

ĐẠI HỌC QUỐC GIA
THÀNH PHỐ HỒ CHÍ MINH

TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN
KHOA CÔNG NGHỆ THÔNG TIN

Báo cáo đồ án Socket

CSC10008 – MẠNG MÁY TÍNH

Ngô Nguyễn Thế Khoa 23127065
Hành Diễm Xuân 23127524

Ngày 25 tháng 6 năm 2024

Mục lục

| | | |
|-----|--|----|
| 1 | Thông tin nhóm | 2 |
| 2 | Bảng đánh giá mức độ hoàn thành | 3 |
| 3 | Nội dung | 4 |
| 3.1 | Thông tin chung về đề án: | 4 |
| 3.2 | Một số lưu ý | 4 |
| 3.3 | Cấu trúc đề án | 4 |
| 3.4 | Kịch bản giao tiếp chương trình: | 5 |
| 4 | Hướng dẫn sử dụng tính năng chương trình | 6 |
| 5 | Bảng phân công việc | 8 |
| 6 | Ảnh chụp màn hình | 9 |
| 7 | Nguồn tham khảo | 10 |

1 Thông tin nhóm

- **Môn:** Mạng máy tính
- **Lớp học phần:** 23CLC09
- **Giảng viên hướng dẫn:** Lê Hà Minh
- **Thành viên:**

| STT | Họ và tên | MSSV | Email | Vai trò |
|-----|---------------------|----------|--|-------------|
| 1 | Ngô Nguyễn Thế Khoa | 23127065 | nntkhoa23@clc.fitus.edu.vn | Trưởng nhóm |
| 2 | Hình Diễm Xuân | 23127524 | hdxuan23@clc.fitus.edu.vn | Thành viên |

2 Bảng đánh giá mức độ hoàn thành

| STT | Yêu cầu | Tiến độ | Minh họa |
|--|--|---------|----------|
| Phần I | | | |
| 1 | Client có thể nhận được danh sách các file từ Server và ctrl-c | 100% | Figure 2 |
| 2 | Client có thể nhận lần lượt từng file thành công từ Server. Server có thể gửi file thành công tới Client | 100% | Figure 2 |
| 3 | Hiển thị percent download file và phát hiện những file cần download tiếp theo | 100% | Figure 2 |
| Phần II | | | |
| 4 | Client có thể nhận được danh sách các file từ Server và ctrl-c | 100% | Figure 2 |
| 5 | 2s quét file input.txt 1 lần | 100% | Figure 2 |
| 6 | Hiển thị percent download files | 100% | Figure 2 |
| 7 | Client có thể nhận files thành công từ Server. Tập tin sau khi download phải đúng và đủ dung lượng | 100% | Figure 2 |
| 8 | Độ ưu tiên CRITICAL, HIGH, NORMAL | 100% | Figure 2 |
| Tổng kết báo cáo <i>Hoàn thành 100% yêu cầu đồ án</i> <i>Không xảy ra lỗi khi vận hành chương trình</i> | | | |

3 Nội dung

3.1 Thông tin chung về đồ án:

1. **Tên đồ án:** Lập trình Socket
2. **Môi trường lập trình:** Visual Studio Code, Windows
3. **Ngôn ngữ lập trình:** Python
4. **Danh sách thư viện chuẩn được sử dụng (standard libs)**
 - `argparse` - parse command line arguments
 - `enum` - usage of enum type
 - `json` - data serialization
 - `os` - file operations
 - `re` - usage of regex
 - `socket` - creating socket wrapper
 - `sys` - rendering progress bar
 - `threading` - handling multiple clients
 - `time` - time operations
 - `typing` - usage of generic type
5. **Danh sách thư viện bổ sung được sử dụng (external libs)**
 - `customtkinter` - build GUI
 - `python-dotenv` - load configs from `.env` files
 - `watchdog` - watching file changes
6. **Giao thức trao đổi giữa Client và Server:** Giao thức TCP

3.2 Một số lưu ý

- Cần cấu hình file `.env` trước khi chạy chương trình (cấu hình cho Server và Client)
- Tải thư viện bổ sung bằng lệnh `pip install -r requirements.txt` (trong thư mục Source)
- Chạy Server trước khi chạy bất kì Client nào

3.3 Cấu trúc đồ án

(Source): Thư mục chứa mã nguồn của chương trình

(app): Thư mục chứa mã nguồn của Server và Client

(client): Thư mục ứng dụng hoạt động (có thể tùy chỉnh)

(downloads): Thư mục chứa các file đã tải về (đường dẫn mặc định)

input.txt: File chứa danh sách file cần tải về (đường dẫn mặc định)

(server): Thư mục ứng dụng hoạt động (có thể tùy chỉnh)

(resources): Thư mục chứa các file có thể tải về (đường dẫn mặc định)

resources.json: File chứa danh sách file có thể tải về (tự động tạo và cập nhật)

(classes): Thư mục chứa các class hỗ trợ cho Server và Client

(shared): Thư mục chứa các biến chung cho Server và Client

(tests): Thư mục chứa các chương trình dùng để test hàm

(utils): Thư mục chứa các hàm hỗ trợ cho Server và Client

.env: File cấu hình cho Server và Client

requirements.txt: File chứa danh sách thư viện cần cài đặt

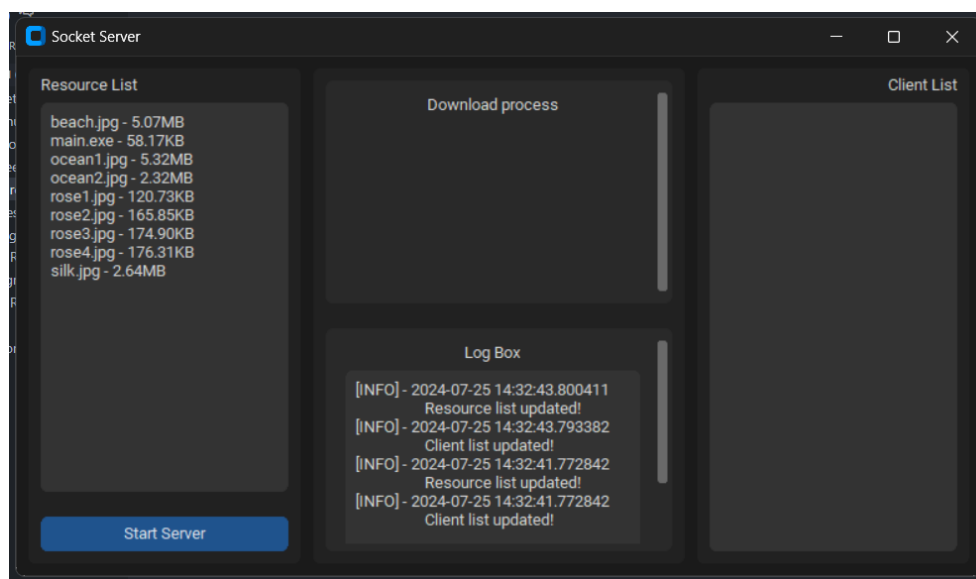
server.py: Mã nguồn của Server

client.py: Mã nguồn của Client

Report.pdf Báo cáo đồ án

3.4 Kịch bản giao tiếp chương trình:

- **Giao thức kết nối:** TCP
- **Cấu trúc thông điệp:**
 - **quit:** Thoát chương trình
 - **list:** Hiển thị danh sách file
- **Kiểu dữ liệu của thông điệp:** kiểu **string**, gửi bằng lệnh **send**, **sendall** và nhận bằng lệnh **recv** theo **FORMAT** được quy ước
- **Sơ đồ minh họa một phần quá trình giao tiếp giữa Client và Server sử dụng giao thức TCP:**



Hình 1: Sơ đồ minh họa kịch bản giao tiếp

4 Hướng dẫn sử dụng tính năng chương trình

Chuẩn bị: Cấu hình file `.env` trước khi chạy chương trình (cấu hình cho Server và Client) và tải thư viện bổ sung bằng lệnh `pip install -r requirements.txt` (trong thư mục Source).

- **Server:**

- **HOST:** Địa chỉ IP của Server
- **PORT:** Cổng kết nối của Server
- **SERVER_RESOURCES_PATH:** Đường dẫn chứa tài nguyên có thể tải về

- **Client:**

- **HOST:** Địa chỉ IP của Server
- **PORT:** Cổng kết nối của Server
- **CLIENT_DOWNLOADS_PATH:** Đường dẫn chứa tài nguyên đã tải về
- **CLIENT_REQUEST_INPUT:** Đường dẫn file chứa danh sách file cần tải về

Bước 1: Chạy file `server.py` bằng lệnh `py server.py`, lúc này một Socket Server sẽ chạy và chờ các kết nối từ Client.

Chạy file `client.py` bằng lệnh `py client.py` khởi tạo một Socket Client và tự động kết nối đến Server.

Bước 2: Sau khi Client kết nối thành công, Server sẽ hiển thị địa chỉ của Client đã kết nối và gửi đi danh sách tài nguyên có thể tải về cho Client.

Bước 3: Nhập những file muốn tải vào `input.txt` trong đường dẫn của ứng dụng Client (mặc định ở `Source/app/client/input.txt`). Client sẽ tự động đọc (mỗi 2s) và tải

file trong danh sách theo mức độ ưu tiên được quy ước từ trước.

Ngoài ra còn có thể nhập lệnh bằng `console` với quy tắc giao tiếp:

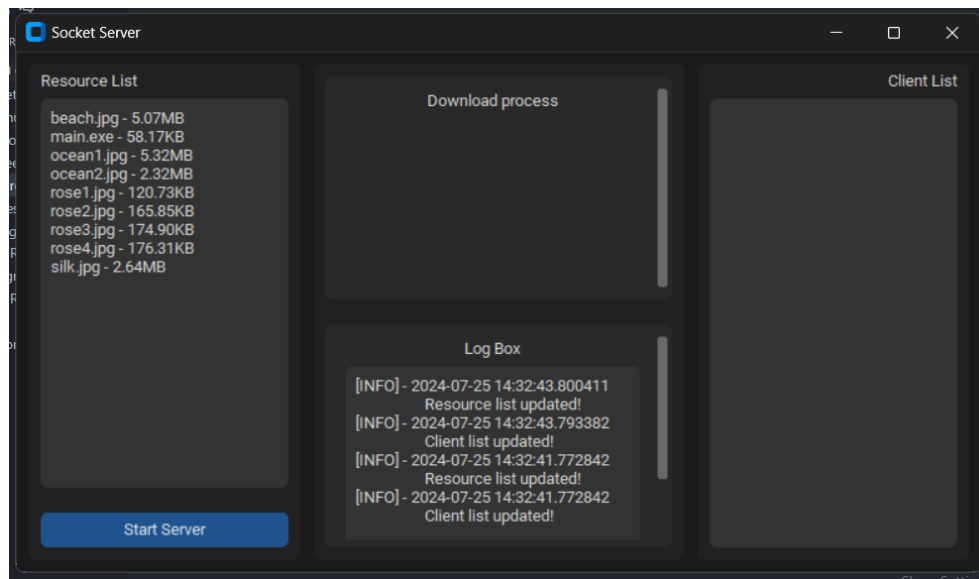
- `"help", "h"`: khi gõ câu lệnh này Server sẽ trả lời `Show all commands` và in ra tất cả câu lệnh của `commands`.
- `"quit", "q"`: khi gõ câu lệnh này Server sẽ trả lời `Quit the program` và sẽ ngắt kết nối với Client.
- `"list", "l"`: khi gõ câu lệnh này Server sẽ trả lời `Get lists of available files from Server` và in ra danh sách các file có thể tải xuống từ server.
- `"get", "g"`: khi gõ lệnh này người dùng có thể tải file từ server theo tên file người dùng nhập bằng `console`.

Bước 4: Ngắt kết nối bằng cách `Ctrl+C` (có thể thực hiện ở cả server và client) hoặc nhập lệnh `"q"` (chỉ ở client).

5 Bảng phân công việc

| STT | Công việc | Người thực hiện |
|----------------|--|-----------------|
| Part I | | |
| 1 | Client có thể nhận được danh sách các file từ Server và ctrl-c | 0,5 |
| 2 | Client có thể nhận lần lượt từng file thành công từ Server. Server có thể gửi file thành công tới Client | 2 |
| 3 | Hiển thị percent download file và phát hiện những file cần download tiếp theo 2s quét file input.txt 1 lần | 0,5 |
| Phần II | | |
| 4 | Client có thể nhận được danh sách các file từ Server và ctrl-c | 0,5 |
| 5 | 2s quét file input.txt 1 lần | 0,5 |
| 6 | Hiển thị percent download files | 1 |
| 7 | Client có thể nhận files thành công từ Server. Tập tin sau khi download phải đúng và đủ dung lượng | 3 |
| 8 | Độ ưu tiên CRITICAL, HIGH, NORMAL | 1,5 |
| 9 | Viết báo cáo | 1,5 |

6 Ảnh chụp màn hình



Hình 2: Giao diện Server

7 Nguồn tham khảo

1. [geeksforgeeks](#)
2. [Keyboard interrupt sockets and threads](#)