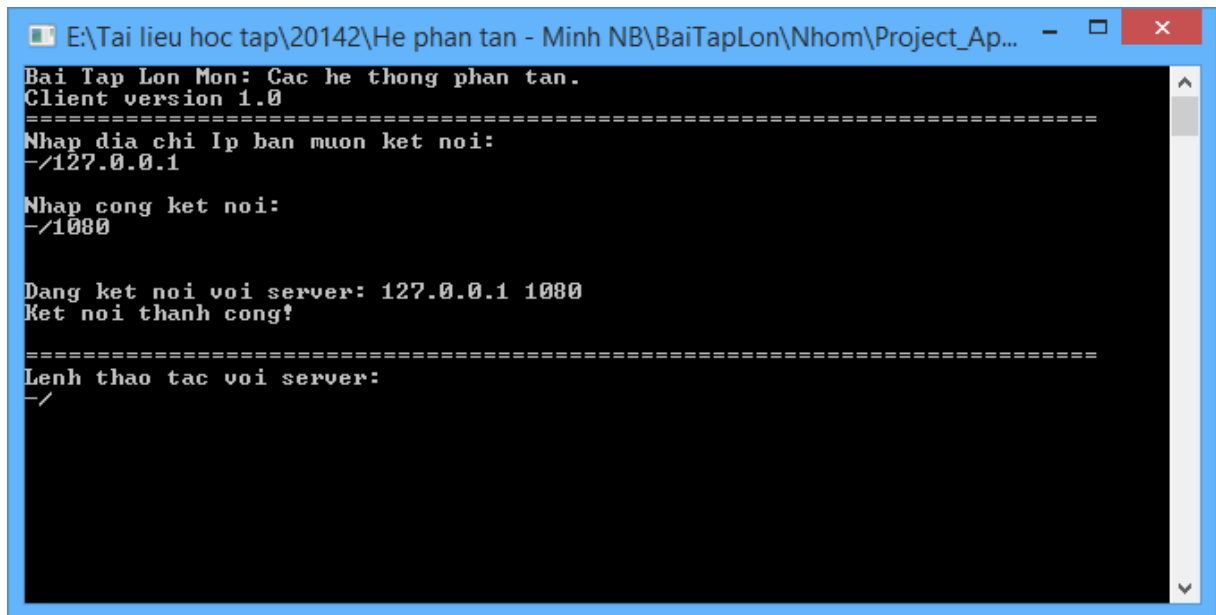
Để chạy server, vào thư mục Project_Application\Server chạy file TCPServer.exe.

Để chạy client, vào thư mục Project_Application\Client chạy file TCPClient.exe.

Lưu ý: chạy server trước client, nếu client không thiết lập được kết nối, nó sẽ thoát chương trình.

*Hình 3 : Server chạy**Hình 4: Client chạy*

Client: Nhập IP server và port là 1080 để kết nối tới server.



```
E:\Tai lieu hoc tap\20142\He phan tan - Minh NB\BaiTapLon\Nhom\Project_Ap...
Bai Tap Lon Mon: Cac he thong phan tan.
Client version 1.0
=====
Nhap dia chi Ip ban muon ket noi:
-/127.0.0.1

Nhap cong ket noi:
-/1080

Dang ket noi voi server: 127.0.0.1 1080
Ket noi thanh cong!
=====
Lenh thao tac voi server:
-/
```

Hình 5: Đã kết nối tới server

Dùng lệnh help để được hướng dẫn về các lệnh của hệ thống.

```

E:\Tai lieu hoc tap\20142\He phan tan - Minh NB\BaiTapLon\Nhom\Project_Ap...
Bai Tap Lon Mon: Cac he thong phan tan.
Client version 1.0
=====
Nhap dia chi Ip ban muon ket noi:
-/127.0.0.1

Nhap cong ket noi:
-/1080

Dang ket noi voi server: 127.0.0.1 1080
Ket noi thanh cong!

=====
Lenh thao tac voi server:
-/help

Cac lenh thao tac voi server:
=====
|   Lenh thao tac           ||           Y nghia           |
=====
| DangNhap "UserName" "Password" ||- Dang nhap vao server      |
| directory                  ||- Nhan duong dan thu muc hien tai. |
| listview                   ||- Xem danh sach cac file va thu muc |
|                             || trong thu muc hien tai         |
| DeleteFile "PathFile"      ||- Xoa file voi duong dan PathFile  |
| DeleteFolder "PathFolder"  ||- Xoa file voi duong dan PathFolder|
| MoveFile "a" "b"           ||- Chuyen file a sang file b       |
| MoveFolder "a" "b"         ||- Chuyen thu muc a sang thu muc b  |
| GetTime "Opt"              ||- Lay thoi gian server.         |
|                             || Opt = local => Lay thoi gian Local|
|                             || Opt = system => Lay thoi gian UTC |
| GetFile "a"                ||- Lay file a tu server ve may     |
| UpLoad "a"                 ||- Upload file len server         |
| help                       ||- Xem huong dan su dung         |
=====
Lenh thao tac voi server:
-/

```

Hình 6: Dùng lệnh help để được trợ giúp

Một số hình ảnh về các lệnh:

```

E:\Tai lieu hoc tap\20142\He phan tan - Minh NB\BaiTapLon\Nhom\Project_Ap...
! GetTime "Opt"                                     !!- Lay thoi gian server.      !
!                                                     !! Opt = local => Lay thoi gian Local !!
!                                                     !! Opt = system => Lay thoi gian UTC  !
!-----!
! GetFile "a"                                       !!- Lay file a tu server ve may  !
!-----!
! UpLoad "a"                                        !!- Upload file len server      !
!-----!
! help                                              !!- Xem huong dan su dung      !
!-----!
Lenh thao tac voi server:
-/listview

Server phan hoi danh sach folder va file :
Ban phai dang nhap: DangNhap "UserName" "Pass"

=====
Lenh thao tac voi server:
-/

```

Hình 7: Không đăng nhập thì không thao tác được

```

E:\Tai lieu hoc tap\20142\He phan tan - Minh NB\BaiTapLon\Nhom\Project_Ap...
!                                                     !! Opt = system => Lay thoi gian UTC !
!-----!
! GetFile "a"                                       !!- Lay file a tu server ve may  !
!-----!
! UpLoad "a"                                        !!- Upload file len server      !
!-----!
! help                                              !!- Xem huong dan su dung      !
!-----!
Lenh thao tac voi server:
-/listview

Server phan hoi danh sach folder va file :
Ban phai dang nhap: DangNhap "UserName" "Pass"

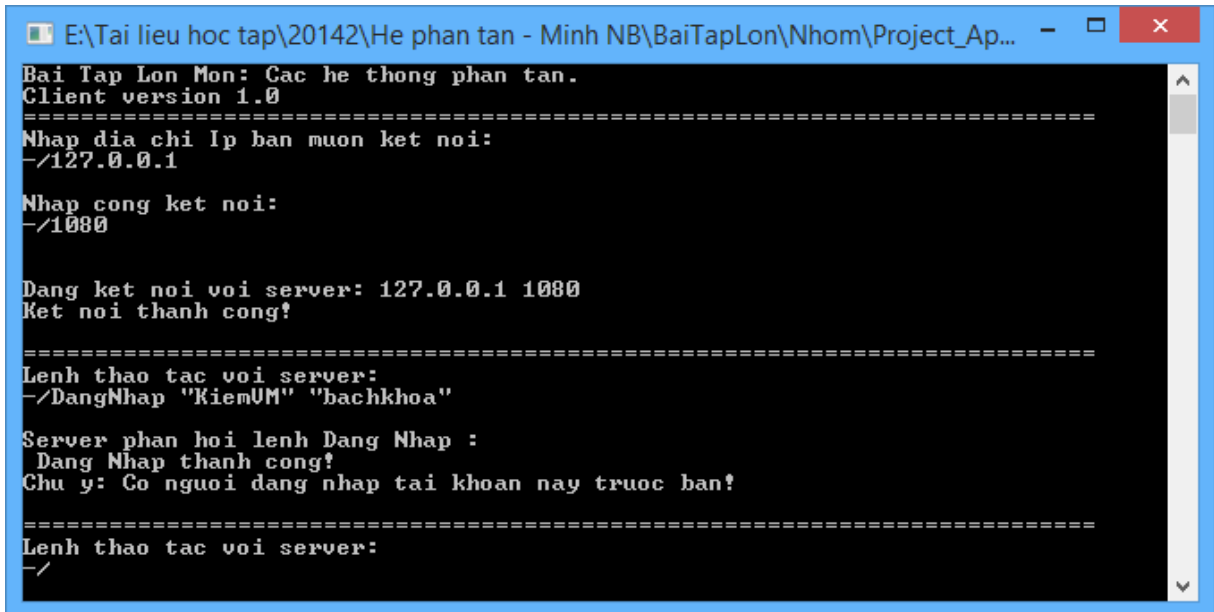
=====
Lenh thao tac voi server:
-/DangNhap "KiemUM" "bachkhoa"

Server phan hoi lenh Dang Nhap :
Dang Nhap thanh cong!

=====
Lenh thao tac voi server:
-/

```

Hình 8: Đăng nhập thành công



```

E:\Tai lieu hoc tap\20142\He phan tan - Minh NB\BaiTapLon\Nhom\Project_Ap...
Bai Tap Lon Mon: Cac he thong phan tan.
Client version 1.0
=====
Nhap dia chi Ip ban muon ket noi:
-/127.0.0.1

Nhap cong ket noi:
-/1080

Dang ket noi voi server: 127.0.0.1 1080
Ket noi thanh cong!

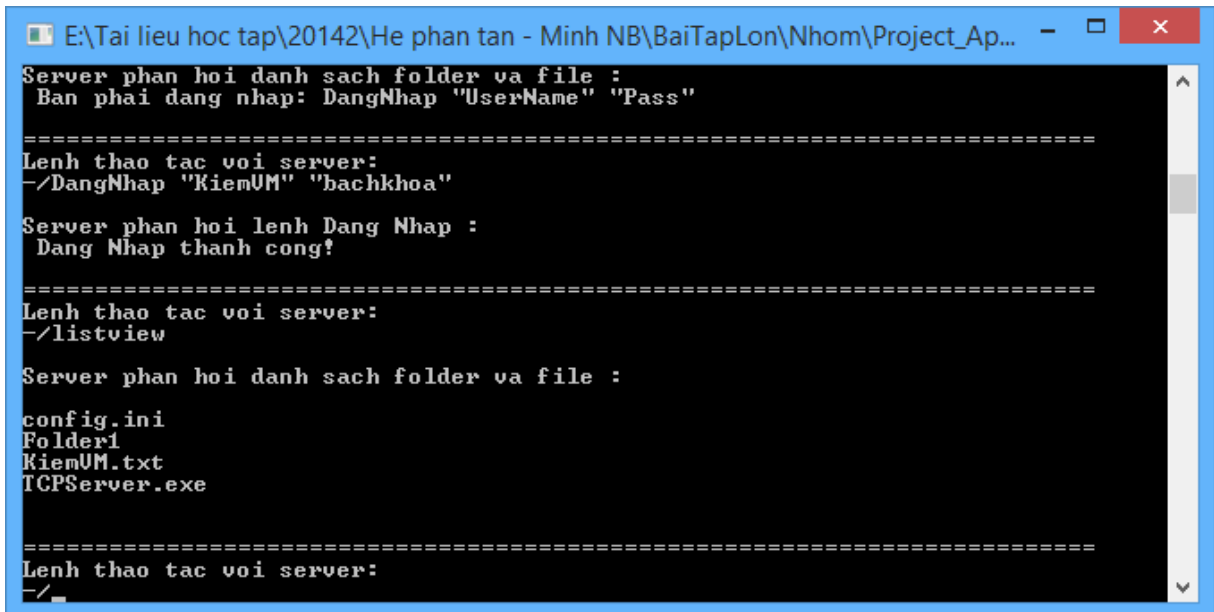
=====
Lenh thao tac voi server:
-/DangNhap "KiemUM" "bachkhoa"

Server phan hoi lenh Dang Nhap :
Dang Nhap thanh cong!
Chu y: Co nguoi dang nhap tai khoan nay truoac ban!

=====
Lenh thao tac voi server:
-/

```

Hình 9: Cảnh báo khi có 2 người cùng đăng nhập một tài khoản



```

E:\Tai lieu hoc tap\20142\He phan tan - Minh NB\BaiTapLon\Nhom\Project_Ap...
Server phan hoi danh sach folder va file :
Ban phải đăng nhập: DangNhap "UserName" "Pass"

=====
Lenh thao tac voi server:
-/DangNhap "KiemUM" "bachkhoa"

Server phan hoi lenh Dang Nhap :
Dang Nhap thanh cong!

=====
Lenh thao tac voi server:
-/listview

Server phan hoi danh sach folder va file :
config.ini
Folder1
KiemUM.txt
TCPServer.exe

=====
Lenh thao tac voi server:
-/

```

Hình 10: Lệnh listview

```

E:\Tai lieu hoc tap\20142\He phan tan - Minh NB\BaiTapLon\Nhom\Project_Ap...
config.ini
Folder1
KiemUM.txt
TCPServer.exe

=====
Lenh thao tac voi server:
-/GetTime "local"

Server phan hoi lenh get time :
Thoi gian Local yyyy/mm/dd : hh//mm//ss - 2015//05//07 : 00//38//21

=====
Lenh thao tac voi server:
-/GetTime "system"

Server phan hoi lenh get time :
Thoi gian System yyyy/mm/dd : hh//mm//ss - 2015//05//06 : 17//39//15

=====
Lenh thao tac voi server:
-/

```

Hình 11: Lấy thời gian Local và UTC

```

E:\Tai lieu hoc tap\20142\He phan tan - Minh NB\BaiTapLon\Nhom\Project_Ap...
=====
Lenh thao tac voi server:
-/GetTime "system"

Server phan hoi lenh get time :
Thoi gian System yyyy/mm/dd : hh//mm//ss - 2015//05//06 : 17//39//15

=====
Lenh thao tac voi server:
-/GetFile "KiemUM.txt"

Server phan hoi lenh Get file :
Thanh cong!

=====
Lenh thao tac voi server:
-/UpLoad "client.txt"

Server phan hoi lenh Up file :
Thanh cong!

=====
Lenh thao tac voi server:
-/

```

Hình 12: Lấy file từ server và upfile lên server

KẾT LUẬN VÀ LỜI CẢM ƠN

Chúng em xin chân thành cảm ơn thầy Nguyễn Bình Minh đã giảng dạy cho chúng em tận tình trong môn học các hệ phân tán. Chúng em đã tiếp thu được rất nhiều kiến thức thú vị từ mộ học này. Chúng em hy vọng nó sẽ là nền tảng và hành trang vững chắc giúp chúng em có thể tự tin khi làm việc cũng như đóng sức mình cho đất nước.

Project này do nhóm em tự code trên cơ sở phân tích yêu cầu đề bài, có một số nội dung khi tra hàm chúng em có tham khảo ví dụ code về hàm đó trên MSDN. Do đó chúng em nghĩ rằng mình không sử dụng mã nguồn mở hay code mẫu nào cho Project của mình.

*
* *

TÀI LIỆU THAM KHẢO

[1] Dr. Nguyen Binh Minh, lecture *IT4371: Distributed Systems*, Department of Information Systems, School of Information and Communication Technology, Hanoi University of Science and Technology, Spring 2015.

[2] Google code, *Introduction to Distributed System Design*, <http://www.hpcs.cs.tsukuba.ac.jp/~tatebe/lecture/h23/dsys/dsd-tutorial.html>, truy cập lần cuối ngày 03/05/2015.

[3] TS. Doãn Trung Tùng, slide bài giảng *Distributed Systems: Client – Server Programming*, Viện nghiên cứu quốc tế về khoa học và kỹ thuật tính toán.

[4] Hướng dẫn lập trình socket trên windows:
<http://diendan.congdongcviet.com/threads/t7362::lap-trinh-mang-thu-vien-winsoc-tren-visual-cpp.cpp>