**TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TPHCM**

**KHOA ĐÀO TẠO CHẤT LƯỢNG CAO**

**----------------**

Logo

Description automatically generated

**KHÓA LUẬN TỐT NGHIỆP**

**THIẾT KẾ VÀ XÂY DỰNG WEB APPLICATION**

**CHỈNH SỬA VIDEO**

**Giáo viên hướng dẫn: Ts. Huỳnh Xuân Phụng**

**Sinh viên thực hiện: Trần Quốc Việt**

**MSSV: 18110238**

**Khóa: K18**

**Ngành: Công Nghệ Thông Tin**

Tp.Hồ Chí Minh, … tháng … năm 2023

**TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TPHCM**

**KHOA ĐÀO TẠO CHẤT LƯỢNG CAO**

**----------------**

Logo

Description automatically generated

**KHÓA LUẬN TỐT NGHIỆP**

**THIẾT KẾ VÀ XÂY DỰNG WEB APPLICATION**

**CHỈNH SỬA VIDEO**

**Giáo viên hướng dẫn: Ts. Huỳnh Xuân Phụng**

**Sinh viên thực hiện: Trần Quốc Việt**

**MSSV: 18110238**

**Khóa: K18**

**Ngành: Công Nghệ Thông Tin**

Tp.Hồ Chí Minh, … tháng … năm 2023



CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM

Độc lập – Tự do – Hạnh phúc

---\*\*\*---

Tp. Hồ Chí Minh, ngày … tháng … năm …

# NHIỆM VỤ ĐỒ ÁN TỐT NGHIỆP

**Họ và tên Sinh viên**: Trần Quốc Việt MSSV: 18110238

**Ngành**: Công Nghệ Thông Tin Lớp: 18110ST1

**Giảng viên hướng dẫn**: Ts. Huỳnh Xuân Phụng

**Ngày nhận đề tài**: **Ngày nộp đề tài**:

1. **Đề tài**: THIẾT KẾ VÀ XÂY DỰNG WEB APPLICATION CHỈNH SỬA VIDEO
2. **Nội dung thực hiện đề tài:**

* Tìm hiểu các tác vụ cơ bản của việc chỉnh sửa video.
* Tìm hiểu về giới hạn của web application khi thực hiện việc chỉnh sửa video
* Tìm hiểu và thiết kế giao diện một web application bằng framework Angular
* Tìm hiểu về các công cụ cho phép thao tác lên video
* Tìm hiều giới hạn và xây dựng hệ thống có thể rendering video bằng NodeJS

1. **Sản phẩm:** -WebApplication có thể thực hiện được các thao tác chỉnh sửa video thông dụng và hệ thống rendering video

TRƯỞNG NGÀNH GIÁO VIÊN HƯỚNG DẪN



CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM

Độc lập – Tự do – Hạnh phúc

---\*\*\*---

# NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN

**Họ và tên Sinh viên**: Trần Quốc Việt MSSV: 18110238

**Tên đề tài:** THIẾT KẾ VÀ XÂY DỰNG WEB APPLICATION CHỈNH SỬA VIDEO

**Họ và tên Giáo viên hướng dẫn**: Ts. Huỳnh Xuân Phụng

**NHẬN XÉT**

1. Về nội dung đề tài và khối lượng thực hiện:

1. Ưu điểm:

1. Khuyết điểm:

1. Đề nghị cho bảo vệ hay không?

1. Điểm (Bằng chữ: )

Tp. Hồ Chí Minh, ngày … tháng … năm 20…

Giáo viên hướng dẫn

(Ký tên & ghi rõ họ tên)



CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM

Độc lập – Tự do – Hạnh phúc

---\*\*\*---

# NHẬN XÉT CỦA GIÁO VIÊN PHẢN BIỆN

**Họ và tên Sinh viên**: Trần Quốc Việt MSSV: 18110238

**Tên đề tài:** THIẾT KẾ VÀ XÂY DỰNG WEB APPLICATION CHỈNH SỬA VIDEO

**Họ và tên Giáo viên phản biện**:

**NHẬN XÉT**

1. Về nội dung đề tài và khối lượng thực hiện:

1. Ưu điểm:

1. Khuyết điểm:

1. Đề nghị cho bảo vệ hay không?

1. Điểm (Bằng chữ: )

Tp. Hồ Chí Minh, ngày … tháng … năm 20…

Giáo viên phản biện

(Ký tên & ghi rõ họ tên)

# LỜI CẢM ƠN

Đầu tiên em xin gửi lời cảm ơn đến thầy cô và lãnh đạo của Trường Đại học Sư phạm và Kỹ Thuật TP. Hồ Chí Minh, đã tạo điều kiện thuận lợi về học thuật và cở sở vật chất cho em có thể học tập, rèn luyện, củng cố kiến thức và phát triển bản thân trong suốt thời gian em theo học tại trường.

Nhờ sự giúp đỡ từ các giáo viên của khoa Công nghệ thông tin cũng như khoa Đào tạo Chất Lượng Cao nói chung, và giáo viên hướng dẫn của em, thầy Huỳnh Xuân Phụng, nói riêng đã luôn theo sát, luôn luôn giải đáp thắc mắc và hỗ trợ em đưa ra các hướng giải quyết các vấn đề khó khăn em gặp phải trong khoảng thời gian em thực hiện đề tài. Em thật sự chân thành gửi lời cảm ơn đến với các thầy cô.

Cùng với đó em xin gửi lời cảm ơn đến bạn bè đã cùng em tham gia học tập, đặt biệt là các bạn ở lớp CL2B, đã cùng em cố gắng học tập để có thể đi tới thời điểm hiện tại. Ngoài ra, em cũng muốn gửi lời cảm ơn đến các anh/ chị Ban Truyền thông Khoa Chất Lượng Cao, đã tạo điều kiện cho em tiếp xúc với nhiều vấn đề ngoài học thuật chuyên môn, từ đó cho em nhiều góc nhìn về phương pháp giải quyết vấn đề.

Và cuối cùng là lời cảm ơn gửi tới gia đinh đã luôn tin tưởng, đặt niềm tin vào con đường học thuật mà em chọn và luôn động viên em trong khoản thời gian làm đề tài này.

Em xin chân thành cảm ơn.

# TÓM TẮT

Hiện tại dưới sự phát triển nhanh chóng của mạng Internet, đặt biệt là sau đợt bùng phát dịch vào hai quí cuối năm 2021, số lượt truy cập vào các mạng xã hội từ nổi tiếng cho tới những cái tên xa lạ tăng nhanh, ta có thể dễ dàng thấy được điều này thông qua việc các trang tin, báo đài trong thời gian này đề cập tới các mạng xã hội thuần Việt như Lotus, Butta, Gapo, …Tuy có sự tăng trưởng lại tương đối, nhưng trong thời điểm này Tiktok là một trong những cái tên bùng nổ, với danh hiệu mạng xã hội chia sẽ video lớn nhất thế giới vượt qua cả Youtube. Như Isaac Newton từng nói “Nếu tôi nhìn xa hơn đó là vì tôi đang đứng trên vai người khổng lồ”, vì vậy thay vì tạo ra các mạng xã hội và chen chút với các ông lớn trong ngành thì em hướng tới các công cụ bổ trợ cho họ và thế ý tưởng về một “Web Application chỉnh sửa video” được ra đời. Web Application này sẽ hướng tới các đối tượng người dùng mới bắt đầu, với cách thức làm việc tối giản để đạt hiệu quả về hiệu năng tốt nhất

# MỤC LỤC

[NHIỆM VỤ ĐỒ ÁN TỐT NGHIỆP 1](#_Toc132911828)

[NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN II](#_Toc132911829)

[NHẬN XÉT CỦA GIÁO VIÊN PHẢN BIỆN 3](#_Toc132911830)

[LỜI CẢM ƠN IV](#_Toc132911831)

[TÓM TẮT V](#_Toc132911832)

[MỤC LỤC VI](#_Toc132911833)

[DANH MỤC BẢNG IX](#_Toc132911834)

[DANH MỤC HÌNH X](#_Toc132911835)

[DANH MỤC TỪ VIẾT TẮT XI](#_Toc132911836)

[CHƯƠNG 1. TỔNG QUAN VỀ ĐỀ TÀI 1](#_Toc132911837)

[1.1. Lý do chọn đề tài 1](#_Toc132911838)

[1.1. Mục tiêu 1](#_Toc132911839)

[1.2. Công nghệ sử dụng. 1](#_Toc132911840)

[CHƯƠNG 2. CƠ SỞ LÝ THUYẾT 2](#_Toc132911841)

[2.1. NodeJS 2](#_Toc132911842)

[2.1.1. Giới thiệu. 2](#_Toc132911843)

[2.1.2. Ưu điểm. 2](#_Toc132911844)

[2.1.3. Nhược điểm. 2](#_Toc132911845)

[2.1.4. Lý do sử dụng. 2](#_Toc132911846)

[2.2. Angular. 2](#_Toc132911847)

[2.2.1. Giới thiệu. 2](#_Toc132911848)

[2.2.2. Ưu điểm. 3](#_Toc132911849)

[2.2.3. Nhược điểm. 3](#_Toc132911850)

[2.2.4. Lý do sử dụng. 3](#_Toc132911851)

[2.2.5. Các thư viện sử dụng. 3](#_Toc132911852)

[2.3. FFmpeg. 3](#_Toc132911853)

[2.3.1. Giới thiệu. 3](#_Toc132911854)

[2.3.2. Ưu điểm. 4](#_Toc132911855)

[2.3.3. Nhược điểm. 4](#_Toc132911856)

[2.3.4. Lý do sử dụng. 4](#_Toc132911857)

[CHƯƠNG 3. KHẢO SÁT HIỆN TRẠNG VÀ XÁC ĐỊNH YÊU CẦU 5](#_Toc132911858)

[3.1. Khảo sát hiện trạng 5](#_Toc132911859)

[3.1.1. Trang web Clipchamp.com 5](#_Toc132911860)

[3.1.2. Phần mềm DaVinci Resolve 6](#_Toc132911861)

[3.2. Xác định yêu cầu. 6](#_Toc132911862)

[3.2.1. Nghiệp vụ 6](#_Toc132911863)

[3.2.1.1. Đối tượng sử dụng 6](#_Toc132911864)

[3.2.1.2. Chức năng 7](#_Toc132911865)

[3.3. Đánh giá và lựa chọn công nghệ 7](#_Toc132911866)

[3.3.1. Mục tiêu đề tài 7](#_Toc132911867)

[3.3.2. Nội dung 7](#_Toc132911868)

[3.3.3. Môi trường lập trình 7](#_Toc132911869)

[3.3.4. Phương pháp nghiên cứu 8](#_Toc132911870)

[3.3.5. Ý nghĩa khoa học, thực tiễn 8](#_Toc132911871)

[3.4. Mô hình hóa hệ thống 9](#_Toc132911872)

[3.4.1. Sơ đồ usecase 9](#_Toc132911873)

[3.4.2. Mô tả sơ đồ usecase 11](#_Toc132911874)

[3.5. Đặc tả Usecase 12](#_Toc132911875)

[3.5.1. Usecase đăng tải video 12](#_Toc132911876)

[3.5.2. Usecase tạo điểm cắt video 13](#_Toc132911877)

[3.5.3. Usecase gửi yêu cầu render 13](#_Toc132911878)

[3.5.4. Usecase nhận video từ client 14](#_Toc132911879)

[3.5.5. Usecaser ghép video 15](#_Toc132911880)

[3.5.6. Usecase render video 15](#_Toc132911881)

[3.5.7. Usecase xóa dữ liệu cũ 16](#_Toc132911882)

[CHƯƠNG 4. THIẾT KẾ WEBSITE 17](#_Toc132911883)

[4.1. Thiết kế hệ thống 17](#_Toc132911884)

[4.1.1. Lược đồ 17](#_Toc132911885)

[4.1.1.1. Lược đồ tuần tự 17](#_Toc132911886)

[4.1.1.2. Lược đồ hoạt động 21](#_Toc132911887)

[4.1.2. Tổng quát chương trình 23](#_Toc132911888)

[4.1.2.1. Tổng quát tại Client 23](#_Toc132911889)

[4.1.2.2. Tổng quát tại Server 23](#_Toc132911890)

[4.2. Thiết kế giao diện 24](#_Toc132911891)

[4.2.1. Danh sách giao diện 24](#_Toc132911892)

[4.2.2. Mô tả chi tiết giao diện 26](#_Toc132911893)

[4.2.2.1. Giao diện đăng tải nội dung video 26](#_Toc132911894)

[4.2.2.2. Giao diện video 27](#_Toc132911895)

[4.2.2.3. Giao diện danh sách các điểm cắt 30](#_Toc132911896)

[4.2.2.4. Giao diện ghép nội dung video 31](#_Toc132911897)

[4.2.2.5. Giao diện modal setting 33](#_Toc132911898)

[4.3. Thiết kế xử lý dữ liệu đoạn video 34](#_Toc132911899)

[4.3.1. Thiết kế cách thức cắt video 34](#_Toc132911900)

[4.3.2. Thiết kế cách thức ghép video 35](#_Toc132911901)

[4.3.3. Thiết kê phương thức tăng tốc video 36](#_Toc132911902)

[CHƯƠNG 5. CÀI ĐẶT VÀ KIỂM THỬ 37](#_Toc132911903)

[5.1. Cài đặt 37](#_Toc132911904)

[5.1.1. Cài đặt môi trường 37](#_Toc132911905)

[5.1.2. Cài đặt phần mềm 37](#_Toc132911906)

[5.2. Kiểm thử 38](#_Toc132911907)

[5.2.1. Kiểm thử chức năng upload 38](#_Toc132911908)

[5.2.2. Kiểm thử chức năng send request và render video 40](#_Toc132911909)

[CHƯƠNG 6. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN 42](#_Toc132911910)

[6.1. Tổng kết 42](#_Toc132911911)

[6.2. Kết luận 42](#_Toc132911912)

[6.2.1. Về mặt lý thuyết 42](#_Toc132911913)

[6.2.2. Về mặt ứng dụng 42](#_Toc132911914)

[6.2.3. Ưu điểm 42](#_Toc132911915)

[6.2.4. Nhược điểm 42](#_Toc132911916)

[6.3. Hướng phát triển 43](#_Toc132911917)

[TÀI LIỆU THAM KHẢO 44](#_Toc132911918)

# DANH MỤC BẢNG

[Bảng 1: Bảng mô tả nội chi tiết dung sơ đồ usecase 11](#_Toc132911962)

[Bảng 2 Bảng mô tả chi tiết usecase đăng tải video 12](#_Toc132911963)

[Bảng 3 bảng mô tả chi tiết usecase tạo điểm cắt video 13](#_Toc132911964)

[Bảng 4 bản mô tả chi tiết usecase yêu cầu render 13](#_Toc132911965)

[Bảng 5 bảng mô tả chi tiết usecase nhận video từ client 14](#_Toc132911966)

[Bảng 6 bảng mô tả chi tiết usecase ghép video 15](#_Toc132911967)

[Bảng 7 bảng mô tả chi tiết usecase render video 15](#_Toc132911968)

[Bảng 8 bảng mô tả chi tiết usecase xóa dữ liệu cũ 16](#_Toc132911969)

[Bảng 9 Bảng giới thiệu về các thư viện sử dụng chung với Angular 14 tại Frontend 23](#_Toc132911970)

[Bảng 10 Bảng giới thiều về các thư viện được sử dụng với NodeJS tại Backend 23](#_Toc132911971)

[Bảng 11 bảng mô tả giao diện chung 25](#_Toc132911972)

[Bảng 12 bảng mô tả chi tiết giao diện đăng tải nội dung video 26](#_Toc132911973)

[Bảng 13 bảng mô tả chi tiết giao diện video 28](#_Toc132911974)

[Bảng 14 bảng mô tả chi tiết giao diện danh sách các điểm cắt 30](#_Toc132911975)

[Bảng 15 bảng mô tả chi tiết giao diện ghép nội dung video 31](#_Toc132911976)

[Bảng 16 bảng mô tả chi tiết giao diện modal setting 33](#_Toc132911977)

[Bảng 17 Bảng mô tả chi tiết thiết kế cách thức cắt video 34](#_Toc132911978)

[Bảng 18 Bảng mô tả chi tiết thiết kế cách thức ghép video 35](#_Toc132911979)

[Bảng 19 Bảng mô tả chi tiết thiết kế cách thức tăng tốc video 36](#_Toc132911980)

[Bảng 20 Bảng mô tả kiểm thử chức năng upload video 38](#_Toc132911981)

[Bảng 21 Bảng mô tả chức năng send request và render video 40](#_Toc132911982)

# DANH MỤC HÌNH

[Hình 1 Hình anh chương trình clipchamp tại trang clipchamp.com 6](#_Toc122198520)

[Hình 2 Hình anh phần mềm DaVici Resolve 18 7](#_Toc122198521)

[Hình 3 Hình ảnh Usecase của người dùng 10](#_Toc122198522)

[Hình 4 hình ảnh usecase của máy 11](#_Toc122198523)

[Hình 5 Hình ảnh lược đồ tuần tự chung của chương trình 17](#_Toc122198524)

[Hình 6 Hình ảnh lược đồ tuần tự của chức năng đăng tải video 18](#_Toc122198525)

[Hình 7 Hình ảnh lược đồ tuần tự của chức năng tạo thông tin cắt ghép 19](#_Toc122198526)

[Hình 8 Hình ảnh lược đồ tuần tự của chức năng yêu cầu xử lý video 20](#_Toc122198527)

[Hình 9 Hình ảnh lược đồ hoạt động của chức năng đăng tải video 21](#_Toc122198528)

[Hình 10 Hình ảnh lược đồ hoạt động của chức năng xử lý video 22](#_Toc122198529)

[Hình 11 Hình ảnh giao diện chung của chương trình 23](#_Toc122198530)

[Hình 12 Hình ảnh giao diện chung của chương trình 23](#_Toc122198531)

[Hình 13 Hình ảnh giao diên chi tiết của giao diện đăng tải nội dung video 25](#_Toc122198532)

[Hình 14 Hình ảnh giao diên chi tiết của giao diện video 26](#_Toc122198533)

[Hình 15 Hình ảnh giao diên chi tiết của giao diện video 26](#_Toc122198534)

[Hình 16 Hình ảnh giao diên chi tiết của giao diện danh sách điểm cắt 29](#_Toc122198535)

[Hình 17Hình ảnh giao diên chi tiết của giao diện ghép nội dung video 30](#_Toc122198536)

[Hình 18 Hình ảnh giao diên chi tiết của giao diện modal setting 31](#_Toc122198537)

# DANH MỤC TỪ VIẾT TẮT

UI: User-Interface

CPU: Central Processing Unit

UX: User-Experience

Web: Website

CORS: Cross-origin resource sharing

Webapp: Web Application

HTML: Hyper-Text Markup Language

CSS: Cascading Style Sheet

JS: JavaScript

BE: Back-End

UC: Use-case

SFolder: Session Folder

# TỔNG QUAN VỀ ĐỀ TÀI

## Lý do chọn đề tài

Trong những năm gần đây, dưới sự phát triển của công nghệ, đặt biệt là mạng xã hội, con người truy cập internet nhằm mục đích kết nối với nhau trở nên nhiều hơn. Có thể nói là nhu cầu này không thể giảm mà chỉ có thể tiếp tục tăng lên, ta có thể dễ dàng nhân ra được việc này thông qua việc các trang tin đưa tin thông qua mạng xã hội này càng nhiều hơn. Trong đó các bài đăng bằng video ngắn ngày càng phát triển, có thể lấy Tiktok ra làm một ví dụ điển hình cho sự phát triển này, theo tin tức của tờ ICTNews cho thấy tỉ lệ người dùng sử dụng ứng dụng này tăng từ 34% (2020) lên 53% (2021); thời lượng sử dụng ứng dụng thì tăng gấp đôi từ 4% lên 8% (2021). Ngoài ra thì có thể thấy được hai ông lớn trong làng công nghệ là Meta và Alphabet (Google) đã cập nhật tính năng chia sẽ video ngắn cho hai nền tản xã hội của họ là Facebook Reel và Youtube Short.

Nắm bắt được xu hướng thị trường, thay vì tạo ra một nền tản mạng xã hội để cạnh tranh với các ông lớn, em đã nghiên cứu phát triển nên một phần mền nói đúng hơn là một web application nhằm mục đích tạo ra các video ngắn để người dùng cơ bản có thể sử dụng và tạo ra các video của riêng họ và đăng tải nó lên các mạng xã hội

## Mục tiêu

* Xây dựng trang web cho phép người dùng chỉnh sửa các video của họ upload lên, như cắt ghép và hiệu chỉnh một vài thông số cơ bản của một video, hướng tới nhóm người dùng mới bắt đầu vào cũng đang bắt đầu quen dần với internet
* Giao diện trang web hướng tới sự đơn giản, dễ thao tác và dễ sử dụng.
* Nghiên cứu về thao tác người dùng được áp dụng để có thể tận dụng tối đa sự quen biết của người dùng

## Công nghệ sử dụng.

* Back-end (server): chương trình sử dụng NodeJS cụ thể hơn là ExpressJS để tạo ra các point để thực hiện các yêu cầu từ client.
* Front-end (client): sử dụng Angular để tạo nên giao diện người dùng.
* Xử lý video: ffmpeg (ganya.dev)

# CƠ SỞ LÝ THUYẾT

## NodeJS

### Giới thiệu.

NodeJS được xây dựng dựa trên JavaScript runtime, nhưng về phần lõi thì NodeJS được viết dựa trên C++ và Javascript. NodeJS là một nền tảng có mã nguồn mở và trực thuộc OpenJS Foundation nơi có sự đóng góp của rất nhiều cái tên lớn như Google, github, IBM, …

### Ưu điểm.

Về ưu điểm như tên gọi của nó JS, nó cho phép người dùng sử dụng JavaScript là ngôn ngữ lập trình chính. Cùng với đó NodeJS cho phép thực hiện các sự kiện bất đồng bộ, từ đó có thể xử lý nhiều việc cùng một lúc. Ngoài ra thì NodeJS sở hữu một lượng lớn các package giúp lập trình viên dễ dàng sử dụng, và ta có thể tìm kiếm các gói package này thông qua trang web npmjs.com

### Nhược điểm.

Có thể nói javascript cũng là một trong những rào cản, vì tuy đơn giản nhưng javascript sẽ trở nên rất phức tạp và khó chịu nếu thực hiện các công việc liên quan tới nhiều CPU. Ngoài ra thì một trong những vấn đề rắc rối đó là callback hell, thứ mà rất nhiều người dùng gặp phải, tuy có rất nhiều phương pháp đồng bộ đã được đưa ra như promise hoặc async nhưng nếu không cẩn thận thì chúng ta vẫn mắc phải những vấn đề này.

### Lý do sử dụng.

Về lý do sử dụng, NodeJS có một cộng đồng người dùng thật sự lớn và cộng đồng sửa lỗi cũng được coi là khá lớn, ngoài ra dùng NodeJS có thể giúp bản thân chương trình được đồng bộ về mặt ngôn ngữ lập trình từ đầu tới cuối, từ đó có thể dễ dàng sửa chửa.

## Angular.

### Giới thiệu.

Angular là một nền tảng và front-end framework để có thể tạo nên các single-page client application sử dụng HTML và Typescript. Với mục đích là hướng tới đa nền tảng, ta có thể dùng angular để tạo ra nhiều thứ như: trang web, mobile web, native mobile và native desktop

### Ưu điểm.

Về ưu điểm, xét về mặt lập trình, angular sử dụng typescript, đây là một ngôn ngữ được dựa trên javascript, nó khắc phục được kha khá các điểm yếu của javascrpit, ngoài ra nó còn có thể giúp code trở nên tường mình hơn. Ngoài ra cấu trúc của một project angular là một ưu điểm, với việc chia các file ra riêng biệt, từ đó việc viết các đoạn code HTML hoặc CSS sẽ trở nên quen thuộc hơn. Về tương lai của Angular, phía sau Angular là Google, nên có thể nói rằng tương lại của angular là có tồn tại, cùng với đó là hằng năm Angular vẫn cập nhật các phiên bản lớn.

### Nhược điểm.

Về nhược điểm, tuy TypeScript là một ngôn ngữ rất tốt, nhưng vì nó là một phiên bản của JavaScript nên việc cập nhật các function mới của JS lên TS sẽ nhiều lúc bị chậm trễ.

### Lý do sử dụng.

Angular được chọn trong project lần này vì đây là một project khá là đặt thù, các thông tin xử lý cần được xử lý tường minh, để tránh bị nhầm với nhiều thứ khác. Có thể nói chương trình này là một phần mềm chỉnh sửa video được tối ưu để sử dụng trên web, vì vậy ta cần một framework front-end hoàn chỉnh và Angular có thể giải quyết được vấn đề này. Ngoài ra em có kinh nghiệm sử dụng Angular, nên sẽ tốt khi có nhiều vấn đề cần phải giải quyết

## FFmpeg.

### Giới thiệu.

FFmpeg là một phần mềm open source cho các tác vụ multimedia. Chương trình này có các thư viện và các phần mềm để có thể thao tác lên video, audio và những têp đa phương tiện và phát sóng trực tiếp.Bộ công cụ này hiện có thể chạy được trên các hệ điều hành lớn như Windows, MacOS và Linux. Được ra đời vào cuối năm 2000, với mục đích chỉ để cắt, ghép video, hiện tại phần mềm này đã có tuổi đời 21 năm, phát triên thêm nhiều tính năng như scaling, compress, filtering, …, trong thời gian phát triển đó có rất nhiều phần mềm nổi tiếng đã sử dụng bộ công cụ này, ta có thể kể tới như VLC (một phần mềm chơi đa phương tiện), hay được sử dụng trong lõi của viêc xử lý video cho Youtube và Bilibili.

### Ưu điểm.

Đầu tiên có thể nhắc đến đây là bộ công cụ mã nguồn mở, đồng nghĩ với việc nó miễn phí và có thể có rất nhiều biến thể từ những lập trình viên khác nhau. Ngoài ra kĩ thuật xử lý video nén của nó có tốc độ nhanh, và theo một số nhận xét thì, tốc độ encode với decode của bộ dụng cụ này nhanh hơn so với các phần mềm chỉnh sửa chuyên nghiệp của Adobe.

### Nhược điểm.

Bộ công cụ này hiện tại không có một UI, nên để tiếp cận và học tập nó tất cả phải thông qua các câu lệnh thực hiện trên commad prompt. Về tư liệu học tập cũng không phổ biến, ngoài các tài liệu được cung cấp bởi chính chủ ra thì gần như rất hiếm các tài liệu viết lại hoặc các bài giảng để việc học tập trở nên nhanh chóng. Việc này có lẽ nằm ở bản chất của chương trình, vì được tạo ra như một nền tảng xử lý nên người dùng trực tiếp trở nên quá ít, dẫn tới cộng đồng trở nên nhỏ và các câu hỏi đặc ra được ít trả lời.

### Lý do sử dụng.

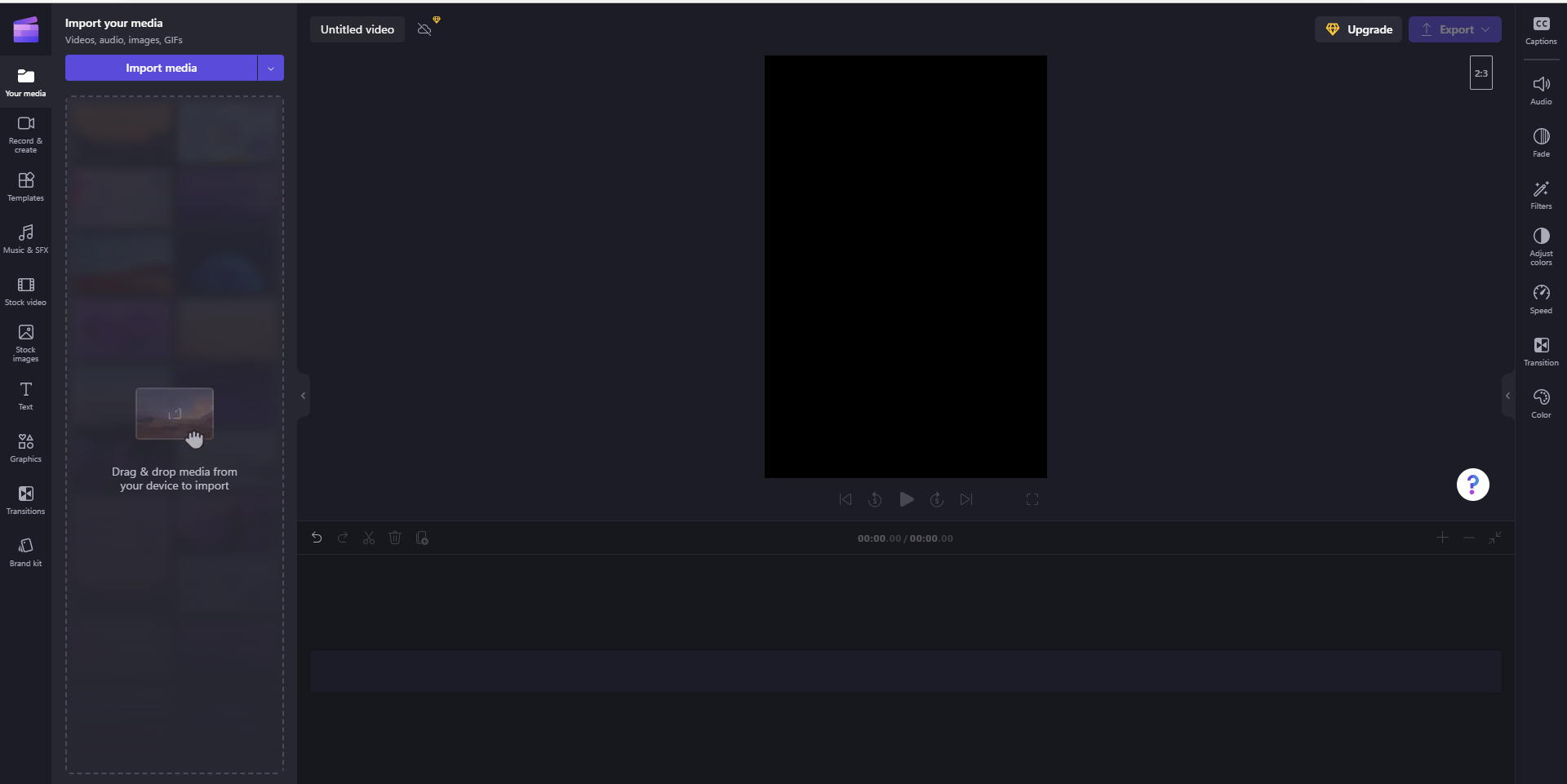
Như đã nhắc tới ở trên thì đây là bộ công cụ open source, từ đó có thể dễ dàng sử dụng và phát triển nó không cần quan tâm tới các vấn đề liên quan tới bản quyền chương trình. Ngoài ra như đã nhắc ở trên tốc độ xử lý là một trong những yếu tố được cân nhắc, vì phải xử lý thông qua server, từ đó giảm thời gian chờ của người dùng cũng là một điểm cần được chú ý.

# KHẢO SÁT HIỆN TRẠNG VÀ XÁC ĐỊNH YÊU CẦU

## Khảo sát hiện trạng

### Trang web Clipchamp.com

Trang web Clipchamp là một trang web cho phép người dùng đưa video của bản thân lên trang web này và thực hiện chỉnh sửa video gồm các thao tác cơ bản. Trang web này được phát triển triển và phát hình bởi đội ngũ của Microsoft, ngoài ra phiên bản desktop của trang web này được Microsoft phát hành kèm chung với Windows 11 để trở thành một trong những công cụ có sẵn của phiên bản windows này. Có thể thấy vì là được mới phát hành gần đây nhưng Clipchamp đã có một lượng người dùng tương đối và để lại đánh giá khá tốt tại thang điểm của google search.



Hình Hình anh chương trình clipchamp tại trang clipchamp.com

Ưu điểm

* Chương trình được thiết kế với phiên bản web dễ dàng tiếp cận.
* Giao diện hiện đại, có sự giao thoa giữa ngôn ngữ thiết kế của Windows 11 và Material Design.
* Các thao tác quen thuộc, khá giống với các chương trình chỉnh sửa video trên desktop.
* Có nhiều template được dựng sẵn để người dùng sử dụng.
* Phiên bản miễn phí khá đầy đủ tính năng

### Phần mềm DaVinci Resolve

Phần mềm DaVinci Resolve là chương trình chỉnh sửa video chuyên nghiệp được BlackmagicDesign, một thương hiện sản suất ra các máy quay chuyên nghiệp, thực hiện và phát hành. Có thể nói đây là một trong những phần mềm cực kỳ nổi tiến trong cộng đồng editor, vì phiên bản miễn phí được công ti này trực tiếp phát hành và phiên bản miễn phí này cung cấp một lượng lớn công cụ mạnh mẽ để thực hiện các thao tác chỉnh sửa.

A screenshot of a computer

Description automatically generated with medium confidence

Hình Hình anh phần mềm DaVici Resolve 18

Ưu điểm:

* Phần mềm được phát hành miễn phí
* Bộ công cụ đa dạng và mạnh mẽ để thực hiện các thao tác chuyên nghiệp.
* Hệ thống cân chỉnh màu sắc cực kì mạnh mẽ, gồm nhiều thông số chuyên biệt.

## Xác định yêu cầu.

### Nghiệp vụ

#### Đối tượng sử dụng

* Người mới bắt đầu làm quen với chỉnh sửa
* Người muốn cắt ghép video, nhưng gặp giới hạn về hiệu năng
* Người không chuyên, không có nhu cầu cao về chức năng chỉnh sửa

#### Chức năng

* Cắt video
* Ghép video
* Thay đổi kích thước khung ảnh
* Thay đổi tỉ lệ khung ảnh.
* Thay đổi các thông số video
* Thay đổi tốc độ phát của video
* Đổi tên mở rộng cho video

## Đánh giá và lựa chọn công nghệ

### Mục tiêu đề tài

Xây dựng một website cho phép người dùng đăng tải các video lên và thực hiện các thao tác chỉnh sửa video như cắt ghép và chỉnh sửa các thông số cơ bản. Với mục tiêu là tận dụng toàn bộ giới hạn của web-design để tạo ra các thao tác tương đồng với các thao tác chuyên nghiệp.

### Nội dung

Tìm hiểu lý thuyết và giới hạn về các công nghệ trong việc thực hiện đề tài:

* Tìm hiểu về NodeJS và Express và giới hạn tương tác giữ server và client
* Tìm hiểu về Angular về lý thuyết và giới hạn có thể thực hiện.
* Tìm hiểu về công cụ cho phép thực hiện can thiệp vào các tệp đa phương tiện: FFmpeg

Phân tích yêu cầu và hiện thực hóa chương trình:

* Thiết kế các lượt đồ
* Mô tả chức năng
* Thiết kế giao diện
* Xử lý dữ liệu tương tác giữ client và server

### Môi trường lập trình

Về môi trường lập trình, cả front-end và back-end đề sử dụng phần mềm code editor Visual Studio Code để phát triển, tổ chức vào quản lý code.

Trong đó Visual Studio Code là một code editor miễn phí và phát triển trên đa nền tảng (Microsoft Windows, MacOS, Linux) được phát triển và phát hành bởi Microsoft. Code Editor này có thể coi như là một phiên bản nhỏ gọn và hợp xu hướng của Visual Studio. Có thể nói là code editor miễn phí có số lượng người dùng lớn nhất và được nhiều người nghĩ tới đầu tiên khi muốn coding.

### Phương pháp nghiên cứu

Nghiên cứ về lý thuyết khi ứng dụng Angular và NodeJS làm cốt lõi của chương trình, từ đó đưa ra giới hạn hiện tại của bản thân và công nghệ sắp xếp lại các tính năng

Mô hình hóa và tham chiếu các chương trình đã có sẵn như DaVinci Resolve, capcut, Adobe Premire từ đó sử dụng HTML5 triệt để để giả lập lại các thao tác tương tự

Tham khảo ý kiến từ cộng đồng, giáo viên hướng dẫn, bạn bè cùng với kinh nghiệm lập trình bản thân hấp thụ được trong thời gian học.

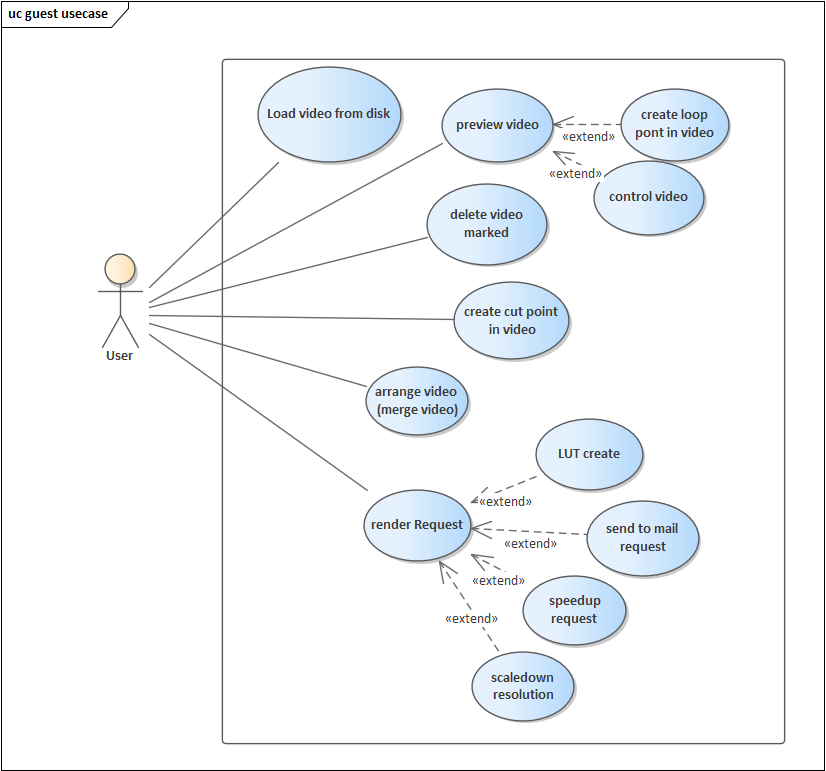
### Ý nghĩa khoa học, thực tiễn

Về khoa học, chương trình có thể tận dụng tất cả khả năng của HTML5, CSS, JavaScript để tạo nên nền giao diện thao tác chuyên nghiệp. Từ đó nghiên cứu được khả năng đích thực trong thời điểm này

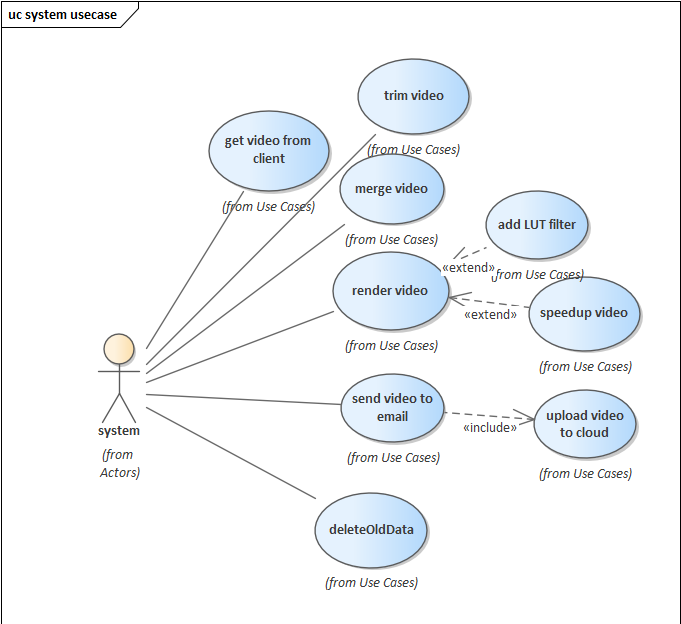
Về thực tiễn, nghiên cứu được các thao tác cơ bản của người dùng từ đó tìm hiểu và điều chỉnh giao diện phù hợp với thói quen thao tác.

## Mô hình hóa hệ thống

### Sơ đồ usecase



Hình Hình ảnh Usecase của người dùng



Hình hình ảnh usecase của máy

### Mô tả sơ đồ usecase

Bảng 1: Bảng mô tả nội chi tiết dung sơ đồ usecase

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Chức năng | Mô tả trên hình | Mô tả chi tiết |
| UC\_1 | Đăng tải video | Load video from disk | Load video từ ổ đĩa lên trên trang web, đồng thời gửi video lên trên server |
| UC\_2 | Tạo ra các điểm cắt video | Create cut point in video | Tạo ra các điểm cắt từ đó có thể đưa ra thông số cho server render |
| UC\_3 | Xóa các điểm đánh dấu | Delete video marked | Xóa các điểm đánh dấu các đoạn cắt |
| UC\_4 | Xem lại các video đã upload lên | Preview video | Xem lại các video đã chọn, từ đó dễ dàng đưa ra quyết định về các điểm cắt của video sau cùng, và xem lại kết quả cuối cùng |
| UC\_5 | Tạo vị trí lập lại | Create loop point in video | Tạo ra vị trí lập lại trên video, khi phát video thì video sẽ chỉ lập lại đoạn được đánh dấu |
| UC\_6 | Điều khiển video | Control video | Điều khiển video phát, dừng, hoặc lặp đi lặp lại, … |
| UC\_7 | Sắp xếp lại các video | Arrange video | Sắp xếp lại các đoạn cắt từ đó đưa ra đoạn video hoàn chỉnh, thứ tự của các video được sắp xếp là thứ tự của các video sau khi xếp lại |
| UC\_8 | Tạo màng lọc hình ảnh | LUT create | Tạo ra lớp màng giả lập tại client để người nhìn có thể thay đổi nhóm màu sắc trên của video thông qua vòng tròn HUE |
| UC\_9 | Yêu cầu tăng tốc độ video | Speedup request | Yêu cầu server render ra video có tốc độ tằng lên so với tốc độ đầu vào của video |
| UC\_10 | Giảm dộ phân giải | Scaledown resolution | Yêu cầu server render ra video có độ phân giải giảm nhỏ hơn so với ban đầu |
| UC\_11 | Yêu cầu gửi video qua mail | Send to email request | Yêu cầu server sau khi render xong video gửi đoạn video đó vào mail cá nhân. |
| UC\_12 | Gửi yêu cầu render | Render request | Yêu cầu server render video theo mô tả video được sắp xếp, trong đó gồm các thông số thuộc tính của video |
| UC\_13 | Cắt video | Trim video | Cắt các video theo yêu cầu của người dùng, được đánh dấu bằng điểm bắt đầu và kết thúc |
| UC\_14 | Ghép video | Merge video | Nối các video lại theo thứ tự đã được đề xuất từ yêu cầu render từ client |
| UC\_15 | Nhận video từ client | Get video from client | Hoạt động chung UC\_1, chức năng sẽ nhận video và trả về tên video được đánh dấu, đồng thời là nhận dạng cho lượt hoạt động |
| UC\_16 | Tăng tốc video | Speed up video | Tăng tốc video lên theo yêu cầu từ render request |
| UC\_17 | Render video | Render video | Render video theo yêu cầu được gửi từ client, thông qua các thông số gửi kèm |
| UC\_18 | Xóa dữ liệu cũ | deleteOldData | Xóa các dữ liệu cũ tồn đọng trên server |
| UC\_19 | Tạo lớp màng lên video | Add LUT filter | Server tạo ra một lớp lọc đê thay đổi màu sắc trên video theo hệ vòng tròn HUE |
| UC\_20 | Đăng tải video lên cloud | Upload video to cloud | Hệ thống sẽ đưa video lên cloud để lưu trữ lại video với mục đích làm nguồn dẫn khi gửi video thông qua mail |
| UC\_21 | Gửi video qua mail | Send video to mail | Video sau khi dược render xong, server sẽ gửi video kết quả lên cloud để lưu trữ lại vài gửi video này thông qua mail để đảm bảo dung lượng mail khi gửi. |

## 

## Đặc tả Usecase

### Usecase đăng tải video

Bảng 2 Bảng mô tả chi tiết usecase đăng tải video

|  |  |
| --- | --- |
| Usecase đăng tải video | |
| Mô tả | Đây là chức năng khi người dùng load một video lên chương trình từ đó chương trình tạo ra một bản ghi tại client song song đó gửi đoạn video lên server |
| Tác nhân kích hoạt | User |
| Tiền điều kiện | Không |
| Các bước thực hiện | 1. Bấm vào nút upload video 2. Chọn các video cần thao tác 3. Nhấn đăng tải |
| Thành công | Chương trình cập nhật đoạn video lên giao diện đăng tải |
| Thất bại | Không thể hiện nội dung bị lỗi lên giao diện |

### Usecase tạo điểm cắt video

Bảng bảng mô tả chi tiết usecase tạo điểm cắt video

|  |  |
| --- | --- |
| Usecase tạo điểm cắt video | |
| Mô tả | Chức năng hỗ trợ người dùng tạo ra các điểm cắt của từng đoạn video, tạo nên các đánh dấu, tạo cảm giác như cắt các đoạn cắt |
| Tác nhân kích hoạt | User |
| Tiền điều kiện | Có video được đăng tải thành công lên giao diện |
| Các bước thực hiện | 1. Chọn một video đã được đăng tải lên giao diện 2. Chọn vào chức năng tạo các điểm cắt video 3. Chọn ra hai điểm đánh dấu đầu vào cuối của video 4. Bấm chọn tạo đánh dấu |
| Thành công | Điểm dánh dấu sẽ xuất hiện tại giao diện danh sách điểm cắt |
| Thất bại | Không thể hiện đoạn cắt tại giao diện danh sách điểm cắt |

### Usecase gửi yêu cầu render.

Bảng bản mô tả chi tiết usecase yêu cầu render.

|  |  |
| --- | --- |
| Usecase gửi yêu cầu render video | |
| Mô tả | Chức năng thao tác tại client giúp người dùng gửi yêu cầu render video lên server từ đó server trực tiếp render ra video theo yêu cầu |
| Tác nhân kích hoạt | User |
| Tiền điều kiện | Có video tại giao diện sắp xếp vị trí video |
| Các bước thực hiện | 1. Bấm chọn vào nút yêu cầu render. 2. Chọn khung hình cho video 3. Chọn tốc độ cho video 4. Bấm chọn gửi yêu cầu |
| Thành công | Màn hình loading khởi động, sau khi video đã được chuẩn bị hoàn tất, video sẽ được gửi trực tiếp về máy theo phương thức tải xuống |

### Usecase nhận video từ client

Bảng bảng mô tả chi tiết usecase nhận video từ client

|  |  |
| --- | --- |
| Usecase nhập video từ client | |
| Mô tả | Video được nhận từ client, đánh dấu lại video vừa được tải lên. |
| Tác nhân kích hoạt | Server |
| Tiền điều kiện | Có video được gửi từ client |
| Các bước thực hiện | 1. Bắt video được gửi từ client 2. Tạo ra unique session id 3. Tạo folder cho video được gửi lên theo session id 4. Copy video vào thư mục vừa tạo 5. Gửi lại thông tin về unique id về client, để nhận dạng đánh dấu |
| Thành công | Loading icon tại client sẽ được tắt đi |

### Usecaser ghép video

Bảng bảng mô tả chi tiết usecase ghép video

|  |  |
| --- | --- |
| Usecase ghép video | |
| Mô tả | Chức năng ghép video theo từng đôi một |
| Tác nhân kích hoạt | Server |
| Tiền điều kiện | Thông tin trong yêu cầu cắt có ít nhất hai dữ kiện |
| Các bước thực hiện | 1. Nhận thông tin từ các yêu cầu 2. Lấy thông tin về hai video cần ghép lại 3. Tạo ra tên giả cho video được ghép lại 4. Xác định yêu cầu sau khi ghép hai video lại 5. Thực hiện ghép hai video theo yêu cầu 6. Gửi lại tên giả của video lại cho vị trí yêu cầu |
| Thành công | Chương trình cập nhật đoạn video lên giao diện đăng tải |
| Thất bại | Không thể hiện nội dung bị lỗi lên giao diện |

### Usecase render video.

Bảng bảng mô tả chi tiết usecase render video.

|  |  |
| --- | --- |
| Usecase render video | |
| Mô tả | Server nhận yêu cầu từ client, dựa vào các thông tin đặt ra |
| Tác nhân kích hoạt | Server |
| Tiền điều kiện | Nhận được yêu cầu render từ client |
| Các bước thực hiện | 1. Nhận thông tin từ yêu cầu 2. Phân tích thông tin từ yêu cầu 3. Tạo ra các đoạn cắt như yêu cầu đề ra 4. Ghép nối video theo từng đôi một theo yêu cầu 5. Xử lý các yêu cầu phụ từ client 6. Gửi video được render hoàn thành về client |
| Thành công | Video sẽ được gửi trực tiếp từ server về client dưới hình thức download |

### Usecase addLUTFilter

Bảng bảng mô tả chi tiết usecase addLUTFilter

|  |  |
| --- | --- |
| Usecase addLUTFilter | |
| Mô tả | Đây là chức năng để cho server render bộ màu sắc cho video |
| Tác nhân kích hoạt | Server |
| Tiền điều kiện | Có được tên của giai đoạn trước đó thực hiện |
| Các bước thực hiện | 1. Kiểm tra thông tin từ request 2. Phân tích thông tin từ yêu cầu 3. Sắp xếp các thuộc tính lại. 4. Thực hiện việc thêm bộ lọc màu 5. Gửi lại tên giả của video tại vị trí yêu cầu |
| Thành công | Dữ liệu về tên của video kết quả |

### Usecase send video to e-mail.

Bảng bảng mô tả chi tiết usecase send video to email.

|  |  |
| --- | --- |
| Usecase send video to email | |
| Mô tả | Đây là chức năng để cho server sau khi render sẽ gửi video đã hoàn thành vào mail của người dùng |
| Tác nhân kích hoạt | Server |
| Tiền điều kiện | Video kết quả được render xong |
| Các bước thực hiện | 1. Kiểm tra thông tin từ request 2. Lấy email từ yêu cầu được giử từ ban đầu 3. Kiểm tra thông tin của video kết quả 4. Gửi video kết quả lên cloud và nhận về liên kết 5. Lấy địa chỉ liên kết đưa vào nội dung mail 6. Thiết lập email gửi 7. Gửi email |
| Thành công | Người dùng nhận được email từ hệ thống, trong đó có liên kết tới video thực hiện |

### Usecase xóa dữ liệu cũ

Bảng bảng mô tả chi tiết usecase xóa dữ liệu cũ

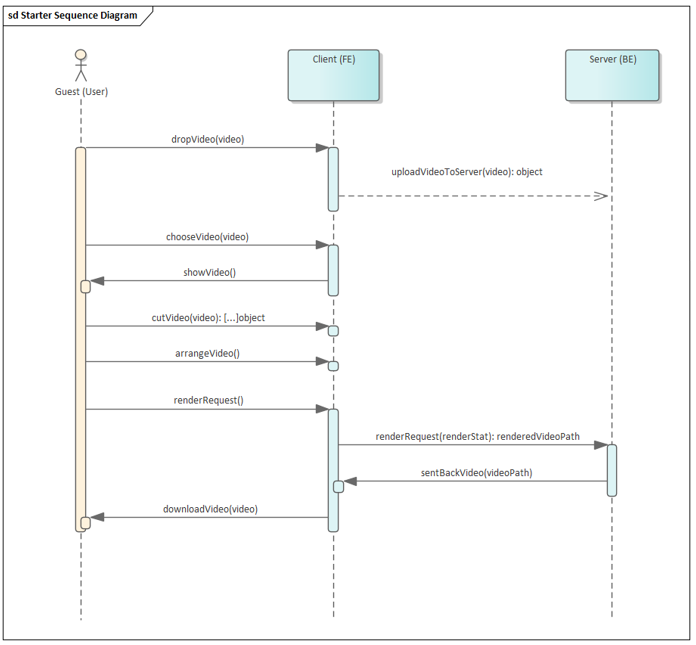
|  |  |
| --- | --- |
| Usecase xóa dữ liệu cũ | |
| Mô tả | Đây là chức năng để cho server giảm bớt dung lượng trên server |
| Tác nhân kích hoạt | Server |
| Tiền điều kiện | Không |
| Các bước thực hiện | 1. Kiểm tra thời gian hoạt động chương trình 2. Load toàn bộ thông tin của các session folder 3. Kiểm tra thời gian chỉnh sửa của SFolder 4. Xóa folder có thời gian tồn đọng trên 7 ngày |
| Thành công | Dữ liệu về các folder |
| Thất bại | Không biểu hiện |

# THIẾT KẾ WEBSITE

## Thiết kế hệ thống

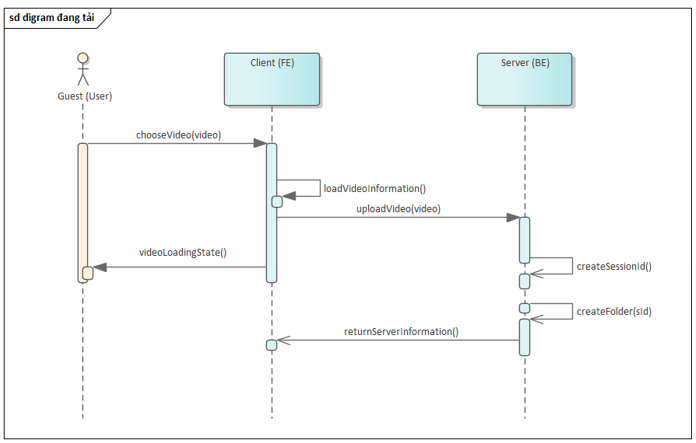
### Lược đồ

#### Lược đồ tuần tự



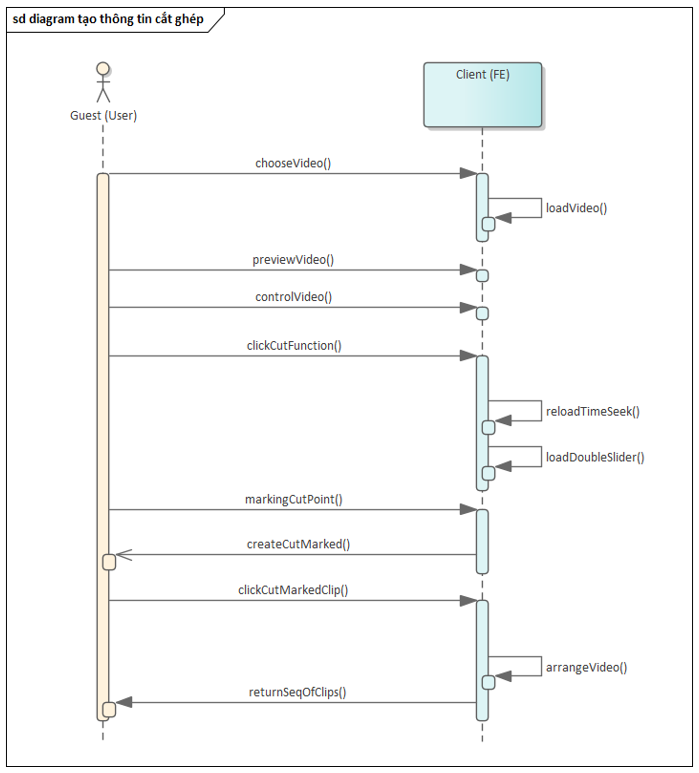
Hình Hình ảnh lược đồ tuần tự chung của chương trình

##### Lược đồ tuần tự đăng tải video



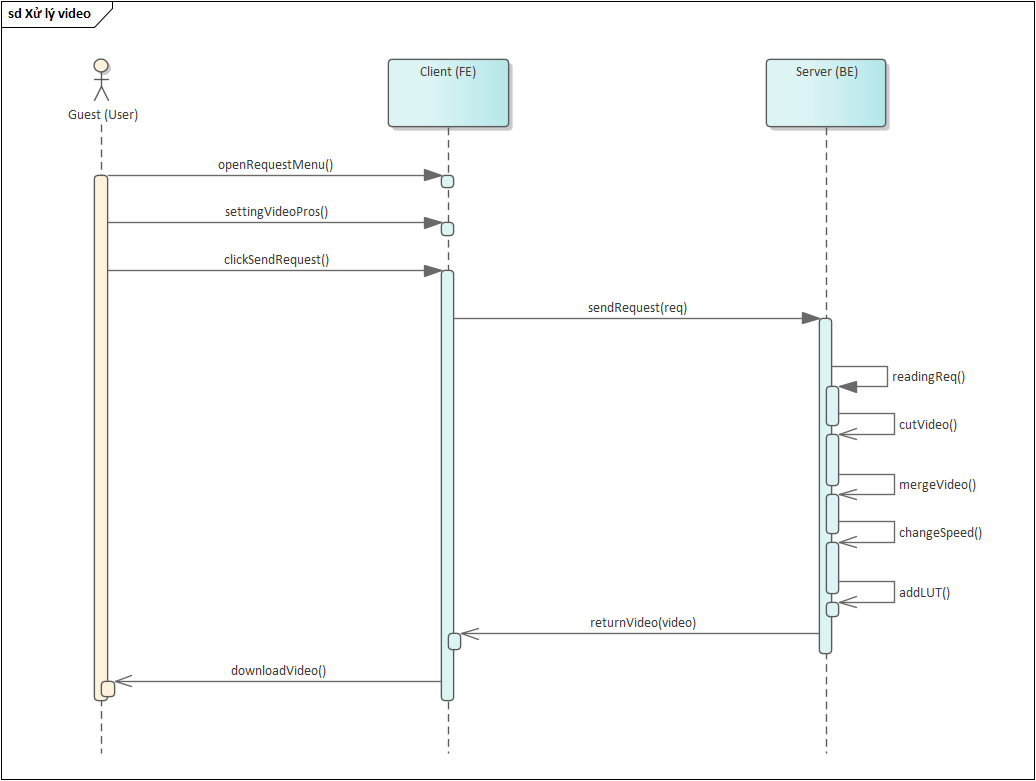
Hình Hình ảnh lược đồ tuần tự của chức năng đăng tải video

##### Lược đồ tuần tự tạo điểm cắt và sắp xếp video (ghép video tại client)



Hình Hình ảnh lược đồ tuần tự của chức năng tạo thông tin cắt ghép

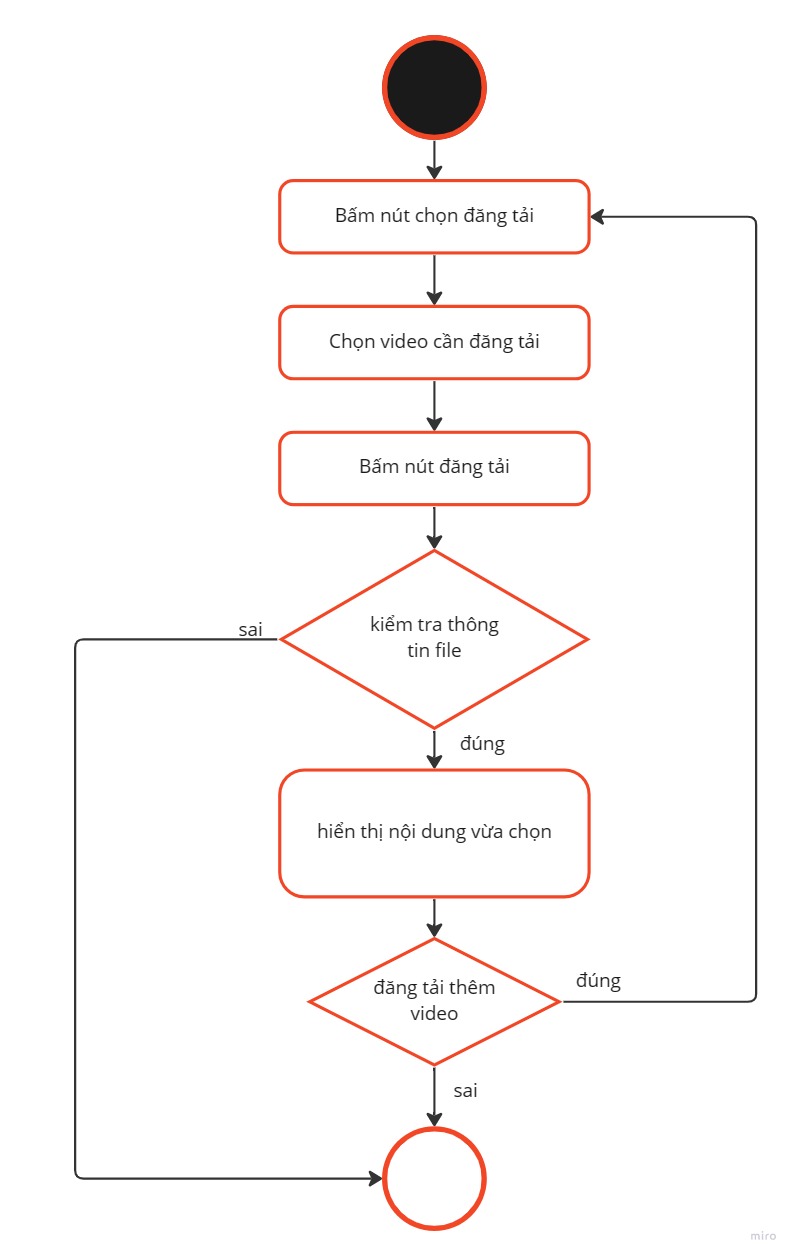
##### Lược đồ tạo yêu cầu xử lý video



Hình Hình ảnh lược đồ tuần tự của chức năng yêu cầu xử lý video

#### Lược đồ hoạt động

##### Lược đồ đăng tải video



Hình Hình ảnh lược đồ hoạt động của chức năng đăng tải video

##### Lược đồ xử lý video

A picture containing text, diagram, screenshot, font

Description automatically generated

Hình Hình ảnh lược đồ hoạt động của chức năng xử lý video

### Tổng quát chương trình

#### Tổng quát tại Client

Bảng Bảng giới thiệu về các thư viện sử dụng chung với Angular 14 tại Frontend

|  |  |  |
| --- | --- | --- |
| STT | Tên thư viện | Lý do sử dụng |
| 1 | Angular 14 Framework pack | Đây là framework frontend hoàn chỉnh, có mọi thứ người dùng cần, trong đó, từ việc định hướng cho tới tích hợp các gói animation hỗ trợ giao diện |
| 2 | File saver | Là một thư viện hỗ trợ việc download các file được gửi từ server |
| 3 | Angular2 uuid | Là thư viện giúp tạo ra các id được dùng để hỗ trợ việc xác định các object trong mảng |

#### Tổng quát tại Server

Các thư viện sử dụng.

Bảng Bảng giới thiều về các thư viện được sử dụng với NodeJS tại Backend

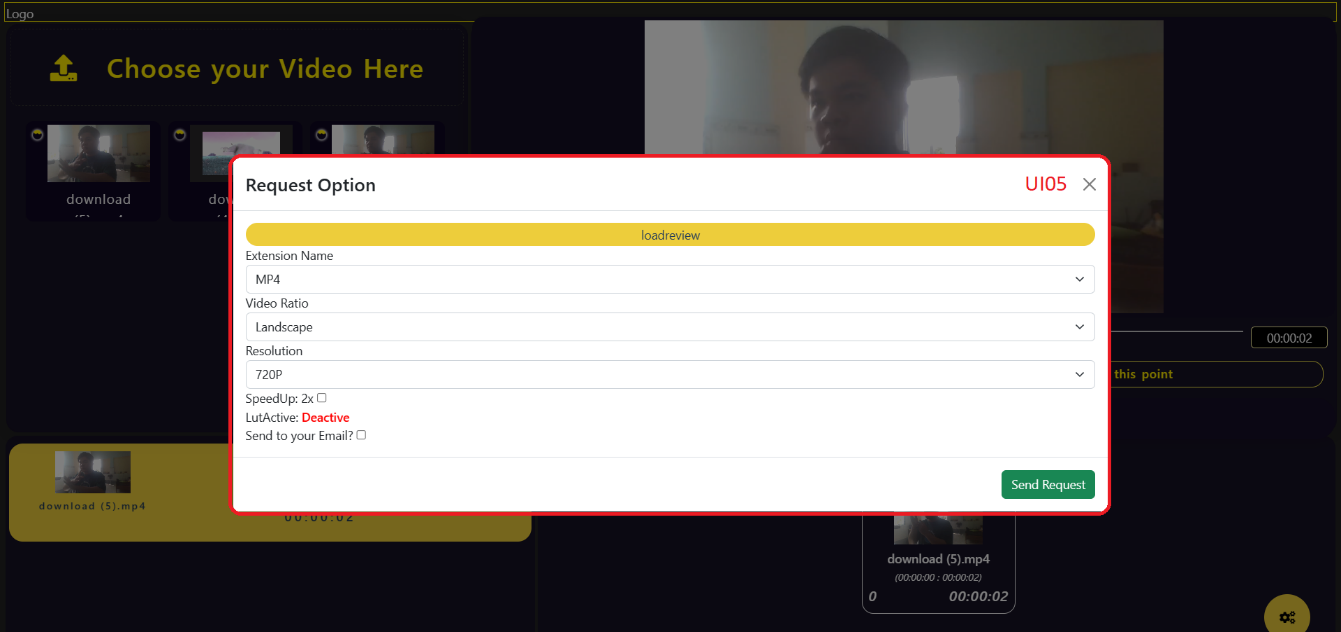
|  |  |  |
| --- | --- | --- |
| STT | Tên thư viện | Lý do sử dụng |
| 1 | Express | Uu điểm: đây là framework hướng tới sự tối giản nên nó sẽ cực kỳ nhẹ và nhanh. Ngoài ra việc tạo các Endpoint bằng framework này khá dễ dàng, cùng với đó ta có thể đặt các middleware vào API cũng rất dễ dàng. Ngoài ra nhóm phát triển cũng thuộc OpenJS Foundation nên về bản chất ExpressJS sinh ra là cho NodeJS |
| 2 | Nodemon | Đây là một thư viện hỗ trợ cho việc phát triển phần mềm, mỗi lần ta save lại nó sẽ tự động restart lại project từ đó có thể rút ngắn được thời gian lập trình |
| 3 | Body parser | Là một middleware làm tối giản lại các dữ liệu được gửi thông qua body trên request được gửi từ client |
| 4 | Cors | Là middleware giúp các tài nguyên được chia sẻ chéo. Giúp client và server có thể giao tiếp với nhau và đảm bảo được lỗi bảo mật CORS |
| 5 | Dotenv | Là thư viện để tạo ra các biến ảo để thiết lập môi trường |
| 6 | Multer | Là thư viện hỗ trợ việc chuyền dẫn các file giữa client và server |
| 7 | Short unique id | Là một thư viện tạo ra các định danh độc nhất ngắn, giúp phân biệt giữ các file được sinh ra trong một session |
| 8 | Nodemailer | Là thư viện giúp người dùng gửi mail một cách tự động, ngoài ra thư viện này hỗ trợ rất nhiều các dịch vụ mail thông dụng để dễ dàng thực hiện gửi mail |
| 9 | Cloudinary | Là một thư viện hỗ trợ môi trường giao tiếp với dữ liệu được lưu thê hệ thống của Cloudinary |

## Thiết kế giao diện

### Danh sách giao diện



Hình Hình ảnh giao diện chung của chương trình



Hình Hình ảnh giao diện chung của chương trình

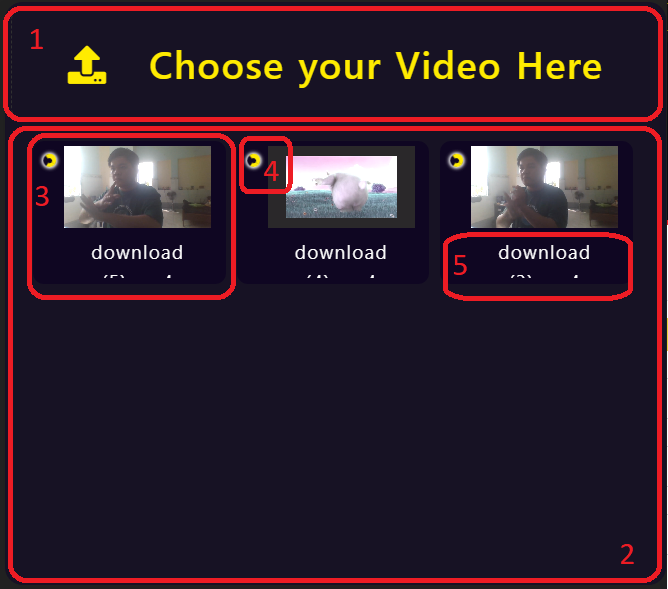
Bảng bảng mô tả giao diện chung

|  |  |  |
| --- | --- | --- |
| STT | Tên giao diện | Mô tả |
| UI01 | Đăng tải nội dung video | Giao diện sinh ra với mục đích tạo nên vị trí để người dùng đưa nội dung của họ lên chỉnh sửa |
| UI02 | Video | Giao diện để người dùng có thể xem lại các video, cùng với đó là khu vực tạo nên các điểm cắt. |
| UI03 | Danh sách các điểm cắt video | Phần giao diện thể hiện các điểm cắt video mà người dùng đã tạo nên |
| UI04 | Ghép nội dung video | Giao diện thể hiện các nội dung và vị trí ghép lại của các nội dung được đánh dấu |
| UI05 | Modal setting | Giao diện điều chỉnh lại một số thông tin liên quan tới video cuối cùng |

### 

### Mô tả chi tiết giao diện

#### Giao diện đăng tải nội dung video



Hình Hình ảnh giao diên chi tiết của giao diện đăng tải nội dung video

Bảng bảng mô tả chi tiết giao diện đăng tải nội dung video

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên đối tượng | Loại | Mô tả |
| 1 | Input video | Thẻ INPUT | Khu vực để ta có thể chọn video từ trong máy tính, hoặc ta có thể kéo thả các video |
| 2 | Video wraper | Thẻ Div | Khu vực chứa các video được upload lên |
| 3 | Video item | Thẻ button | Thể thể hiện video được upload lên, cùng với đó là thumbnail của video đó. |
| 4 | Video loading | Thẻ span | Thể hiện video đã được gửi lên server hay chưa, khi thẻ này còn xuất hiện thì video vẫn đang được tải lên server |
| 5 | Video name | Thẻ div | Là khu vực thể hiện tên của video được tải lên video |

#### Giao diện video



Hình Hình ảnh giao diên chi tiết của giao diện video



Hình Hình ảnh giao diên chi tiết của giao diện video

Bảng bảng mô tả chi tiết giao diện video

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên đối tượng | Loại | Mô tả |
| 1 | Video player | Thẻ video | Đây là khu vực phát video sau khi được chọn tại giao diện đăng tải nội dung video, trạng thái ban đầu của video sẽ là tạm dừng |
| 2 | Timer | Thẻ div | Đây là vị trí thể hiện video đang được phát với đâu theo số học, định dạng của khu vực này là giờ, phút, giây |
| 3 | Time Seeker | Thẻ Input[slider] | Đây là khu vựng thể hiện thời gian video đang được phát tới đâu theo phương thức hình học |
| 4 | Stop button | Thẻ button | Đây là nút để bấm dừng video lại, video sẽ được tự chuyển vể vị trí khung ảnh đầu tiên |
| 5 | Play/ Pause button | Thẻ button | Đây là nút làm cho video có thể tạm dừng phát tại vị trí đang phát, và phát tiếp tục tại vị trí đang phát. |
| 6 | Previous button | Thẻ button | Đây là nút làm cho video tua ngược lại vị trí 5 giây trước tại vị trí đang được phát |
| 7 | Next button | Thẻ button | Đây là nút làm cho video tua tới 5 giây tại vị trí đang được phát |
| 8 | Cut function button | Thẻ button | Nút làm chuyển đổi thanh time seeker trở thành thanh đánh dấu vị trí. |
| 9 | Mute button | Thẻ button | Đây là nut giúp người dùng tắt tiếng nhanh chóng |
| 10 | Volume slider | Thẻ input[slider] | Là một thanh slider cho phép người dùng kéo, điều chỉnh âm lương của video đang phát |
| 11 | Start marker | Thẻ input | Là vị trí thể hiện điểm bắt đầu đánh dấu, từ đó có thể xác định vị trí ban đầu của các điểm cắt hoặc lập lại |
| 12 | Double slider | Thẻ input[slider] | Đây là thẻ hỗn hợp, được thiết lập từ hai thanh slider. Mục đính là để người dùng có thể chọn vị trí ban đâu và kết thúc một cách thuận tiên |
| 13 | End marker | Thẻ input | Là vị trí thể hiện điểm kết thúc theo định dạng thời gian, mục tiêu là giúp người dùng có thể nhìn thấy chính xác thời điểm được đánh dấu kết thúc |
| 14 | Cut button | Thẻ button | Là nút tạo ra các điểm cắt, các điểm cắt, các điểm này sẽ tạo thành từng item riêng biệt và sẽ được gửi sang giao diện danh sách cắt |
| 15 | Loop button | Thẻ button | Là nút để tạo ra vị trí lập đi lập lại, sau đó khi phát đoạn video thì nó sẽ tự động lặp lại đoạn thời gian đã chọn |

#### Giao diện danh sách các điểm cắt

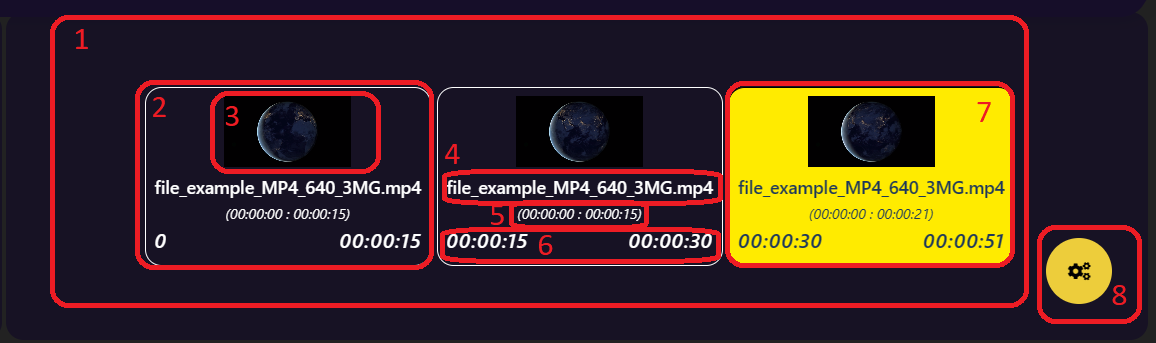


Hình Hình ảnh giao diên chi tiết của giao diện danh sách điểm cắt

Bảng bảng mô tả chi tiết giao diện danh sách các điểm cắt

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên đối tượng | Loại | Mô tả |
| 1 | Marked cut item | Thẻ div | Là thẻ chứa đựng và thể hiện các thông tin về các điểm cắt |
| 2 | Video thumbnail | Video | Là vị trí thể hiện thmbnail của video từ đó có theerr dễ dàng xác định video |
| 3 | Video name | Div | Vị trí thể hiện tên của video được tạo điểm cắt |
| 4 | Cut point | Div | Là vị trí thể hiện thời gian mà điểm cắt tạo ra, bới điểm đầu (thông số bên phải), và điểm kết thúc (thông số bên trái) |
| 5 | Delete button | Thẻ button | Là nút để có thể xóa khỏi danh sách điểm cắt |
| 6 | Hover div | Thẻ div | Là hình ảnh của đối tượng 1 khi giữ chuột trên thẻ, tạo thành event hover |

#### Giao diện ghép nội dung video

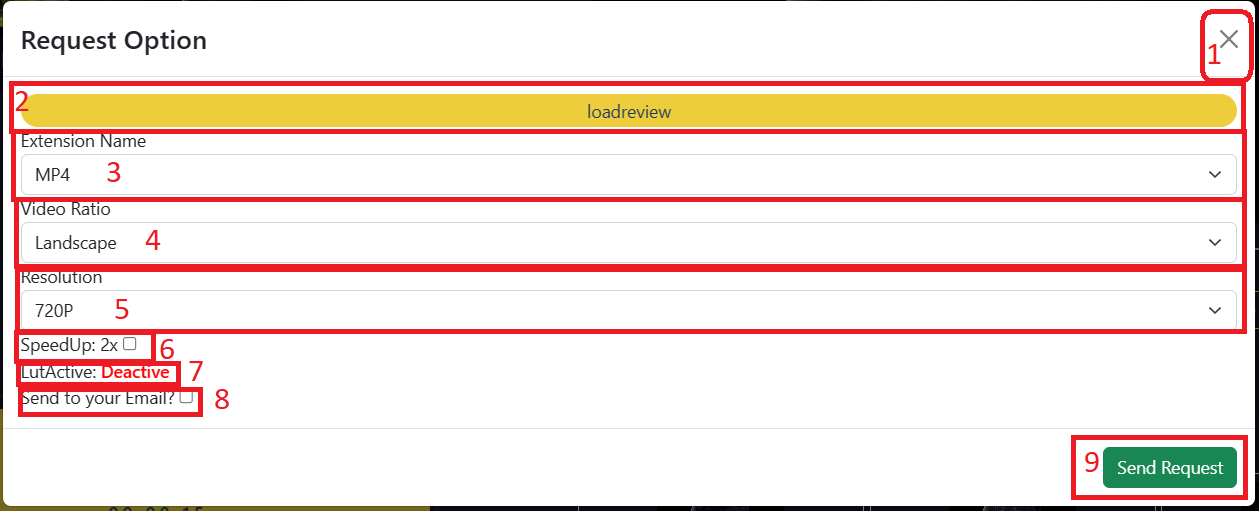


Hình Hình ảnh giao diên chi tiết của giao diện ghép nội dung video

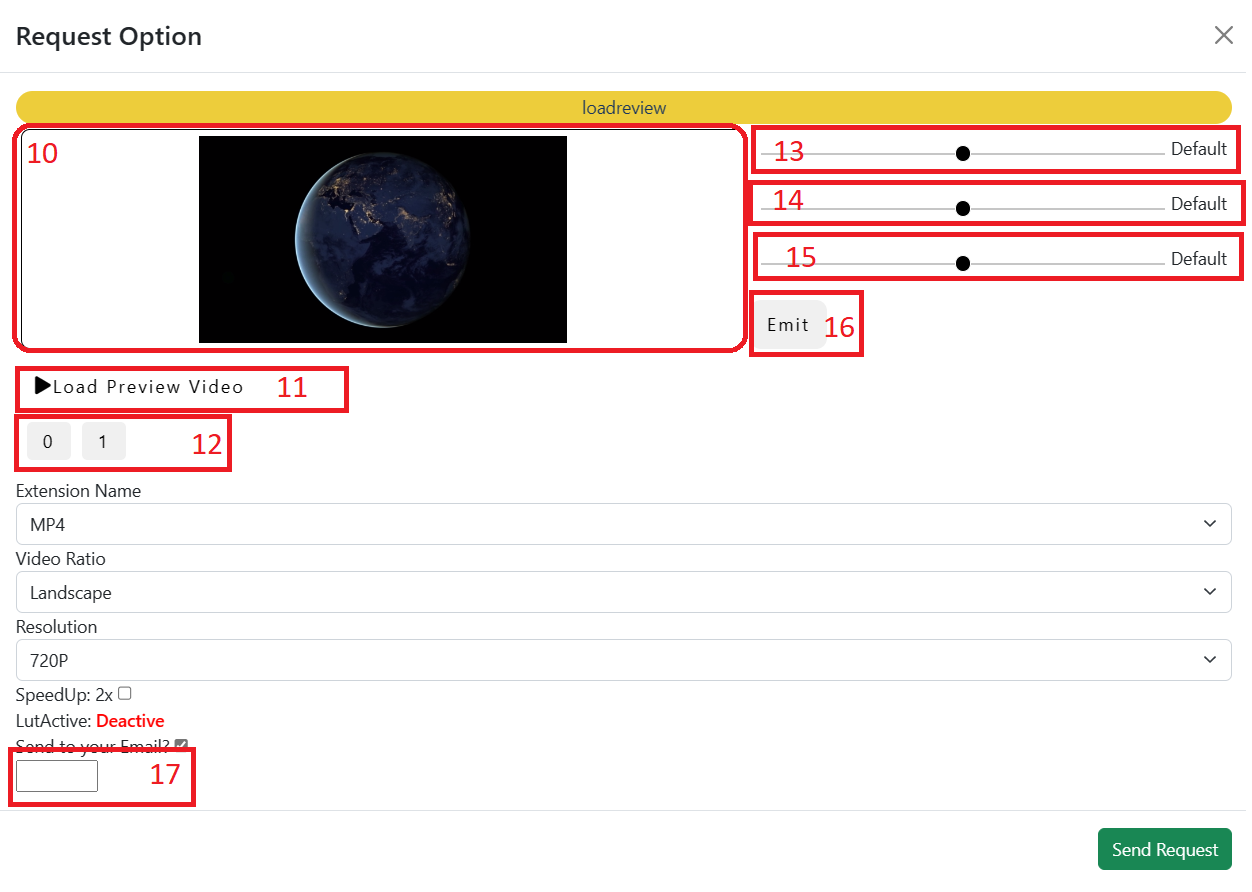
Bảng bảng mô tả chi tiết giao diện ghép nội dung video

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên đối tượng | Loại | Mô tả |
| 1 | Arrange video wraper | Thẻ Div | Là khung bao trọn lại toàn bộ các video đã được sắp xếp, khung này sẽ tự canh giữa và mở rộng theo chiều ngang, từ đó sẽ giúp người dùng dễ chú ý hơn. |
| 2 | Video item | Thẻ div | Là khung chứa bao lại toàn bộ các thông tin của một video và nó chia cắt các video ra, từ đó ta có thể dễ dàng xác định được vị trí của từng video, ngoài ra ta có thể nhấn vào tag này để xóa đi thông tin trong thẻ wraper |
| 3 | Video thumbnail | Thẻ video | Đây là thẻ video được thiết kế để nó trở thành một ảnh thumbnail cho vị trí video cắt. |
| 4 | Video name | Thẻ div | Là vị trí thể hiện tên của đoạn video, đây trở thành một phương thức xác định đúng video cần thay đổi |
| 5 | Video point | Thẻ div | Là vị trí thể hiện đoạn video đó được cắt từ đâu tới đâu. Phía dãy số bên trái thể hiện điểm bắt đầu, còn dãy bên phải là hiện kết thúc |
| 6 | Video duration | Thẻ div | Là vị trí thể hiện thời gian đoạn video xuất hiện trong video được render, nó thể hiện thời gian đoạn cắt đó được bắt đầu và thời gian đoạn cắt đó kết thúc |
| 7 | Video item hover | Thẻ div hover | Thể hiện event hover của thẻ video item khi người dùng đưa chuột lại gần tạo ra highlight để người người phân biệt thẻ đang được chọn |
| 8 | Render option button | Thẻ button | Là nút để mở modal pop up để điều chỉnh các thông số của video trước khi gửi thông tin yêu cầu render, các thông tin như khung hình thể hiện, tốc độ phát. |

#### Giao diện modal setting



Hình Hình ảnh giao diên chi tiết của giao diện modal setting



Hình HÌnh ảnh giao diện chi tiết của hộp thoại modal setting sau khi khích hoạt loadreview

Bảng bảng mô tả chi tiết giao diện modal setting

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên đối tượng | Loại | Mô tả |
| 1 | Close popup button | Button | Là nút để tắt đi bảng popup đang hiển thị |
| 2 | Loadreview | Button | là nút để mở hợp thoại xem trước |
| 3 | Extension Name | Selection | Là thanh lựa chọn để chọn phần đuôi mở rộng của video sau khi xuất |
| 4 | Video ratio selection | Selection | Là thẻ selector để chọn ra hình dạng khung hình cho video xuất ra, là video ngang hay video dọc |
| 5 | Video Resolution | Selection | Là thẻ select để lựa chọn độ phân giải cảu video. |
| 6 | Speedup checkbox | Input[checkbox] | Là checkbox để server xác định là có thay đổi tốc độ phát của video lên 2 lần hay không |
| 7 | LUT Activate | Thẻ div | Là thẻ div thể hiện việc người dùng đã thêm filter vào video chưa, tại đây có 2 trạng thái là active và deactive thể hiện là có đặt filter lên hay chưa |
| 8 | Email checkbox | Checkbox | Checkbox thể hiện yêu cầu gửi qua email, người dùng chọn vào checkbox này thì video sau khi render sẽ gửi thẳng vào email của người dùng |
| 9 | Send request button | Button | Là nút để có thể gửi toàn bộ thông tin về video được yêu cầu thực hiện lên server, trong đó có cả các thông tin về render trong bảng chọn render option |
| 10 | Video Review | Video | Là thẻ video để người dùng xem lại video gồm các điểm cắt ghép thô |
| 11 | Load Preview Video | Button | Là nút để play video được chỉnh sửa thô |
| 12 | Cut point skip | Button | Là nhóm nút để tua để chổ đoạn cắt video |
| 13 | Hue Rotate slider | Input[range] | Là thanh để thay đổi hue của video chỉ số được thể hiện từ -180 tới 180 |
| 14 | Saturation slider | Input[range] | Là thanh đẻ thay đổi độ tươi của màu sắc, nó giúp màu trở nên đậm hơn hoặc nhạt hơn |
| 15 | Brightness slider | Input[range] | Là thanh đẻ thay đổi độ sáng của màu sắc, nó giúp hình ảnh trên video sẽ sáng hoặc tối hơn |
| 16 | Emit button | Button | Là nút để kiết xuất màu vào video để đưa cho server render |
| 17 | Email input | Input[text] | Là vùng đẻ người dùng nhập email khi có email thì hệ thống sẽ gửi video về email khi mà server render xong |

## Thiết kế xử lý dữ liệu đoạn video

Về chương trình chỉnh sửa video này, phần xử lý video được sử dụng nền tảng FFmpg để tạo nên, nền tảng này cho phép thực hiện các thao tác lên video ở mức độ bitrate từ đó việc tác động lên video sẽ trở nên đơn giản. Tuy nhiều nền tảng này không có một giao diện lý thuyết, để có thể thao tác được ta thường phải thực hiện có đoạn yêu cầu lên nên tảng, các yêu cầu này được viết như các đoạn code, từ đó có thể tương tác lên video.

### Thiết kế cách thức cắt video

Bảng Bảng mô tả chi tiết thiết kế cách thức cắt video

|  |  |  |
| --- | --- | --- |
| Đoạn code | ffmpeg -y -ss **startPoint** -to **endPoint** -i **fileInputName** -c copy **fileOutputName** | |
| STT | Thuộc tính | Ý nghĩa |
| 1 | Ffmpeg | Đây là thuộc tính để gọi nền tảng này ra để thực hiện các thao tác sau đó |
| 2 | -y | Có ý nghĩ là đồng ý với tất cả yêu cầu, câu lênh này được sinh ra để các thông tin có thể ghi đè lên nếu xảy ra sự cố |
| 3 | -ss | Là thuộc tính xác định vị trí bắt đầu của điểm cần cắt ở đây sẽ là **startPoint** |
| 4 | -to | Là thuộc tính xác định điểm cuối cùng của đoạn cắt ở đây sẽ là **endPoint** |
| 5 | -i | Là thuộc tính chỉ nội dung đầu vào, ở đây sẽ là **fileInputName** |
| 6 | -c | Là thuộc tính chỉ về codec, nó nói về phương thức encodes và decodes trên từng bit data |
| 7 | Copy | Là thuộc tính chỉ việc copy ra, và ở đây là copy ra file **fileOutputName** |

### Thiết kế cách thức ghép video

Bảng Bảng mô tả chi tiết thiết kế cách thức ghép video

|  |  |  |  |
| --- | --- | --- | --- |
| Đoạn code | | ffmpeg -y -i **fileNameNo1** -i **fileNameNo2** -filter\_complex "[0]scale=**frameRatio**:force\_original\_aspect\_ratio=decrease, pad=**frameRatio**:(ow-iw)/2:(oh-ih)/2, setsar=1[v0];[1]scale=**frameRatio**:force\_original\_aspect\_ratio=decrease,pad=**frameRatio**:(ow-iw)/2:(oh-ih)/2, setsar=1[v1];[v0][0:a:0][v1][1:a:0]concat=n=2:v=1:a=1[v][a]" -map "[v]" -map "[a]" **fileOutputName** | |
| STT | Thuộc tính | | Ý nghĩa |
| 1 | -i | | Là thuộc tính chỉ nội dung đầu vào, ở đây sẽ là **fileNameNo1** và **fileNameNo2** |
| 2 | -filter\_complex | | Là thuộc tính để tạo ra các câu lệnh phức hợp tác động lên video, ta có thể tổ hợp từ 2 câu lênh lại với nhau, nhưng sau đó cần phải ghép các thông tin lại với nhau |
| 3 | [variable] | | Đây là thuộc tính nói về vị trí của file được đưa vào thông qua thuộc tính -I, nếu variable là số thì chỉ vị trí của video trong mảng, nếu là chữ thì chỉ thuộc tính tác động lên |
| 4 | scale= | | Đẩy kích thước của video lên đúng với kích thước của **frameRatio** |
| 5 | force\_original\_aspect\_ratio=decrease | | Là lệch yêu cầu không thay đổi chiều kéo dãn |
| 6 | pad=**frameRatio**:(ow-iw)/2:(oh-ih)/2 | | Là lệnh yêu cầu tạo ra khung ảnh chính để có thể đặt các video lên, ow là output-width, iw là input-width, oh là output-height và ih là input-height |
| 7 | setsar=1[v1];[v0][0:a:0][v1][1:a:0] concat=n=2:v=1:a=1[v][a] | | Là đoạn lênh yêu cầu ghép vị trí tương đối giữa hai khung ảnh, và ghép hai âm thanh lại |
| 8 | -map "[v]" -map "[a]" | | Nối hai phần video và audio lại và cuối cùng là ghi nào vào một file tên **fileOutputName** |

### Thiết kê phương thức tăng tốc video

Bảng Bảng mô tả chi tiết thiết kế cách thức tăng tốc video

|  |  |  |  |
| --- | --- | --- | --- |
| Đoạn code | | ffmpeg -i **fileInputName** -filter\_complex "[0:v]setpts=0.5\*PTS[v];[0:a]atempo=2.0[a]" -map "[v]" -map "[a]" **fileOutputName** | |
| STT | Thuộc tính | | Ý nghĩa |
| 1 | -i | | Là thuộc tính chỉ nội dung đầu vào, ở đây sẽ là **fileInputName** |
| 2 | -filter\_complex | | Là thuộc tính để tạo ra các câu lệnh tác động lên video một cách phức hợp |
| 3 | [0:v]setpts=0.5\*PTS[v]; [0:a]atempo=2.0[a] | | Là đoạn lệnh ghép trong đó setpts làm thay đổi tốc độ video, còn atempo là nén phần âm thanh lại, PTS là tốc độ phát. |
| 4 | -map "[v]" -map "[a]" | | Nối hai phần video và audio lại và cuối cùng là ghi nào vào một file tên **fileOutputName** |

### Thiết kế phương thức thay đổi màu sắc video

Bảng Bảng mô tả chi tiết thiết kế cách thức thay đổi màu sắc video

|  |  |  |  |
| --- | --- | --- | --- |
| Đoạn code | | ffmpeg -y -i **fileInputName** -vf hue=h=**filterData.hue**:s=**filterData.saturate**:b=**filterData.brightness** **fileOutputName** | |
| STT | Thuộc tính | | Ý nghĩa |
| 1 | -i | | Là thuộc tính chỉ nội dung đầu vào, ở đây sẽ là **fileInputName** |
| 2 | -vf | | Là thuộc tính để tạo ra các câu lệnh phức tạp nhưng được phân cách bởi “**;**”, thường được dùng cho nhóm lệnh đơn giản |
| 3 | Hue | | Là thuộc tính chỉ về màu sắc để tác động lên |
| 4 | h | | Là thuộc tính chỉ hue rotate được đo bằng độ |
| 5 | s | | Là thuộc tính chỉ saturation, điểm cân bằng là 1 từ đó sẽ thay đổi tăng giảm để độ tươi của màu thay đổi theo |
| 6 | b | | Là thuộc tính chỉ brightness, điểm cân bằng là 1 từ đó tăng hoặc giảm sẽ làm cho độ sáng của video thay đổi theo |

# CÀI ĐẶT VÀ KIỂM THỬ

## Cài đặt

### Cài đặt môi trường

Cài đặt NodeJS phiên bản 16.17.1, tại trang chủ chính thức của NodeJS: <https://nodejs.org/en/download/releases/>

Tải xuống FFmpeg tại trang chủ: <https://ffmpeg.org/download.html>

Tải xuống chương trình tại: <https://github.com/thedeadsheep/ang-VideoEditing>

### Cài đặt phần mềm

**Front-end:**

* Mở thư mục của chương trình vừa tải từ trang github
* Mở command line tại chính thư mục đó và nhập lênh:

1. Chạy **npm install (hoặc npm i)** để cài đặt các thư viện cần thiết để chạy chương trình
2. Chạy **npm start** để khời động chương trình, về mặc định chương trình sẽ chạy tại **localhost:4200/**

**Back-end:**

* Tại thư mục project, ta truy cập vào thư mục BE.
* Tại thư mục BE ta tạo mới một thư mục **uploads**, đây là thư mục sẽ chứa các đoạn video được người dùng đăng tải lên server.
* Cũng tại thư mục BE này, ta mở command line là thực hiện các đoạn lệnh sau:

1. Chạy **npm install (hoặc npm i)** để cài đặt các thư viện cần thiết để chạy chương trình
2. Chạy **npm start** để khời động chương trình, về mặc định chương trình sẽ chạy tại localhost:3000

**FFmpeg:**

Sau khi download được phiên bản bất kỳ trên trang chủ của FFmpeg. Đối với Windows ta thực hiện **giải nén file tải về** đặt thư mục này tại một vị trí bất kỳ, truy cập vào **Edit the system Environment variables, Environment Variables,** tại vị trí **Path** ta thêm vị trí thư mục giải nén của ffmpeg, sau đó mở commandline nhập **ffmpeg** để kiểm tra ffmpeg đã có thể hoạt động chưa.

**Truy cập vào chương trình:**

* Ta mở một web browser bất kỳ và truy cập vào địa chỉ sau: <http://localhost:4200/> để truy cập vào chương trình

## Kiểm thử

### Kiểm thử chức năng upload

Bảng Bảng mô tả kiểm thử chức năng upload video

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Testcase | Test title | Steps | Data | Expected result | Actual result | Pass |
| TCupload1 | Đăng tải video bất kỳ | 1. Mở hộp thoại đăng tải 2. Chọn video 1.mp4 3. Bấm nút đăng tải. | Video 1.mp4 | Video được upload lên, và không sảy ra sự cố | Video được upload lên, và không sảy ra sự cố | Pass |
| TCupload2 | Đăng tải video, lại một video đã được upload | 1. Mở hộp thoại đăng tải 2. Chọn video 1.mp4 3. Bấm nút đăng tải. 4. Lập lại thao tác mở hộp thoại đăng tải. 5. Chọn lại video 1.mp4 6. Bấm nút đăng tải | Video 1.mp4 | Video mới được chọn không hiển thị lên chương trình | Video sau không xuất hiện | Pass |
| TCupload3 | Kéo thả các video chương trình | 1. Mở thư mục chứa video cần đăng tải. 2. Nắm kéo thả các video vào chương trình | Video 1.mp4 | Các video được thả vào chương trình, xuất hiện trên giao diện | Video xuất hiện trên chương trình | Pass |

### Kiểm thử chức năng send request và render video

Bảng Bảng mô tả chức năng send request và render video

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Testcase | Test title | Steps | Data | Expected result | Actual result | Pass |
| TCrender01 | Yêu cầu render hai video tương đồng | Chọn video từ máy tính  Click chuột vào video được đăng tải  Chọn sang chế động đánh dấu  Đánh dấu video từ 0 giây tới 4 giây.  Đánh dấu video từ 6 giây tới 12 giây  Sắp xếp 2 đoạn cắt lại  Gửi yêu cầu render video | Video 1.mp4 | Video trả về dài 10 giây có khung hình tương đồng với video gốc | Video trả về với độ dài 10 giây | Pass |
| TCrender02 | Yêu cầu render một video ngang thành một video dọc | Chọn video từ máy tính  Click chuột vào video được đăng tải  Chọn sang chế động đánh dấu  Đánh dấu video từ 0 giây tới 4 giây.  Chọn đoạn cắt.  Gửi yêu cầu xử lý video với video ratio là landscape. | Video 2.mp4 | Video trả về có độ dài là 4 giây và khung hình video sẽ bị bóp lại thành chiều dọc | Video dọc có độ dài 4 giây | Pass |

# KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

## Tổng kết

Sau khoảng 15 tuần, từ lúc nhận đề tài cho tới khi bắt tay đặt nền móng đầu tiên cho chương trình, bản thân em đánh giá chương trình đã đạt được những mục tiêu ban đầu đề ra, đó là có khả năng cắt ghép các đoạn video lại với nhau. Cùng đó qua việc tìm hiều về đề tài, bản thân em tiếp thu được rất nhiều kiến thức từ trong việc lập trình và cả rất nhiều kinh nghiệm trong việc chỉnh sửa video. Ngoài ra từ đó bản thân em có được thêm nhiều kiến thức trong việc chuyển đổi một phần mềm từ nền tảng này lên một nền tảng khác, từ đó trau dồi kỹ năng phân tích và giải quyết vấn đề, hiểu được giới hạn của một số nền tảng.

## Kết luận

### Về mặt lý thuyết

Sau thời gian thực hiện đề tài, bản thân em đã tìm hiểu được một vài điều sau đây:

* Sử dụng Angular và NodeJS để tạo nên một phần mềm ứng dụng chạy trên nền tảng web
* Tận dụng giới hạn của webapp để tối đa hóa hiệu năng
* Cách tận tạo ra một server điều kiển một máy tính cá nhân.
* Ứng dung phần học của môn tương tác máy tính vào chương trình để thay đỏi UX cho hợp với một application chạy trên web.

### Về mặt ứng dụng

WebApplicaton chỉnh sửa video đã được hoàn thành một cách tương đối, có thể sử dụng nó như một công cụ để cắt các đoạn video một cách nhanh chóng, ngoài ra có thể ghép các đoạn lại thành một video hoàn chỉnh theo ý thích

### Ưu điểm

Chương trình đã thay đổi một ít về mặt UX so với các phần mềm khác, để trở nên thân thiện với người dùng web hơn. Ngoài ra, về việc giảm thiểu việc tương tác với server một cách tối đa đã làm chương trình trở nên nhanh hơn tương đối nếu để so với một webapp, tận dụng được giới hạn của HTML, làm nó trở nên gói gọn hơn và dùng ít tài nguyên của máy tính hơn.

### Nhược điểm

Về nhược điểm, chương trình thực chất có thời gian phát triển rất ít, vì phần lớn thời gian là để tìm phương pháp và tài liệu liên quan, cùng với đó toàn bộ được một mình em phát triển nên nó trở nên chậm chạp. Ngoài ra phần mềm hiện tại rất đơn sơ, tuy có thể cắt ghép được nhưng một phần cũng không thể nói hoàn chỉnh vì các testcase được người phát triển trực tiếp nghĩ ra nên không thể bao quát được toàn bộ.

## Hướng phát triển

* Thêm bộ lọc màu vào chương trình (đây là một tính năng cốt lõi ban đầu, nhưng vì không thể hiểu được cách thức bộ lọc màu xử lý nên kết quả không đưa ra được kết quả mong đợi)
* Thêm vào một số chức năng tùy chỉnh thông số về mặt kỹ thuật như bitrate.
* Thêm vào phần thêm âm nhạc đè lên phần âm thanh, hoặc tách riêng phần âm thanh.
* Cải thiện giao diện lại để thân thiện hơn.
* Cải thiện hiệu xuất phần ghép video.

# TÀI LIỆU THAM KHẢO

**Tiếng Việt**

[1] Web cơ bản, Thẻ <video> trong HTML, [Online]. Available: <https://webcoban.vn/html/the-video-trong-html-nhung-mot-trinh-xem-phim-vao-trang-web.html>

[2] @quandv Viblo, Sử dụng @Input() và @Output() trong angular, [Online]. Available: <https://viblo.asia/p/su-dung-atinput-va-atoutput-trong-angular-3P0lPJP8Kox>

**Tiếng Anh**

[3] FFmpeg, Document, [Online]. Available: <https://ffmpeg.org/ffmpeg.html>

[4] gopinav, Angular-Tutorial, [Online] Available: <https://github.com/gopinav/Angular-Tutorials>

[5] Samer Buna, Node.js Child Processes: Everything you need to know, [Online]. Available: <https://www.freecodecamp.org/news/node-js-child-processes-everything-you-need-to-know-e69498fe970a/>