**ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH**

**TRƯỜNG ĐẠI HỌC BÁCH KHOA**

**KHOA KHOA HỌC VÀ KỸ THUẬT MÁY TÍNH**



**MẠNG MÁY TÍNH**

|  |
| --- |
| Assignment 1:  **STREAMMING VIDEO USING RTSP AND RTP** |

**Giáo viên hướng dẫn: Nguyễn Hồng Nam**

**Lớp:** L14

**Đề tài 1:** Xem xét quản lý mua / bán sách tại hiệu sách trực tuyến.

**DANH SÁCH THÀNH VIÊN**

1. Đào Thanh Phương – 1813611
2. Thạch Hoàng Sơn – 1813863
3. Ngô Lê Gia Thuấn – 1810562
4. Nguyễn Đăng Anh Phương – 1813621

TP. HỒ CHÍ MINH - THÁNG 11/2020

1. **REQUIREMENTS ANALYSIS**

Functional requirements:

* Truyền được video từ server đến client, hiện thực sao cho việc truyền đi từ server là liên tục đến khi gặp lệnh pause hoặc yêu cầu ngừng truyền.
* Đầy đủ các chức năng và các chức năng hoạt động đúng yêu cầu:

+ Setup: thiết lập video, khởi tạo kết nối.

+ Play: bắt đầu phát video.

+ Pause: tạm thời ngừng nhận gói tin từ server, phía client tạm thời ngừng phát video.

+ Teardown: ngừng hoàn toàn việc phát và gửi packet, đóng cửa sổ GUI.

Non-functional requirements:

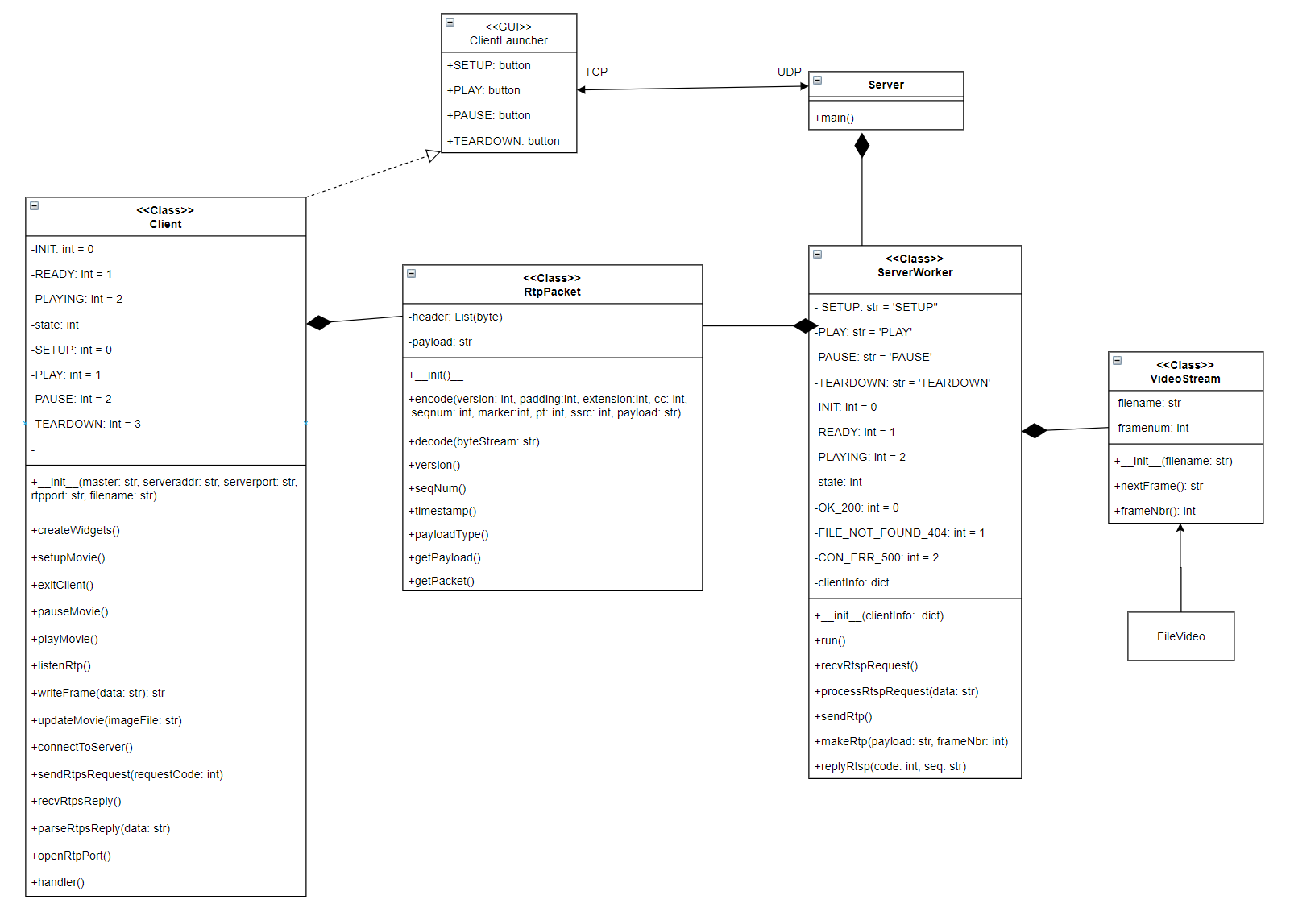
* Hiện thực sao cho giảm thiểu tình trạng giật, lag xảy ra ở phía client.
* Không yêu cầu GUI quá hoàn chỉnh, chỉ cần đủ chức năng.
* Khi đóng GUI bằng bất kỳ hình thức nào cũng đều cần phải hủy kết nối giữa server và client.

1. **FUNCTION DESCRIPTION**

Function description:

|  |  |
| --- | --- |
| Name | Description |
| **ClientLauncher.py** | |
| Client(...) | thêm các chức năng của app |
| Tk() | tạo một GUI |
| mainloop() | lặp liên tục tới đóng cửa sổ |
| **VideoStream.py** | |
| \_\_init\_\_(filename) | khởi tạo con trỏ đọc filename, đặt giá trị frameNum = 0 |
| nextFrame() | lấy frame tiếp theo |
| frameNbr() | lấy frameNum |
| **Server.py** | |
| main() | tạo/mở một server port; nhận rtp packet, lấy thông tin client gửi đi xử lý |
| **ServerWorker.py** | |
| \_\_init\_\_(clientInfo) | nhận clientInfo được truyền đến |
| run() | tạo thread nhận rtsp request |
| recvRtspRequest() | nhận rtsp request từ khách hàng |
| processRtspRequest() | xử lý rtsp request nhận được từ khách hàng |
| sendRtp() | gửi rtp packet cho khách hàng qua udp |
| makeRtp(payload, frame Nbr) | tạo một rtp packet lưu dữ liệu của video |
| replyRtsp(code, seq) | gửi rtsp reply cho khách hàng |
| **RtpPacket.py** | |
| \_\_init\_\_ |  |
| encode(...) | tạo header và payload cho rtp packet |
| decode(byteStream) | tách header và payload trong rtp packet |
| version() | trả về version |
| seqNum() | trả về seqNum |
| timestamp() | trả về timestamp |
| payloadType() | trả về payload type |
| getPayload() | lấy payload |
| getPacket() | lấy rtp packet |
| **Client.py** | |
| \_\_init\_\_ | khởi tạo địa chỉ server, server port, rtp port, tên file |
| createWidgets() | tạo GUI (gồm 4 nút: setup, play, pause, teardown) |
| setupMovie() | gửi Rtsp request với mã setup |
| exitClient() | gửi Rtsp request với mã teardown, và đóng gui |
| pauseMovie() | gửi Rtsp request với mã pause |
| playMovie() | gửi Rtsp request với mã play(kiểm tra trạng thái của rtsp, tạo thread nhận rtp packets) |
| listenRtp() | nhận và xử lý rtp packets, ngừng khi có request PAUSE hoặc TEARDOWN |
| writeFrame(data) | ghi các frame nhận được vào file ảnh, trả về file ảnh |
| updateMovie(imageFile) | cập nhật file ảnh trên GUI |
| connectToServer() | kết nối tới server, tạo một rtsp/tcp mới |
| sendRtspResquest(requestCode) | gửi Rtsp request tới server |
| recvRtspReply() | nhận một Rtsp reply từ server |
| parseRtspReply(data) | phân tích Rtsp reply từ server |
| openRtpPort() | mở một Rtp socket liên kết tới port được chỉ định |
| handler() | xử lý tác vụ đóng cửa sổ GUI |

1. **CLASS DIAGRAM**

****

1. **A SUMMATIVE EVALUATION OF RESULTS ACHIEVED**

Các thành viên trong nhóm làm tròn nhiệm vụ đã được trưởng nhóm giao phó, nộp kết quả đúng tiến độ. Về cơ bản nhóm đã hoàn thành đủ các tiêu chí mà Requirements Analysis đã đề ra:

1. Đã hiện thực được việc đóng gói video thành các RTP và gửi đi đúng yêu cầu.
2. Đã hiện thực được RTSP và giữ cho client’s state luôn cập nhật.
3. Các chức năng Setup, Play, Pause, Teardown thực hiện đúng theo yêu cầu người dùng.
4. Khi đóng GUI, kết nối được ngắt.
5. Việc phát video hầu như không bị giật lag.
6. **USER MANUAL**

Dùng lệnh python Server.py server\_port để mở server. Port chuẩn của RTSP là 554, nhưng người dùng nên chọn một port bất kỳ lớn hơn 1024. Tiếp theo, mở client bằng lệnh python ./ClientLauncher.py server\_host server\_port RTP\_port video\_file\_direction trong đó server\_host là tên của máy đang mở server, server\_port là port người dùng chọn ở lệnh trên, RTP\_port là port dùng để nhận RTP packets, video\_file\_direction là tên của video file người dùng yêu cầu. Khi thực hiện đúng các bước trên, một cửa sổ sẽ được mở ra. Gồm có các button: Setup, Play, Pause, Teardown. Chức năng các button được diễn giải như sau:

* Setup: đặt tham số session và transport. Phải sử dụng chức năng này trước khi phát video.
* Play: sử dụng nút này khi muốn bắt đầu phát video.
* Pause: sử dụng nút này khi muốn tạm ngừng video.
* Teardown: sử dụng nút này khi muốn ngừng phát video và thoát khỏi cửa sổ phát video. Có thể sử dụng nút X ở góc phải màn hình cũng cho chức năng tương tự.