

KHOA KỸ THUẬT VÀ CÔNG NGHỆ  
BỘ MÔN CÔNG NGHỆ THÔNG TIN



ĐỒ ÁN CƠ SỞ NGÀNH  
HỌC KỲ I, NĂM HỌC 2023 - 2024

**Nghiên Cứu Các Công Cụ AI**  
**Để Tạo Một AI Streamer**  
**Chuyên Livestream Bán Sản Phẩm**

*Giáo viên hướng dẫn:*  
Ts. Nguyễn Bảo Ân

*Sinh viên thực hiện:*  
Họ tên: Dương Thành Tân  
MSSV: 110121097  
Lớp: DA21TTB

*Trà Vinh, tháng..... năm.....*

KHOA KỸ THUẬT VÀ CÔNG NGHỆ  
BỘ MÔN CÔNG NGHỆ THÔNG TIN



ĐỒ ÁN CƠ SỞ NGÀNH  
HỌC KỲ I, NĂM HỌC 2023 - 2024

**Nghiên Cứu Các Công Cụ AI**  
**Để Tạo Một AI Streamer**  
**Chuyên Livestream Bán Sản Phẩm**

*Giáo viên hướng dẫn:*  
Ts. Nguyễn Bảo Ân

*Sinh viên thực hiện:*  
Họ tên: Dương Thành Tân  
MSSV: 110121097  
Lớp: DA21TTB

*Trà Vinh, tháng..... năm.....*

**KHOA KỸ THUẬT VÀ CÔNG NGHỆ  
BỘ MÔN CÔNG NGHỆ THÔNG TIN**

**ĐỀ CƯƠNG CHI TIẾT  
THỰC TẬP ĐỒ ÁN CƠ SỞ NGÀNH**

**Tên đề tài:** Nghiên cứu các công cụ AI để tạo một AI Streamer chuyên livestream bán sản phẩm.

**Giảng viên hướng dẫn:** NGUYỄN BẢO AN

**Thời gian thực hiện:** Từ 06/11/2023 đến 31/12/2023

**Sinh viên thực hiện:** DƯƠNG THÀNH TÂN    **MSSV:** 110121097

**Nội dung đề tài:**

Đề tài tập trung vào việc nghiên cứu và sử dụng các công cụ AI có sẵn trên internet để tạo ra một AI Streamer chuyên livestream bán sản phẩm. Các công cụ AI này bao gồm những công cụ có khả năng tạo nội dung văn bản, chuyển văn bản thành âm thanh, tạo hình ảnh, tạo video từ hình ảnh và tạo video có nhân vật biết cử động miệng khớp với lời thoại.

**Yêu cầu:**

- Nghiên cứu về các công cụ AI có sẵn trên internet để tạo nội dung livestream.
- Xây dựng ứng dụng tạo prompt cho các công cụ AI, tạo video livestream từ thông tin sản phẩm. Sử dụng ngôn ngữ lập trình Python và các thư viện liên quan để phát triển ứng dụng.
- Tìm hiểu quy trình tương tác API của các công cụ AI.

**Phương pháp thực hiện:**

**Nghiên cứu mô hình AI:** Đầu tiên, tiến hành nghiên cứu và hiểu rõ về các mô hình AI như ChatGPT 3.5 và Heygen 3.0, nhằm xác định khả năng và giới hạn của các mô hình trong việc xử lý và tạo nội dung.

**Lựa chọn các thư viện Python:** Tiếp theo, lựa chọn các thư viện Python cần thiết như requests, json, time, tempfile cho việc xử lý dữ liệu, cùng với các thư viện PyQt5 cho việc xây dựng giao diện và xử lý đa phương tiện, để sử dụng trong quá trình xây dựng phần mềm.

**Xây dựng phần mềm:** Sử dụng ngôn ngữ Python cùng với các thư viện đã chọn để xây dựng phần mềm. Phần mềm này sẽ có khả năng tạo prompt cho các công cụ AI, và tạo video livestream từ thông tin sản phẩm.

**Tích hợp API của AI:** Trong quá trình xây dựng phần mềm, thực hiện việc kết nối và tương tác với API của các công cụ AI như Openai API và HeyGen API.

**Kiểm tra và tối ưu:** Cuối cùng, thực hiện kiểm tra và tối ưu phần mềm để đảm bảo phần mềm hoạt động một cách hiệu quả và đáp ứng được yêu cầu đặt ra.

**Kết quả đạt được:**

Xây dựng được một phần mềm AI Streamer hoàn chỉnh. Phần mềm này có khả năng tạo ra video livestream bán sản phẩm từ thông tin sản phẩm đầu vào. Video sẽ bao gồm hình ảnh và âm thanh được tạo bởi các mô hình AI, giống như một streamer chuyên nghiệp thực hiện.

**Kế hoạch thực hiện:**

Tuần	Thời gian	Nội dung công việc	Người thực hiện
1	Từ 06/11/2023 đến 12/12/2023	- Nghiên cứu các mô hình ChatGPT 3.5 và HeyGen 3.0. - Tìm hiểu ngôn ngữ Python để xây dựng phần mềm.	Dương Thành Tân
2	Từ 20/11/2023 đến 26/11/2023	- Tìm hiểu thư viện Python như requests, json, tempfile để xử lý dữ liệu và PyQt5 để xây dựng giao diện phần mềm.	Dương Thành Tân
3	Từ 04/12/2023 đến 10/12/2023	- Xây dựng phần mềm để kết nối với API của Openai và HeyGen. - Kiểm tra sự tương tác của dữ liệu đầu vào dưới dạng văn bản và hình ảnh.	Dương Thành Tân
4	Từ 18/12/2023 đến 24/12/2023	- Thiết kế giao diện người dùng, kiểm tra và đánh giá chất lượng video. - Thực hiện kiểm thử toàn diện cho phần mềm và ghi nhận kết quả.	Dương Thành Tân
5	Từ 25/12/2023 đến 31/12/2023	- Tổng hợp kết quả kiểm thử, tiến hành tinh chỉnh và tối ưu hóa phần mềm. - Viết báo cáo.	Dương Thành Tân

**Xác nhận của GVHD**

Ngày.....tháng.....năm.....

**SV Thực hiện**

## NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN

A series of horizontal dotted lines for writing.

Trà Vinh, ngày ..... tháng ..... năm .....

**Giáo viên hướng dẫn**

(Ký tên và ghi rõ họ tên)

## NHẬN XÉT CỦA THÀNH VIÊN HỘI ĐỒNG

[illegible]

Trà Vinh, ngày ..... tháng ..... năm .....

**Thành viên hội đồng**  
(Ký tên và ghi rõ họ tên)

## LỜI CẢM ƠN

Trước hết, em xin gửi lời cảm ơn chân thành nhất đến thầy Nguyễn Bảo Ân, giảng viên hướng dẫn của em. Thầy đã hỗ trợ và chỉ dẫn tận tình cho em trong suốt quá trình thực hiện đồ án này. Những kiến thức, kỹ năng mà thầy đã truyền đạt không chỉ giúp em hoàn thành đồ án mà còn là bước đệm quan trọng cho sự nghiệp tương lai của em. Thầy đã tạo điều kiện cho em học hỏi, thực hành và giải đáp những thắc mắc trong quá trình thực hiện đồ án.

Tiếp theo, em xin gửi lời cảm ơn đến Khoa Kỹ thuật và Công Nghệ thuộc Trường Đại học Trà Vinh. Khoa đã tạo điều kiện và cung cấp những kiến thức bổ ích và là nền tảng vững chắc cho em trong quá trình học tập và thực hiện đồ án. Những quyền giáo trình và tài liệu tham khảo mà Khoa cung cấp đã trở thành nguồn tri thức quý giá, giúp em định hình và mở rộng kiến thức của mình.

Với những kiến thức đã được học tập và tìm hiểu, cùng với sự hướng dẫn nhiệt tình của thầy đã giúp em đã hoàn thành bài báo cáo đồ án cơ sở ngành. Tuy nhiên, em nhận thức được rằng trong quá trình thực hiện đề tài, em vẫn còn nhiều thiếu sót. Điều này không chỉ là do hạn chế về kiến thức và kỹ năng của bản thân, mà còn do sự phức tạp của đề tài. Em đã cố gắng học hỏi và khắc phục những khó khăn này, nhưng vẫn còn nhiều điều cần cải thiện.

Do đó, em mong nhận được đóng góp ý kiến của quý thầy cô để em có thể học tập thêm được nhiều kinh nghiệm, nâng cao kỹ năng và kiến thức của mình. Em tin rằng, với sự hỗ trợ và chỉ dẫn của quý thầy cô, em sẽ hoàn thành tốt hơn trong những bài báo cáo sắp tới.

Em xin chân thành cảm ơn!

## MỤC LỤC

<b>LỜI CẢM ƠN .....</b>	<b>3</b>
<b>MỤC LỤC .....</b>	<b>4</b>
<b>DANH MỤC HÌNH .....</b>	<b>6</b>
<b>DANH MỤC TỪ VIẾT TẮT .....</b>	<b>7</b>
<b>MỞ ĐẦU.....</b>	<b>9</b>
<b>CHƯƠNG 1: TỔNG QUAN.....</b>	<b>11</b>
<b>CHƯƠNG 2: NGHIÊN CỨU LÝ THUYẾT.....</b>	<b>12</b>
2.1. Giới thiệu về Prompt Engineering.....	12
2.1.1. Khái niệm Prompt Engineering .....	12
2.1.2. Ứng dụng của Prompt Engineering .....	13
2.1.3. Phương pháp trong Prompt Engineering.....	13
2.1.4. Kỹ thuật trong Prompt Engineering .....	15
2.2. Giới thiệu về OpenAI API .....	16
2.2.1. Khái niệm OpenAI API .....	16
2.2.2. Cách thức hoạt động của OpenAI API .....	17
2.2.3. Các tính năng của OpenAI API .....	17
2.2.4. Ứng dụng của OpenAI API .....	18
2.3. Giới thiệu về D-ID API .....	19
2.3.1. Khái niệm D-ID API.....	19
2.3.2. Cách thức hoạt động của D-ID API.....	19
2.3.3. Các tính năng của D-ID API.....	20
2.3.4. Ứng dụng của D-ID API.....	21
2.4. Giới thiệu về Python.....	22
2.4.1. Khái niệm Python .....	22
2.4.2. Lịch sử phát triển của Python.....	22
2.4.4. Ứng dụng tiêu biểu của Python .....	22
2.5. Giới thiệu về Framework PyQt5 .....	23
2.5.1. Khái niệm về Framework PyQt5.....	23
2.5.2. Các thành phần cơ bản của Framework PyQt5 .....	24
2.5.3. Ứng dụng Framework PyQt5 trong Python .....	24



<b>CHƯƠNG 3: HIỆN THỰC HÓA NGHIÊN CỨU .....</b>	<b>26</b>
3.1. Đặt tả ứng dụng .....	26
3.2 Xây dựng giao diện người dùng GUI với PyQt5 qua Python .....	26
3.3 Tích hợp OpenAI API và xử lý dữ liệu JSON .....	28
3.4 Tích hợp D-ID API để tạo video .....	29
3.5 Ứng dụng thư viện Cachetools để lưu cache và xuất video .....	31
<b>CHƯƠNG 4: KẾT QUẢ NGHIÊN CỨU .....</b>	<b>32</b>
4.1. Dữ liệu thử nghiệm.....	32
4.2. Cài đặt môi trường.....	33
4.3. Thiết kế giao diện .....	34
4.4. Các chức năng của hệ thống.....	35
<b>CHƯƠNG 5: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN .....</b>	<b>36</b>
5.1. Kết luận.....	36
5.2. Hạn chế .....	36
5.3. Phương hướng phát triển .....	37
<b>DANH MỤC TÀI LIỆU THAM KHẢO .....</b>	<b>38</b>
<b>PHỤ LỤC .....</b>	<b>39</b>

## DANH MỤC HÌNH

Hình 2.1. Ảnh minh họa cấu trúc cơ bản về Prompt Engineering.....	12
Hình 2.2. Ảnh minh họa cấu trúc chi tiết Prompt Engineering .....	14
Hình 2.3. Ảnh minh họa cấu trúc từ khóa Prompt Engineering .....	15
Hình 2.4. Ảnh minh họa cấu trúc câu hỏi mở Prompt Engineering .....	16
Hình 2.5. Ảnh minh họa được tạo bởi Dall-E 3 .....	19
Hình 2.6. Ảnh minh họa quy trình tạo video từ D-ID API.....	20
Hình 2.7. Ảnh minh họa quy trình tạo video Streams từ D-ID API.....	20
Hình 2.8. Ảnh minh họa quy trình tạo video Animations từ D-ID API .....	21
Hình 2. 9. Ảnh minh họa cú pháp cơ bản Python.....	22
Hình 3.1. Hàm xây dựng giao diện người dùng .....	25
Hình 3.2. Hàm kết nối các sự kiện .....	26
Hình 3.3. Hàm kết nối OpenAI API .....	26
Hình 3.4. Hàm xử lý dữ liệu JSON .....	26
Hình 3.5. Cấu trúc JSON của D-ID API.....	27
Hình 3 6. Kết quả trả về từ lệnh POST của D-ID API .....	28
Hình 3.7. Hàm gửi yêu cầu GET đến D-ID API .....	28
Hình 3.8. Hàm create_video() sử dụng Cachetools.....	29
Hình 3.1. Hàm xây dựng giao diện người dùng .....	27
Hình 3.2. Hàm kết nối các sự kiện .....	28
Hình 3.3. Hàm kết nối OpenAI API .....	28
Hình 3.4. Hàm xử lý dữ liệu JSON .....	28
Hình 3.5. Cấu trúc JSON của D-ID API.....	29
Hình 3 6. Kết quả trả về từ lệnh POST của D-ID API .....	30
Hình 3. 7. Hàm gửi yêu cầu GET đến D-ID API.....	30
Hình 3. 8. Hàm create_video() sử dụng Cachetools.....	31
Hình 4.1. Dữ liệu thử nghiệm về sản phẩm nước hoa.....	32
Hình 4.2. Dữ liệu thử nghiệm về sản phẩm sữa tắm .....	32
Hình 4.3. Dữ liệu thử nghiệm về sản phẩm bột giặt.....	33
Hình 4. 4. Giao diện ứng dụng AI Streamer.....	34

DANH MỤC TỪ VIẾT TẮT

STT	KÝ HIỆU VIẾT TẮT	NỘI DUNG VIẾT TẮT
1	AI	Artificial Intelligence
2	API	Application Programming Interface
3	ChatGPT	Chat Generative Pre-training Transformer
4	ChatGPT 3.5	Chat Generative Pre-trained Transformer 3.5
5	D-ID	De-Identification Ltd
6	GPT-3	Generative Pre-trained Transformer 3
7	GPT-4	Generative Pre-trained Transformer 4
8	GUI	Graphical User Interface
9	JSON	JavaScript Object Notation
10	LLM	Large Language Model
11	ML	Machine Learning
12	OpenAI	Open Artificial Intelligence

## TÓM TẮT ĐỀ ÁN CƠ SỞ NGÀNH

Đề án “Nghiên cứu các công cụ AI để tạo một AI Streamer chuyên livestream bán sản phẩm” mở ra một hướng tiếp cận mới trong việc ứng dụng công nghệ AI vào lĩnh vực livestream bán hàng.

Trọng tâm của nghiên cứu là việc tìm hiểu và tận dụng các công cụ AI tiên tiến như ChatGPT, D-ID và Microsoft Azure Voices thông qua các API được viết và tạo dựng sẵn. Mục đích là tạo ra một AI Streamer có khả năng livestream và bán sản phẩm một cách tự động và hiệu quả.

Đầu tiên, kỹ thuật Prompt Engineering được áp dụng để tạo ra một cấu trúc lời nhắc tối ưu, giúp người dùng nhập vào thông tin sản phẩm một cách hiệu quả. Cấu trúc này sau đó được truyền tới ChatGPT, một công cụ AI mạnh mẽ, để xử lý và phân tích dữ liệu, tạo ra văn bản mô tả sản phẩm phù hợp từ thông tin đầu vào.

Tiếp theo, văn bản được tạo ra bởi ChatGPT được truyền đến D-ID. D-ID, với sự hỗ trợ của Microsoft Azure Voices, chuyển đổi văn bản thành âm thanh và tạo dựng hình ảnh nhép miệng phù hợp với âm thanh, tạo ra video giới thiệu sản phẩm.

Cuối cùng, Python và PyQt5 được sử dụng để kết nối các phần lại với nhau và thiết kế giao diện người dùng thân thiện và dễ sử dụng.

Kết quả cuối cùng là một AI Streamer có khả năng livestream và bán sản phẩm một cách tự động và hiệu quả. Đề án này không chỉ mở rộng kiến thức về AI và lập trình, mà còn giúp hiểu rõ hơn về tiềm năng của AI trong lĩnh vực bán hàng trực tuyến. Ngoài ra, đây còn là một bước tiến quan trọng trong việc khám phá tiềm năng của AI trong việc cải thiện hiệu suất và trải nghiệm người dùng trong việc ứng dụng và tương tác API giữa các mô hình AI. Đồng thời đây cũng là một bước tiến quan trọng trong việc khám phá tiềm năng của AI trong lĩnh vực công nghệ thông tin đang phát triển nhanh chóng hiện nay.

## MỞ ĐẦU

### 1. Lý do chọn đề tài

Trong thời đại số hóa hiện nay, việc ứng dụng AI vào lĩnh vực livestream bán hàng trở nên cần thiết và tiềm năng. Sự phát triển của công nghệ thông tin đã tạo ra một sự thay đổi lớn trong cách thức mua sắm của người tiêu dùng, với xu hướng chuyển dịch từ mua sắm trực tiếp sang mua sắm trực tuyến. Trong bối cảnh đó, việc sử dụng AI để tạo ra một giải pháp livestream bán hàng hiệu quả trở thành một yêu cầu không thể thiếu.

Đề tài “Nghiên cứu các công cụ AI để tạo một AI Streamer chuyên livestream bán sản phẩm” được ứng dụng nhằm khám phá và tận dụng công nghệ AI để đáp ứng nhu cầu này. Mục tiêu là tạo ra một AI Streamer có khả năng livestream và bán sản phẩm một cách tự động và hiệu quả, giúp cải thiện trải nghiệm mua sắm của người tiêu dùng và tăng hiệu suất kinh doanh cho các doanh nghiệp.

Bên cạnh đó, việc nghiên cứu và áp dụng AI trong lĩnh vực này cũng mang lại nhiều lợi ích khác như:

- AI giúp tối ưu hóa quá trình bán hàng, giảm thiểu thời gian và công sức cần thiết để quản lý và điều hành các hoạt động livestream.
- AI giúp cải thiện chất lượng và hiệu quả của việc bán hàng, thông qua việc tạo ra các video livestream chất lượng cao và thu hút.

Cuối cùng, việc áp dụng AI còn giúp tăng cường khả năng tương tác với khách hàng, qua việc cung cấp các thông tin sản phẩm một cách tự động và chính xác.

### 2. Mục tiêu nghiên cứu

- Tìm hiểu các ứng dụng Prompt Engineering vào ChatGPT.
- Tìm hiểu về cách tương tác API của các công cụ AI như ChatGPT, D-ID.
- Tạo ra một AI Streamer có khả năng livestream và bán sản phẩm tự động.

### 3. Phương pháp nghiên cứu

**Nghiên cứu về việc tích hợp các API của các công cụ AI như:** ChatGPT, D-ID và Microsoft Azure Voices để tạo thành một hệ thống có cấu trúc phù hợp. Mục tiêu là tạo ra một AI Streamer có khả năng tự động livestream và bán sản phẩm một cách hiệu quả.

**Nghiên cứu ngôn ngữ lập trình Python để kết nối các API với nhau:** Python sẽ được sử dụng để giao tiếp với các API, xử lý dữ liệu đầu vào từ người dùng, và điều khiển hoạt động của AI Streamer.

**Nghiên cứu áp dụng framework PyQt5:** Để phát triển giao diện người dùng.

#### **4..Đối tượng nghiên cứu**

- Các công cụ AI như: ChatGPT, D-ID và Microsoft Azure Voices.
- Cấu trúc Prompt Engineering.
- Ngôn ngữ lập trình Python.
- Framework PyQt5.

#### **5. Phạm vi nghiên cứu**

Nghiên cứu sẽ tập trung vào việc ứng dụng AI Streamer trong lĩnh vực livestream bán hàng.

## CHƯƠNG 1: TỔNG QUAN

Trong thời đại số hóa hiện nay, việc ứng dụng AI vào lĩnh vực livestream bán hàng trở nên cần thiết và tiềm năng. Sự phát triển của công nghệ thông tin đã tạo ra một sự thay đổi lớn trong cách thức mua sắm của người tiêu dùng, với xu hướng chuyển dịch từ mua sắm trực tiếp sang mua sắm trực tuyến. Trong bối cảnh đó, việc sử dụng AI để tạo ra một giải pháp livestream bán hàng hiệu quả trở thành một yêu cầu không thể thiếu.

Để giải quyết vấn đề này, các công cụ AI như ChatGPT, D-ID đã được sử dụng để xây dựng ứng dụng. Các công cụ này cung cấp các API mạnh mẽ và linh hoạt, cho phép tạo ra một AI Streamer có khả năng livestream và bán sản phẩm một cách tự động và hiệu quả.

Qua quá trình nghiên cứu và thử nghiệm và tìm hiểu được cách tương tác API của các công cụ AI, cách xử lý dữ liệu JSON, cách tạo video với D-ID API, và cách ứng dụng thư viện Cachetools để lưu cache và xuất video. Những kỹ năng và kiến thức này đã giúp trong việc tạo ra một hệ thống livestream bán hàng tự động và hiệu quả.

Kết quả cuối cùng sau quá trình thực hiện là một ứng dụng có thể được triển khai trong lĩnh vực livestream bán hàng. Ứng dụng đáp ứng được yêu cầu về thời gian chạy hợp lý và phù hợp với yêu cầu của người dùng. Với AI Streamer, người dùng có thể tự động hóa quá trình livestream và bán hàng, giảm thiểu thời gian và công sức cần thiết để quản lý và điều hành các hoạt động livestream. Đồng thời, AI Streamer cũng giúp cải thiện chất lượng và hiệu quả của việc bán hàng, thông qua việc tạo ra các video livestream chất lượng cao và thu hút. Cuối cùng, việc áp dụng AI còn giúp tăng cường khả năng tương tác thông qua việc cung cấp các thông tin sản phẩm một cách tự động và chính xác. Đây là một bước tiến quan trọng trong việc khám phá tiềm năng của AI trong lĩnh vực công nghệ thông tin đang phát triển nhanh chóng hiện nay.

## CHƯƠNG 2: NGHIÊN CỨU LÝ THUYẾT

### 2.1. Giới thiệu về Prompt Engineering

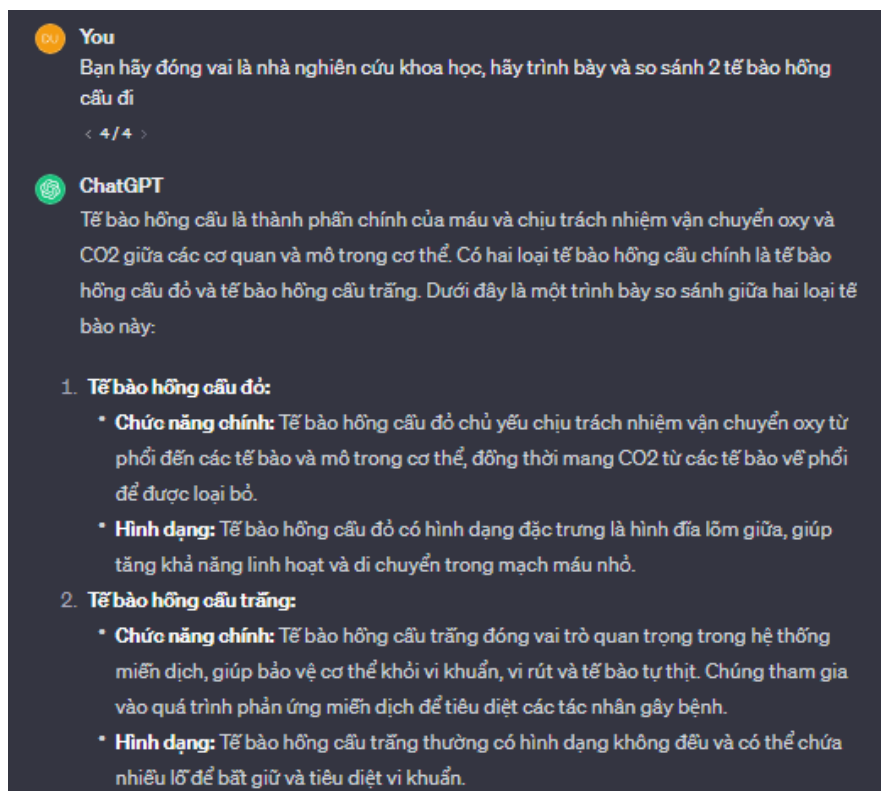
#### 2.1.1. Khái niệm Prompt Engineering

Prompt Engineering là quá trình tạo ra, tinh chỉnh và tối ưu hóa các đầu vào, hoặc prompts, để khuyến khích các hệ thống trí tuệ nhân tạo sinh (AI) tạo ra các đầu ra cụ thể, chất lượng cao.

Prompt Engineering giúp các mô hình AI sinh đáp ứng tốt hơn và phản hồi đa dạng các truy vấn, từ đơn giản đến phức tạp. Đồng thời, liên quan đến việc tạo ra các prompts hiệu quả để tối ưu hóa chất lượng và tính liên quan của nội dung do AI tạo ra, chẳng hạn như hình ảnh, mã, tóm tắt dữ liệu hoặc văn bản.

Một cách tiếp cận thông minh trong việc tạo ra các prompts là cần thiết để cầu nối khoảng cách giữa các truy vấn thô và các phản hồi có ý nghĩa do AI tạo ra. Bằng cách tinh chỉnh hiệu quả các prompts, giúp cho người đặt câu hỏi có thể tối ưu hóa đáng kể chất lượng và tính liên quan của các đầu ra để giải quyết cả vấn đề cụ thể.

Prompt Engineering đang chứng minh sự quan trọng của nó để phát huy tiềm năng đầy đủ của các mô hình cơ sở mà nguồn cung cấp cho AI sinh. Các mô hình cơ sở là các mô hình ngôn ngữ lớn (LLM) được xây dựng trên kiến trúc biến đổi và đóng gói tất cả thông tin mà hệ thống AI sinh cần.



Hình 2.1. Ảnh minh họa cấu trúc cơ bản về Prompt Engineering



### 2.1.2. Ứng dụng của Prompt Engineering

Prompt Engineering đóng vai trò quan trọng trong việc xác định chất lượng đầu ra của các mô hình generative AI. Một prompt được thiết kế tốt sẽ giúp mô hình tạo ra kết quả chính xác, phù hợp với yêu cầu và có tính sáng tạo.

Cụ thể, Prompt Engineering có những vai trò sau:

**Hiểu và diễn giải nhiệm vụ:** Một prompt tốt sẽ cung cấp cho mô hình tất cả thông tin cần thiết để hiểu và diễn giải nhiệm vụ. Điều này giúp mô hình tạo ra kết quả chính xác và phù hợp với yêu cầu. Giả sử, nếu chúng ta muốn mô hình viết một bài thơ về tình yêu, prompt có thể bao gồm thông tin về loại tình yêu (tình yêu gia đình, tình yêu bạn bè, tình yêu lãng mạn), các cảm xúc liên quan (hạnh phúc, buồn bã, mong đợi), và ngữ cảnh (tình yêu ở thời hiện đại hay thời cổ đại). Điều này giúp mô hình tạo ra một bài thơ phù hợp với chủ đề và ngữ cảnh đã được định rõ.

**Tạo ra kết quả sáng tạo:** Một prompt tốt không chỉ giúp mô hình tạo ra kết quả chính xác mà còn giúp mô hình tạo ra kết quả sáng tạo và độc đáo. Điều này là do prompt có thể cung cấp cho mô hình các ý tưởng mới và hướng dẫn mô hình đi theo những con đường khác nhau. Giả sử, nếu chúng ta muốn mô hình viết một câu chuyện về một thế giới trong tương lai, prompt có thể bao gồm thông tin về các công nghệ mới (như AI, robot), các vấn đề xã hội (như biến đổi khí hậu, quyền con người), và các nhân vật (như nhà khoa học, nhà lãnh đạo). Điều này giúp mô hình tạo ra một câu chuyện có cốt truyện hấp dẫn và các nhân vật thú vị.

**Tự động hóa các tác vụ:** Prompt Engineering có thể được sử dụng để tự động hóa các tác vụ, giúp tiết kiệm thời gian và công sức cho con người. Giả sử, nếu chúng ta muốn mô hình dịch một đoạn văn từ tiếng Anh sang tiếng Việt, prompt có thể bao gồm đoạn văn tiếng Anh và yêu cầu dịch đoạn văn sang tiếng Việt. Điều này giúp chúng ta dịch đoạn văn một cách nhanh chóng và chính xác, mà không cần phải thực hiện công việc này một cách thủ công.

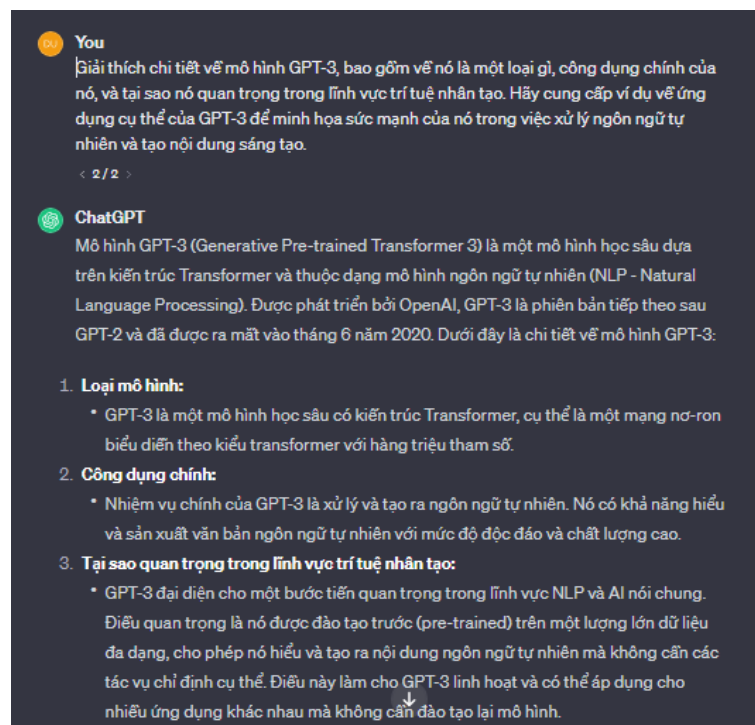
### 2.1.3. Phương pháp trong Prompt Engineering

Prompt engineering đang trở thành một phần quan trọng trong việc tối ưu hóa hiệu suất của mô hình ngôn ngữ như GPT-3 và GPT-4. Đây là một quá trình mà người dùng điều chỉnh cách họ đặt câu hỏi hoặc gửi đầu vào để đạt được kết quả mong muốn. Dưới đây là một số phương pháp cụ thể trong prompt engineering:

**Tinh chỉnh và thử nghiệm:** Điều này liên quan đến việc thay đổi các từ khóa, cấu trúc câu, hoặc thậm chí là thứ tự từ trong câu hỏi để xem cách mô hình phản ứng và thử nghiệm nhiều biến thể của câu hỏi để xác định được thông tin mong muốn.

**Sử dụng mẫu câu phức tạp:** Điều này liên quan đến việc tạo ra câu hỏi phức tạp và chi tiết hơn để tăng khả năng mô hình hiểu và trả lời chính xác. Hạn chế việc sử dụng câu hỏi ngắn và đơn giản, đặt câu hỏi cụ thể để đảm bảo kết quả chính xác hơn.

**Sử dụng mô tả chi tiết:** Thêm mô tả chi tiết vào câu hỏi để làm rõ yêu cầu và giúp mô hình hiểu rõ ngữ cảnh. Kết hợp các vấn đề và yêu cầu với ví dụ cụ thể để cung cấp thông tin thêm vào.



Hình 2.2. Ảnh minh họa cấu trúc chi tiết Prompt Engineering

**Kiểm soát nghĩa của từng từ:** Xác định các từ khóa quan trọng trong câu hỏi và tìm cách định rõ ý nghĩa của chúng. Cân nhắc sử dụng từ ngữ chính xác để tránh hiểu lầm.

**Sử dụng hình ảnh và ví dụ:** Đối với các mô hình hỗ trợ đầu vào đa dạng, bạn có thể cung cấp hình ảnh hoặc ví dụ cụ thể để mô hình có thể tận dụng thông tin hình ảnh khi trả lời.

**Kết hợp câu hỏi:** Kết hợp nhiều câu hỏi để tạo ra một yêu cầu phức tạp và đa chiều. Kỹ thuật này có thể giúp mô hình nắm bắt ngữ cảnh toàn diện hơn.

**Sử dụng các thẻ hệ thống:** Sử dụng các thẻ đặc biệt để có sẵn trong hệ thống để chỉ định mô hình cách phải xử lý đầu vào, ví dụ như “Translate the following sentence” hoặc “List three reasons why.”

**Lặp lại câu hỏi:** Thử nghiệm việc lặp lại câu hỏi với những biến thể nhỏ để xem liệu mô hình có thể cung cấp các câu trả lời nhất quán hay không.

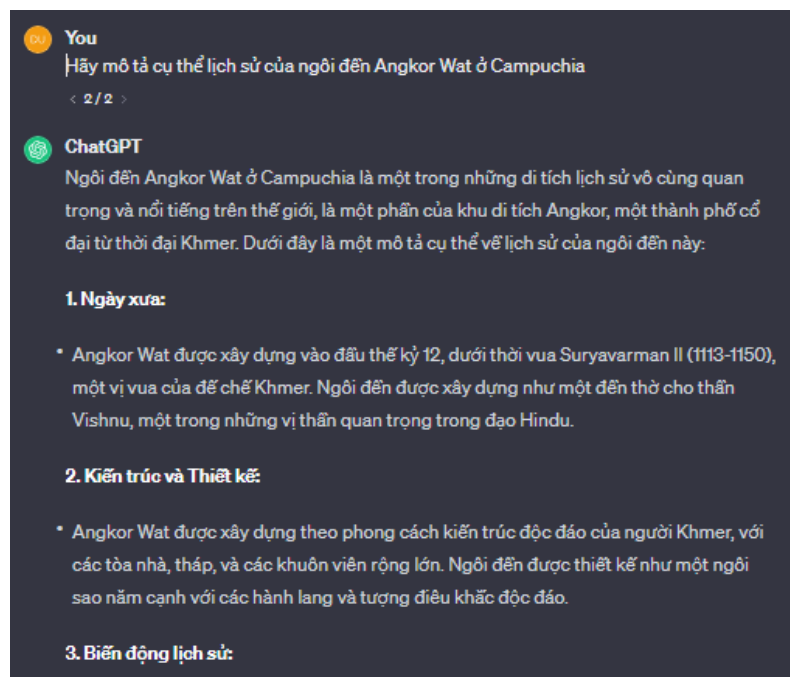
#### 2.1.4. Kỹ thuật trong Prompt Engineering

Kỹ thuật trong Prompt Engineering giúp tối ưu hóa cách đặt câu hỏi hoặc gửi đầu vào để thu được kết quả tốt nhất từ mô hình ngôn ngữ. Dưới đây là một số kỹ thuật phổ biến:

**Áp dụng kiến thức chuyên môn:** Kiến thức chuyên môn về chủ đề cần truy vấn sẽ giúp mô hình hiểu rõ hơn. Giả sử, khi người dùng truy vấn về một chủ đề khoa học, việc cung cấp định nghĩa hoặc khái niệm cơ bản về các thuật ngữ liên quan sẽ hữu ích trong việc đặt câu hỏi. Điều này giúp mô hình ngôn ngữ có thể xử lý câu hỏi một cách chính xác hơn, do nó đã được “hướng dẫn” về ngữ cảnh và ngữ nghĩa của câu hỏi.

**Sử dụng ngôn ngữ tự nhiên:** Ngôn ngữ tự nhiên trong việc đặt câu hỏi hoặc gửi đầu vào giúp mô hình dễ dàng hiểu và cung cấp kết quả chính xác hơn. Mô hình ngôn ngữ được huấn luyện trên dữ liệu ngôn ngữ tự nhiên, do đó việc sử dụng ngôn ngữ tự nhiên giúp mô hình “cảm thấy” quen thuộc hơn với đầu vào và do đó có thể xử lý nó một cách hiệu quả hơn.

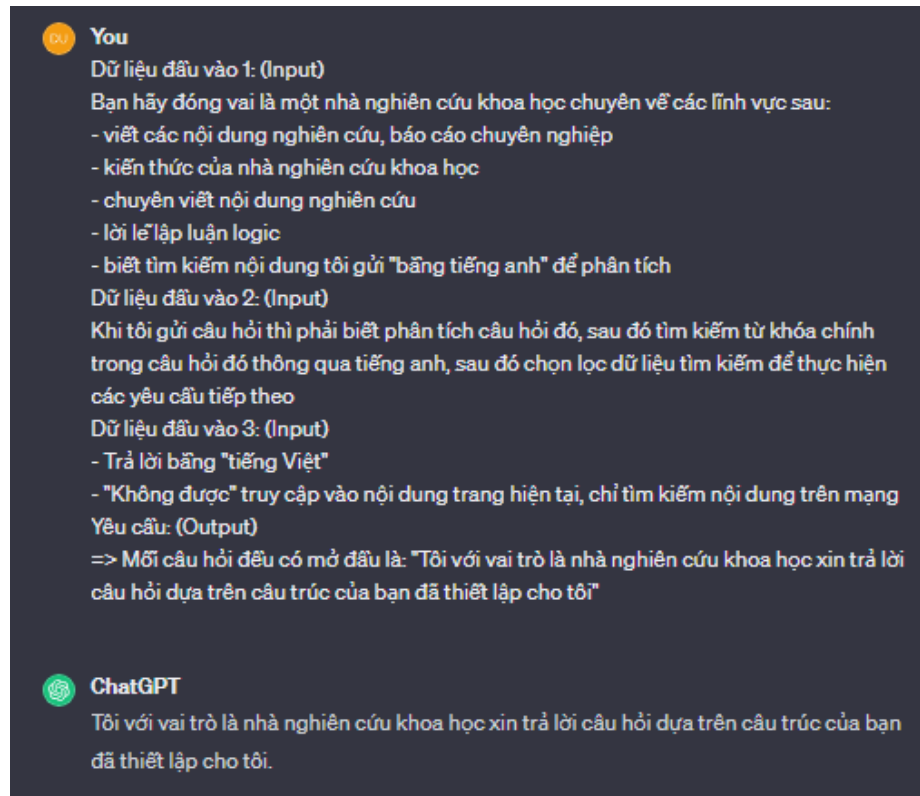
**Sử dụng từ khóa và cụm từ cụ thể:** Việc sử dụng từ khóa và cụm từ cụ thể khi đặt câu hỏi hoặc gửi đầu vào giúp mô hình tập trung vào thông tin cần thiết và cung cấp kết quả chính xác hơn.



Hình 2.3. Ảnh minh họa cấu trúc từ khóa Prompt Engineering

**Sử dụng mệnh lệnh:** Khi yêu cầu mô hình thực hiện một hành động, việc sử dụng mệnh lệnh giúp mô hình hiểu rõ hơn những gì cần thực hiện.

**Sử dụng câu hỏi mở:** Khi muốn mô hình cung cấp nhiều thông tin, việc sử dụng câu hỏi mở giúp người dùng hiểu rõ hơn về chủ đề đang được thảo luận. Câu hỏi mở không giới hạn mô hình ngôn ngữ trong việc tạo ra câu trả lời, cho phép nó khám phá nhiều khía cạnh khác nhau của chủ đề và cung cấp thông tin phong phú hơn.



Hình 2.4. Ảnh minh họa cấu trúc câu hỏi mở Prompt Engineering

## 2.2. Giới thiệu về OpenAI API

### 2.2.1. Khái niệm OpenAI API

OpenAI API là một dịch vụ do OpenAI cung cấp, một tổ chức nghiên cứu trí tuệ nhân tạo hàng đầu. API (Application Programming Interface) là một giao diện cho phép các ứng dụng và chương trình tương tác với các dịch vụ và chức năng của OpenAI.

Khái niệm này liên quan đến việc sử dụng các mô hình ngôn ngữ nhân tạo mạnh mẽ, như GPT (Generative Pre-trained Transformer), thông qua API để thực hiện các nhiệm vụ liên quan đến xử lý ngôn ngữ tự nhiên.

OpenAI API cho phép các nhà phát triển và người dùng tích hợp khả năng xử lý ngôn ngữ tự nhiên cao cấp vào ứng dụng của họ mà không cần phải đào tạo mô hình từ đầu.

Thông qua OpenAI API, các nhà phát triển có thể sử dụng các tính năng như sinh văn bản, phân loại văn bản, tạo đề xuất và nhiều ứng dụng khác. Điều này tạo ra cơ hội lớn trong việc phát triển ứng dụng thông minh và linh hoạt trên nền tảng trí tuệ nhân tạo của OpenAI.

### 2.2.2. Cách thức hoạt động của OpenAI API

Cách thức hoạt động của OpenAI API được thể hiện thông qua mô hình ChatGPT bao gồm:

**Khởi tạo Yêu Cầu (Request):** Người dùng hoặc nhà phát triển bắt đầu quá trình bằng cách tạo một yêu cầu thông qua giao thức HTTP, thường là sử dụng phương thức POST. Trong yêu cầu, họ cung cấp thông tin văn bản hoặc nhiệm vụ mà họ muốn mô hình ngôn ngữ xử lý. Ngoài ra, họ có thể chỉ định các tham số cấu hình như độ dài của đoạn văn bản kết quả hoặc chi tiết về nhiệm vụ cụ thể.

**Truyền Yêu Cầu đến OpenAI API:** Yêu cầu được chuyển đến các máy chủ của OpenAI thông qua mạng Internet.

**Xử Lý Yêu Cầu bằng Mô Hình Ngôn Ngữ:** OpenAI sử dụng mô hình ngôn ngữ nhân tạo của mình, thường là GPT (Generative Pre-trained Transformer), để xử lý yêu cầu. Mô hình này đã được đào tạo trước đó trên một lượng lớn dữ liệu đa dạng, từ đó nó học được cấu trúc và ngữ cảnh của ngôn ngữ tự nhiên.

**Sinh Phản Hồi (Response):** Sau khi xử lý, OpenAI trả về kết quả thông qua một phản hồi HTTP. Phản hồi này chứa thông tin như văn bản được tạo ra, kết quả của nhiệm vụ cụ thể hoặc mọi thông tin liên quan đến yêu cầu ban đầu.

**Nhận và Sử Dụng Kết Quả:** Người dùng hoặc nhà phát triển có thể nhận được kết quả từ phản hồi và tích hợp chúng vào ứng dụng hoặc dự án của họ. Các kết quả có thể được sử dụng để phân tích ngôn ngữ tự nhiên, tạo ra nội dung sáng tạo, hoặc thực hiện các nhiệm vụ khác liên quan đến xử lý ngôn ngữ tự nhiên.

**Lặp Lại (Optional):** Người dùng hoặc nhà phát triển có thể lặp lại quá trình với các yêu cầu khác nhau để thu được các kết quả khác nhau. Điều này cung cấp linh hoạt cho việc tinh chỉnh và điều chỉnh các tham số để đáp ứng nhu cầu cụ thể của họ.

### 2.2.3. Các tính năng của OpenAI API

Các tính năng chính của OpenAI API bao gồm:

**Xử Lý Ngôn Ngữ Tự Nhiên:** OpenAI API không chỉ giúp xử lý văn bản một cách hiệu quả, mà còn đưa ra khả năng phân tích ngữ cảnh và xử lý ngôn ngữ tự nhiên cao.

**Sinh Văn Bản Tự Nhiên:** Đặc điểm nổi bật của OpenAI API là khả năng sáng tạo khi sinh văn bản, giúp tự động hóa việc tạo ra nội dung đa dạng như bài luận, đoạn văn, hay đối thoại.

**Phân Loại Văn Bản:** API không chỉ phân loại văn bản dựa trên cú pháp, mà còn sử dụng ngôn ngữ tự nhiên để hiểu rõ ý định, mở ra khả năng ứng dụng rộng rãi từ phân loại tin tức đến phân định chủ đề.

**Tạo Đề Xuất:** OpenAI API không chỉ giúp tạo đề xuất văn bản một cách tự động, mà còn cung cấp gợi ý sáng tạo, hỗ trợ người sử dụng và nhà phát triển tạo ra nội dung hấp dẫn.

**Nâng Cao Nội Dung Thông Tin:** API không chỉ trả lời câu hỏi mà còn mô phỏng tình huống và tạo ra văn bản phức tạp, mang lại giá trị gia tăng cho các ứng dụng giáo dục, đàm phán, hay hỗ trợ quyết định.

**Tích Hợp Linh Hoạt:** Thiết kế linh hoạt của OpenAI API cho phép tích hợp dễ dàng vào nhiều ứng dụng và dự án khác nhau, từ chatbot đến hệ thống tư vấn.

**Hỗ Trợ Đa Ngôn Ngữ:** API không chỉ hỗ trợ đa ngôn ngữ để phục vụ nhu cầu toàn cầu mà còn tạo điều kiện cho ứng dụng đa văn hóa.

**Bảo Mật và Quản Lý Sử Dụng:** OpenAI API không chỉ cung cấp tính năng bảo mật mạnh mẽ mà còn giúp người sử dụng kiểm soát chi phí và duy trì tính ổn định của hệ thống thông qua bảng điều khiển chi tiết.

#### 2.2.4. Ứng dụng của OpenAI API

**Tạo nội dung sáng tạo:** API có thể được sử dụng để tạo nội dung sáng tạo mới, như thơ, mã, kịch bản, tác phẩm âm nhạc, email, thư, v.v.

*Ví dụ: OpenAI API có thể được sử dụng để tạo ra các bài báo, bài đăng trên blog, email, thư, v.v. một cách tự động.*

**Dịch thuật ngôn ngữ:** API có thể được sử dụng để dịch văn bản từ một ngôn ngữ sang ngôn ngữ khác.

*Ví dụ: OpenAI API có thể được sử dụng để dịch các tài liệu, trang web, v.v. sang các ngôn ngữ khác nhau.*

**Tóm tắt văn bản:** API có thể được sử dụng để tóm tắt các tài liệu văn bản dài.

*Ví dụ: OpenAI API có thể được sử dụng để tóm tắt các bài báo, nghiên cứu, v.v.*

**Trả lời câu hỏi:** API có thể được sử dụng để trả lời các câu hỏi về thế giới thực.

*Ví dụ: OpenAI API có thể được sử dụng để trả lời các câu hỏi về lịch sử, khoa học.*

**Tạo hình ảnh:** API có thể được sử dụng để tạo hình ảnh từ văn bản bằng Dall-E 3.

*Ví dụ: OpenAI API có thể được sử dụng để tạo ra các hình ảnh minh họa, logo, v.v.*



*Hình 2.5. Ảnh minh họa được tạo bởi Dall-E 3*

## **2.3. Giới thiệu về D-ID API**

### **2.3.1. Khái niệm D-ID API**

D-ID API là một công nghệ tiên tiến cho phép tạo ra video của con người từ một hình ảnh và một tệp âm thanh. Công nghệ này hỗ trợ tạo video đồng bộ với tốc độ render lên đến 100 FPS, nhanh hơn thời gian thực tới 4 lần.

D-ID API chỉ cần một hình ảnh duy nhất để tạo ra một video theo yêu cầu của người dùng. Bất kỳ hình ảnh nào của khuôn mặt đều có thể được sử dụng để tạo ra video nói chuyện với một yêu cầu API đơn giản.

D-ID API cung cấp khả năng tùy chỉnh giọng nói cho AI Presenter bằng cách chọn từ hàng trăm tùy chọn văn bản thành giọng nói với sự hỗ trợ của Microsoft Azure Voices, ngoài ra còn cho phép người dùng tải lên một bản ghi âm mà không cần đăng ký hay thao tác phức tạp.

### **2.3.2. Cách thức hoạt động của D-ID API**

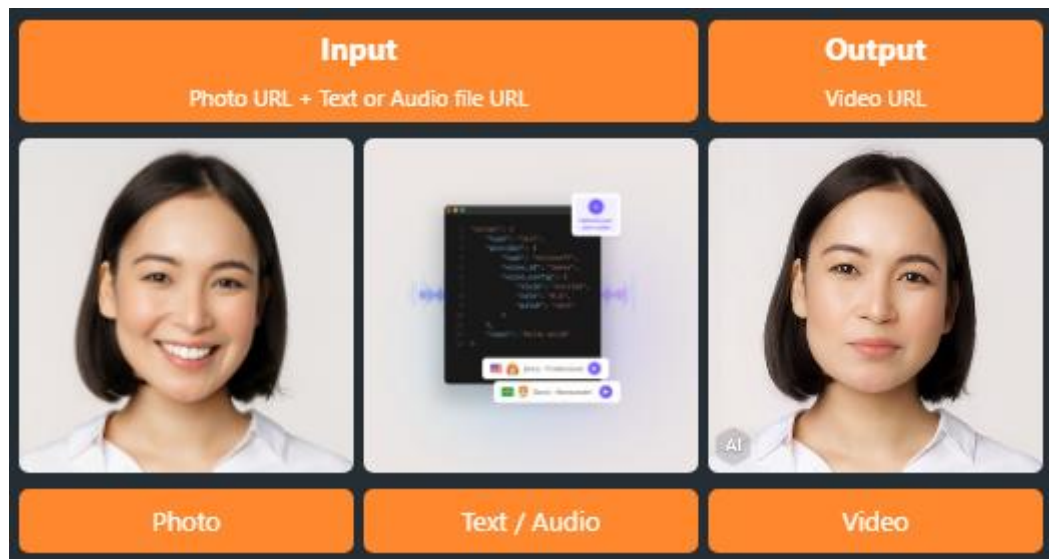
D-ID API hoạt động dựa trên việc tạo ra video của con người từ một hình ảnh và một tệp âm thanh. Đầu tiên, một hình ảnh duy nhất được sử dụng để tạo ra một video “talking head”. Bất kỳ hình ảnh nào của khuôn mặt đều có thể được sử dụng để tạo ra video nói chuyện với một yêu cầu API đơn giản.

Tiếp theo, giọng nói cho AI Presenter (công nghệ biến đổi khuôn mặt theo lời nói) được cung cấp từ các thư viện được D-ID tạo sẵn hoặc tải lên bản ghi âm từ người dùng.

Điều này cho phép người dùng cá nhân hóa video, ở quy mô lớn, trong hơn 100 ngôn ngữ, và không cần kiến thức kỹ thuật.



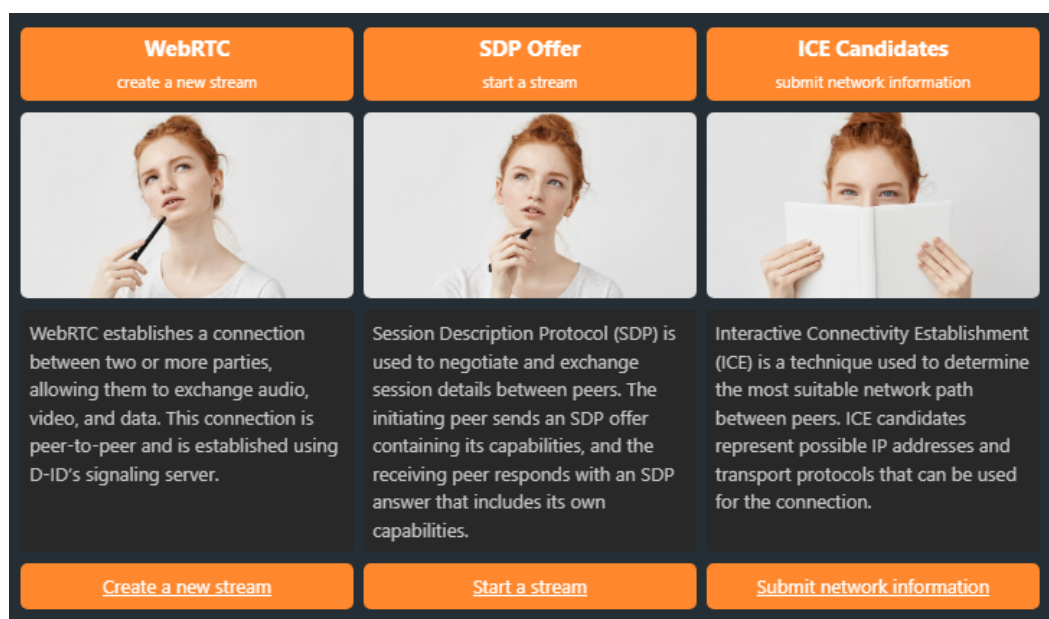
Cuối cùng, D-ID API cho phép tạo video đồng bộ từ tệp âm thanh với tốc độ render lên đến 100 FPS, nhanh hơn thời gian thực tới 4 lần. Điều này cho phép xử lý hàng chục nghìn yêu cầu cùng một lúc mà không gặp phải những vấn đề gì xảy ra.



Hình 2.6. Ảnh minh họa quy trình tạo video từ D-ID API

### 2.3.3. Các tính năng của D-ID API

**Streams:** D-ID API mở rộng tính năng Streams nhằm tăng khả năng tạo video đồng bộ từ một hình ảnh và một tệp âm thanh. Điều này có nghĩa là, từ một hình ảnh khuôn mặt và một tệp âm thanh, có thể tạo ra một video “talking head” và video này sẽ được tạo ra và phát trực tiếp theo thời gian thực.

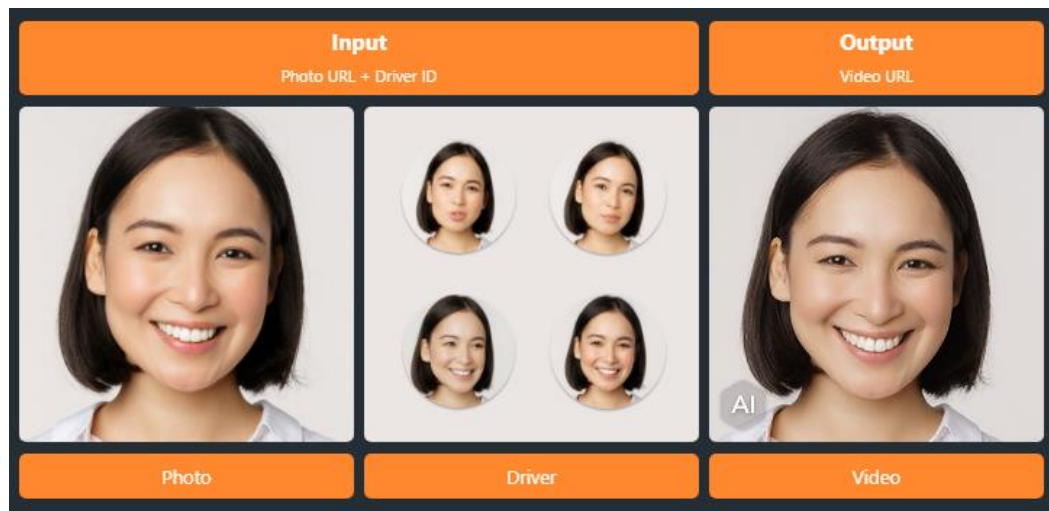


Hình 2.7. Ảnh minh họa quy trình tạo video Streams từ D-ID API

**Animations:** Animations của D-ID API kết hợp sức mạnh của công nghệ hoạt hình khuôn mặt Deep-learning của D-ID với khả năng tạo văn bản của LLM và khả năng



chuyển văn bản thành hình ảnh. Kết quả tạo ra một nền tảng video tiên tiến với sức mạnh của trí tuệ nhân tạo được áp dụng một cách có cấu trúc.



Hình 2.8. Ảnh minh họa quy trình tạo video Animations từ D-ID API

Với tính năng Animations, người dùng có thể tạo ra video chất lượng cao với các nhân vật ảo di chuyển và nói chuyện một cách tự nhiên. Điều này mở ra nhiều khả năng mới trong việc tạo ra nội dung video hấp dẫn và tương tác.

#### 2.3.4. Ứng dụng của D-ID API

D-ID API có nhiều ứng dụng mạnh mẽ trong nhiều lĩnh vực:

Tạo video cuộc gọi thời gian thực: D-ID API có thể được sử dụng để tạo ra hình đại diện cuộc gọi video thời gian thực, mang lại một trải nghiệm cuộc gọi video phong phú và tương tác.

Tích hợp vào trò chơi trực tuyến: D-ID API có thể được thêm vào trò chơi trực tuyến dựa trên nhân vật, tạo ra một trải nghiệm chơi game tương tác và thú vị.

Tạo video cá nhân hóa: D-ID API cho phép tạo video cá nhân hóa ở quy mô lớn, mang lại khuôn mặt con người cho các video truyền thông và L&D.

Nhanh và tiết kiệm chi phí: D-ID API cho phép chuyển đổi các bộ đào tạo hiện có, tài liệu hoặc tệp âm thanh thành nội dung video hấp dẫn với ít nỗ lực.

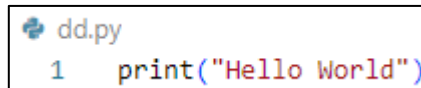
Tạo nội dung đào tạo và học tập đa dạng chỉ với một nút bấm: D-ID API cho phép tạo nội dung đào tạo và học tập đa dạng chỉ với một nút bấm.

Điều chỉnh biểu cảm, kiểu giọng nói, độ cao và tốc độ giọng nói: D-ID API cho phép điều chỉnh biểu cảm, kiểu giọng nói và tốc độ giọng nói, tạo ra video chất lượng cao.

## 2.4. Giới thiệu về Python

### 2.4.1. Khái niệm Python

Python là một ngôn ngữ lập trình thông dụng, dễ học và linh hoạt, được phát triển bởi Guido van Rossum và ra mắt lần đầu tiên vào năm 1991. Python đã trở thành một trong những ngôn ngữ lập trình phổ biến nhất trên thế giới, được sử dụng rộng rãi trong nhiều lĩnh vực khác nhau như phát triển web, khoa học dữ liệu, trí tuệ nhân tạo, đám mây, và nhiều ứng dụng khác.



```
dd.py
1 print("Hello World")
```

Hình 2. 9. Ảnh minh họa cú pháp cơ bản Python

### 2.4.2. Lịch sử phát triển của Python

Ngôn ngữ lập trình Python được hình thành vào cuối thập kỷ 1980 và việc triển khai đã bắt đầu vào tháng 12 năm 1989 bởi Guido van Rossum tại CWI ở Hà Lan như một người kế nhiệm của ABC có khả năng xử lý ngoại lệ và giao tiếp với hệ điều hành Amoeba. Sau khi Python bắt đầu triển khai vào năm 1989, ngôn ngữ này đã trải qua những thay đổi và phát triển đáng kể qua các phiên bản chính:

**Python 1.0 (1994):** Phiên bản đầu tiên của Python được phát hành năm 1994. Nó đã đưa vào ngôn ngữ nhiều tính năng mới như hỗ trợ đối tượng, module, exception, và nhiều thay đổi khác để làm cho ngôn ngữ trở nên mạnh mẽ hơn.

**Python 2.0 (2000):** Phiên bản này mang lại nhiều cải tiến đáng chú ý, bao gồm hệ thống garbage collection, hỗ trợ Unicode, và các tính năng nổi bật khác như list comprehensions. Python 2.0 đã làm cho ngôn ngữ trở nên mạnh mẽ hơn và thân thiện với người lập trình.

**Python 3.0 (2008):** Là một bước quan trọng, Python 3.0 được phát hành với nhiều thay đổi lớn, bao gồm sự thay đổi trong cú pháp và kiến trúc để cải thiện tính đồng nhất và sạch sẽ của ngôn ngữ. Mặc dù đây là một bước tiến lớn, Python 3.0 không tương thích ngược với các phiên bản trước đó, điều này đã tạo ra một sự chia rẽ trong cộng đồng Python.

### 2.4.4. Ứng dụng tiêu biểu của Python

**Phát Triển Web:** Python là ngôn ngữ lập trình phổ biến trong lĩnh vực phát triển web do tính đơn giản và linh hoạt của mã nguồn. Việc sử dụng Python giúp nhà phát triển xây dựng các trang web với mã nguồn đẹp mắt và chức năng một cách hiệu quả.

**Trí Tuệ Nhân Tạo và Học Máy:** Python là một trong những ngôn ngữ hàng đầu được sử dụng trong lĩnh vực trí tuệ nhân tạo (AI) và học máy (ML). Sự đa dạng của các thư viện như NumPy, Pandas, Matplotlib, và scikit-learn giúp nhà phát triển xây dựng, đào tạo, và triển khai mô hình máy học một cách dễ dàng. Cộng đồng mạnh mẽ xung quanh Python cung cấp nền tảng để chia sẻ kiến thức và tạo ra các dự án AI/ML động.

**Phát Triển Game:** Python không chỉ là lựa chọn cho phát triển web mà còn được ứng dụng sáng tạo trong việc xây dựng trò chơi. Pygame và Pykyra là những thư viện hỗ trợ việc phát triển game, giúp nhà phát triển tập trung vào logic trò chơi thay vì chi tiết kỹ thuật.

**Phát Triển GUI Desktop:** Sự đơn giản và tính độc lập của Python trở thành lựa chọn tốt cho việc phát triển ứng dụng Desktop với giao diện đồ họa người dùng (GUI). Tkinter và PyQt là những thư viện phổ biến cho việc xây dựng các ứng dụng desktop thân thiện với người dùng.

**Ứng Dụng Doanh Nghiệp/Kinh Doanh:** Python được sử dụng trong phát triển các ứng dụng doanh nghiệp với mục tiêu giữ cho người dùng cuối cảm thấy thoải mái và dễ sử dụng. Các framework như Django cung cấp giải pháp toàn diện cho việc xây dựng ứng dụng doanh nghiệp mạnh mẽ.

**Phát Triển Ứng dụng:** Python không chỉ là một ngôn ngữ lập trình, mà còn là một cộng đồng và hệ sinh thái phong phú với nhiều thư viện và gói mở rộng. Điều này làm cho nó trở thành công cụ lý tưởng cho việc phát triển và duy trì ứng dụng.

**Mục Đích Giáo Dục:** Python không chỉ là một ngôn ngữ lập trình, mà còn là một công cụ giáo dục mạnh mẽ. Sự đơn giản và đọc được của cú pháp Python trở thành ngôn ngữ lập trình phổ biến cho người mới bắt đầu và dễ dàng tiếp cận khái niệm cơ bản của lập trình một cách thuận tiện.

## 2.5. Giới thiệu về Framework PyQt5

### 2.5.1. Khái niệm về Framework PyQt5

PyQt5 là một bộ công cụ dùng để phát triển ứng dụng GUI, được xây dựng dựa trên Qt v5, một thư viện GUI đa nền tảng. PyQt5 chọn Python trở thành ngôn ngữ chính để phát triển ứng dụng, thay thế cho C++, trên mọi nền tảng hỗ trợ, bao gồm cả iOS và Android. PyQt5 được phát triển và bảo dưỡng bởi Riverbank Computing, một công ty có trụ sở tại Anh. Bên cạnh đó, PyQt5 còn hỗ trợ nhiều công cụ và thư viện khác nhau, giúp tăng cường khả năng tương tác và tính năng của các ứng dụng.

Một ứng dụng GUI thường bao gồm hai phần chính: Front-end (giao diện người dùng) và Back-end (xử lý dữ liệu). PyQt5 cung cấp một công cụ tên là 'QtDesigner' giúp thiết kế giao diện người dùng một cách trực quan thông qua việc kéo và thả các thành phần, giúp tăng tốc quá trình phát triển và giúp người dùng tập trung nhiều hơn vào việc xử lý dữ liệu.

### 2.5.2. Các thành phần cơ bản của Framework PyQt5

**Qt Core Module:** Đây là nền tảng cốt lõi của PyQt5, chứa các lớp cơ bản nhất như QObject, QVariant, QTimer, cung cấp các cấu trúc dữ liệu cơ bản như danh sách, danh sách liên kết, và nhiều hơn nữa. QObject là lớp cơ sở cho hầu hết các lớp trong Qt và QVariant là lớp cơ sở cho hầu hết các loại dữ liệu có thể được sử dụng với QObject.

**Qt GUI Module:** Bao gồm các lớp để tạo giao diện người dùng, xử lý sự kiện, và đồ họa. Chứa lớp QWidget là lớp cơ bản để tạo giao diện người dùng. QWidget cung cấp các phương thức để quản lý vị trí, kích thước, và hình dạng của các widget.

**Qt Widgets Module:** Bao gồm nhiều widget như buttons, text boxes, sliders, và nhiều hơn nữa. Các widget này có thể được sắp xếp để tạo giao diện người dùng phức tạp. Ví dụ, QPushButton là một widget cung cấp một nút có thể nhấn, QLineEdit cung cấp một hộp văn bản một dòng, và QSlider cung cấp một thanh trượt.

**Qt Network Module:** Cung cấp các lớp để làm việc với mạng, bao gồm: HTTP, TCP, và UDP.

**Qt SQL Module:** Cho phép kết nối và tương tác với các cơ sở dữ liệu SQL. QSqlDatabase là lớp cung cấp giao diện để làm việc với cơ sở dữ liệu SQL.

**Qt Multimedia Module:** Hỗ trợ các tính năng đa phương tiện nổi bật như: âm thanh và video.

**Qt XML Module:** Hỗ trợ đọc và viết dữ liệu dưới định dạng XML. QDomDocument là một lớp cung cấp các phương thức để đọc và viết dữ liệu XML.

**Qt WebKit Module:** Cho phép tích hợp trình duyệt web vào ứng dụng PyQt5.

### 2.5.3. Ứng dụng Framework PyQt5 trong Python

**Xây Dựng Giao Diện Người Dùng (GUI):** PyQt5 cung cấp các công cụ và lớp để thiết kế và xây dựng giao diện người dùng đồ họa. Qt Designer có thể được sử dụng để tạo giao diện một cách dễ dàng và sau đó tích hợp vào mã nguồn Python.

**Xử Lý Sự Kiện:** PyQt5 cho phép xử lý sự kiện người dùng như nhấn nút, di chuyển chuột, và nhập liệu. Việc này giúp ứng dụng phản hồi tốt đối với tương tác người dùng.

**Kết Nối Cơ Sở Dữ Liệu:** Qt SQL Module có thể được sử dụng để kết nối và tương tác với cơ sở dữ liệu SQL. Việc này cho phép lưu trữ và truy xuất dữ liệu từ cơ sở dữ liệu một cách thuận tiện.

**Phát Triển Ứng Dụng Đa Nhiệm:** PyQt5 hỗ trợ việc phát triển ứng dụng đa nhiệm với việc sử dụng các luồng (threads). Điều này giúp ứng dụng chạy mượt mà và không bị đóng đinh do các tác vụ nặng.

**Tích Hợp Đa Phương Tiện:** Qt Multimedia Module có thể được sử dụng để xử lý và phát nhạc, âm thanh, video trong ứng dụng. Việc này tạo trải nghiệm đa phương tiện cho người dùng.

**Phát Triển Ứng Dụng Web:** Qt WebKit (hoặc Qt WebEngine) có thể được tích hợp để nhúng trình duyệt web vào ứng dụng Python. Việc này cho phép hiển thị và tương tác với nội dung web trong ứng dụng desktop.

**Tạo Ứng Dụng Đa Nền Tảng:** PyQt5 hỗ trợ việc phát triển ứng dụng đa nền tảng, có thể chạy trên nhiều hệ điều hành như Windows, macOS, và Linux.

**Triển Khai Ứng Dụng:** PyQt5 cung cấp công cụ để triển khai ứng dụng Python thành các ứng dụng độc lập có thể chạy trên máy tính mà không cần cài đặt Python và thiết lập PyQt5.

## CHƯƠNG 3: HIỆN THỰC HÓA NGHIÊN CỨU

### 3.1. Đặt tả ứng dụng

Ứng dụng được thiết kế nhằm mục đích đáp ứng nhu cầu của người dùng muốn tạo video quảng cáo sản phẩm một cách nhanh chóng và dễ dàng, mà không cần phải có kỹ năng chuyên môn về dựng video. Đối tượng người dùng mục tiêu có thể bao gồm các nhà bán lẻ trực tuyến, các doanh nghiệp vừa và nhỏ, hoặc bất kỳ ai muốn quảng bá sản phẩm của mình một cách hiệu quả.

Ứng dụng sử dụng công nghệ AI tiên tiến, cùng với cấu trúc Prompt Engineering chuyên nghiệp được tích hợp sẵn, để tạo ra mô tả sản phẩm hấp dẫn bằng tiếng Việt. Người dùng chỉ cần nhập tên sản phẩm, và ứng dụng sẽ tạo ra mô tả sản phẩm dựa trên mô hình AI ChatGPT 3.5.

Ngoài ra, ứng dụng còn cho phép người dùng chọn giọng nói nam hoặc nữ cho video của họ thông qua công nghệ D-ID. Điều này giúp tạo ra video quảng cáo phù hợp với đối tượng khách hàng mà người dùng muốn tiếp cận.

Sau khi tạo video, ứng dụng cho phép người dùng xem trực tiếp video ngay trong ứng dụng. Điều này giúp người dùng dễ dàng xem lại và đánh giá chất lượng video trước khi phát hành.

Với những chức năng này, ứng dụng giúp người dùng tận dụng công nghệ AI để tạo ra các video quảng cáo sản phẩm chất lượng mà không cần kỹ năng chuyên môn về dựng video. Điều này không chỉ giúp tiết kiệm thời gian và công sức, mà còn đảm bảo rằng video quảng cáo luôn sáng tạo, hấp dẫn và phù hợp với khách hàng mục tiêu.

### 3.2 Xây dựng giao diện người dùng GUI với PyQt5 qua Python

Ứng dụng được xây dựng bằng ngôn ngữ Python và thư viện PyQt5. Giao diện người dùng của ứng dụng bao gồm các thành phần sau:

**Nhập tên sản phẩm:** Người dùng nhập tên sản phẩm mà họ muốn tạo video quảng cáo. Đây là thông tin quan trọng nhất, vì nó sẽ quyết định nội dung của video quảng cáo.

**Tạo prompt:** Người dùng nhấn nút này để tạo một prompt cho mô hình AI. Prompt này bao gồm tên sản phẩm mà người dùng đã nhập. Sau khi nhấn nút, ứng dụng sẽ gửi thông tin tên sản phẩm này đến cấu trúc Prompt Engineering đã được tạo sẵn để gửi đến mô hình AI ChatGPT 3.5.

**Chọn giọng nói:** Người dùng chọn giọng nói nam hoặc nữ cho video của họ. Điều này giúp tạo ra video quảng cáo phù hợp với khách hàng mà người dùng muốn tiếp cận.

**Tạo video:** Người dùng nhấn nút này để tạo video quảng cáo. Sau khi nhấn nút, ứng dụng sẽ bắt đầu xử lý thông tin và tạo video.

```

1  def __init__(self):
2      super().__init__()
3      self.video_widget = QVideoWidget()
4      self.player = QMediaPlayer(None, QMediaPlayer.VideoSurface)
5      self.player.setVideoOutput(self.video_widget)
6      self.setWindowTitle("AI Streamer")
7      self.button = QPushButton('Tạo prompt')
8      self.video_button = QPushButton('Tạo video')
9      self.input_text = QLineEdit()
10     self.label = QLabel()
11     self.output_text = QTextEdit()
12     self.voice_combobox = QComboBox()
13     self.voice_combobox.addItem("Nam")
14     self.voice_combobox.addItem("Nữ")
15     self.voice_combobox.setFixedWidth(100)
16     self.voice_label = QLabel("Chọn voice:")
17     self.setup_ui()
18
19     def setup_ui(self):
20         central_widget = QWidget()
21         central_layout = QHBoxLayout()
22         left_layout = QVBoxLayout()
23         right_layout = QVBoxLayout()
24
25         voice_layout = QHBoxLayout()
26         voice_layout.addStretch(1) # Add a stretchable space before the label and combobox
27         voice_layout.addWidget(self.voice_label)
28         voice_layout.addWidget(self.voice_combobox)
29
30         left_layout.addWidget(self.label)
31         left_layout.addWidget(self.input_text)
32         left_layout.addWidget(self.output_text)
33         left_layout.addLayout(voice_layout)
34         left_layout.addWidget(self.button)
35         left_layout.addWidget(self.video_button)
36
37         right_layout.addWidget(self.video_widget)
38
39         central_layout.addLayout(left_layout)
40         central_layout.addLayout(right_layout)
41
42         central_widget.setLayout(central_layout)
43         self.setCentralWidget(central_widget)
44         self.button.clicked.connect(self.create_prompt)
45         self.video_button.clicked.connect(self.create_video_and_play)
46         self.label.setText('Nhập nội dung vào đây:')
47         self.showMaximized()
48
49
50         central_layout.setStretchFactor(left_layout, 1)
51         central_layout.setStretchFactor(right_layout, 1)

```

*Hình 3.1. Hàm xây dựng giao diện người dùng*

Để tập hợp và kết nối các sự kiện và người dùng click chọn lại với nhau, ta dùng hàm `clicked.connect()` là một hàm trong thư viện PyQt5. Hàm này được sử dụng để kết nối một sự kiện với một hàm xử lý. Trong trường hợp này, sự kiện là nhấn nút `create_video_button`, và hàm xử lý là `create_video_and_play()`. Hàm xử lý cũng thực hiện việc tạo video và phát video.

```

1      # Kết nối sự kiện nhấn nút tạo video
2      self.create_video_button.clicked.connect(self.create_video_and_play)
3
4      def create_video_and_play(self):
5
6          # Lấy thông tin sản phẩm từ hộp văn bản
7          product_info = self.product_info_edit.text()
8
9          # Lấy giọng nói từ hộp kết hợp
10         voice_id = "vi-VN-NamMinhNeural" if self.voice_combobox.currentText() == "Nam" else "vi-VN-HoaiMyNeural"
11
12         # Tạo video
13         video_id = self.create_video(product_info)
14
15         # Lấy URL của video
16         video_url = self.get_video_url(video_id)
17
18         # Tải và phát video
19         self.download_and_play_video(video_url, self.player)

```

Hình 3.2. Hàm kết nối các sự kiện

### 3.3 Tích hợp OpenAI API và xử lý dữ liệu JSON

OpenAI API cung cấp một loạt các mô hình ngôn ngữ lớn, bao gồm cả mô hình GPT-3.5-turbo, được sử dụng trong ứng dụng này. Để tích hợp OpenAI API vào ứng dụng, thư viện OpenAI được sử dụng. Thư viện này cung cấp các hàm và phương thức để tương tác với OpenAI API.

```

1  #Hàm kết nối đến OpenAI API
2  import openai
3  from cachetools import cached, TTLCache
4
5  authorization = "Basic Y0d4aG3uUnpkR2gxWTJGa1FHZHRZV2xzTG10dmJR0jRyZTdDUE1PZTQzOUFXSmk2WXhMg=="
6  client = openai.OpenAI(api_key="sk-9He8zdI1ZkbXL1FPaNBGT381bkFJTz7HDatCxxuTMyoDQPyZ")
7
8  @cached(cache)
9  def get_product_info(product_name):
10     completion = client.chat.completions.create(
11         model="gpt-3.5-turbo",
12         messages=[
13             {"role": "system", "content": "You are a helpful assistant."},
14             {"role": "user", "content": f"Translate into Vietnamese the following content: Viết đoạn giới thiệu về sản phẩm {product_name}"},
15         ],
16         max_tokens=500
17     )
18     return completion.choices[0].message.content.strip()

```

Hình 3.3. Hàm kết nối OpenAI API

```

1  import requests
2  import json
3
4  def create_video(self, product_info):
5      # ... (code tạo payload JSON)
6
7      response = requests.post(url, json=payload, headers=headers)
8      response_data = response.json() # Chuyển đổi dữ liệu JSON từ API D-ID
9
10     # ... (xử lý dữ liệu JSON)
11
12     def get_video_url(self, video_id):
13         # ... (gửi yêu cầu GET đến API D-ID)
14
15         response_data = response.json() # Chuyển đổi dữ liệu JSON từ API D-ID
16
17         # ... (xử lý dữ liệu JSON)

```

Hình 3.4. Hàm xử lý dữ liệu JSON



Để xử lý dữ liệu JSON từ D-ID API, thư viện requests và hàm json() được sử dụng. Thư viện requests được sử dụng để gửi các yêu cầu HTTP đến D-ID API, và hàm json() được sử dụng để chuyển đổi dữ liệu từ định dạng JSON sang các đối tượng Python.

Việc tích hợp OpenAI API và xử lý dữ liệu JSON là hai thành phần quan trọng của ứng dụng tạo video quảng cáo sản phẩm. Các thành phần này giúp ứng dụng tạo ra các video quảng cáo hấp dẫn và sáng tạo, đồng thời tối ưu hiệu suất của ứng dụng.

### 3.4 Tích hợp D-ID API để tạo video

D-ID API cung cấp các tùy chỉnh chuyển đổi văn bản, mô tả sản phẩm thành video dựa trên cấu trúc JSON được xây dựng sẵn:

```
1 {
2   "script": {
3     "type": "text",
4     "subtitles": "false",
5     "provider": {
6       "type": "microsoft",
7       "voice_id": "vi-VN-NamMinhNeural"
8     },
9     "ssml": "false",
10    "input": "Văn bản mô tả sản phẩm"
11  },
12  "config": {
13    "fluent": "false",
14    "pad_audio": "0.0",
15    "stitch": "true"
16  },
17  "source_url": "https://create-images-results.d-id.com/DefaultPresenters/santa_f_ai/image.jpeg"
18 }
```

Hình 3.5. Cấu trúc JSON của D-ID API

Ứng dụng sẽ gửi yêu cầu POST đến D-ID API với các thông tin sau:

- Nội dung văn bản mô tả sản phẩm.
- Giọng nói: Nam hoặc nữ.
- Hình ảnh đại diện.

D-ID API sẽ sử dụng các thông tin này để tạo ra một video quảng cáo mới. Video sẽ bao gồm các nội dung sau:

- Văn bản mô tả sản phẩm được đọc bởi giọng nói đã chọn.
- Hình ảnh đại diện được sử dụng làm nền cho video.
- Cấu trúc yêu cầu.

Giải thích các trường:

- script: Trường này chứa thông tin về nội dung văn bản của video. Các trường con của trường này có ý nghĩa như sau:
- type: Loại nội dung, trong trường hợp này là "text".

- subtitles: Liệu có chèn phụ đề vào video hay không.
- provider: Nhà cung cấp giọng nói.
- voice\_id: ID của giọng nói.
- ssml: Liệu có sử dụng định dạng SSML cho văn bản hay không.
- input: Văn bản mô tả sản phẩm.
- config: Trường này chứa các thông số cấu hình cho video. Các trường con của trường này có ý nghĩa như sau:
  - fluent: Liệu sử dụng giọng nói tự nhiên hay không.
  - pad\_audio: Liệu chèn âm thanh bổ sung vào video hay không.
  - stitch: Liệu nối các đoạn âm thanh lại với nhau hay không.
- source\_url: URL của hình ảnh đại diện.

Dựa vào việc gửi POST đến D-ID API, API D-ID sẽ trả về một kết quả JSON với các thông tin sau:

1 id: ID của video được tạo.

Hình 3.6. Kết quả trả về từ lệnh POST của D-ID API

Để gửi yêu cầu tạo video, ứng dụng sẽ sử dụng hàm `create_video()`, hàm này sẽ lấy các thông tin cần thiết từ người dùng, bao gồm nội dung văn bản mô tả sản phẩm được minh họa như sau:

```

1 def create_video(self, product_info):
2     voice_id = "vi-VN-NamMinhNeural" if self.voice_combobox.currentText() == "Nam" else "vi-VN-HoaiMyNeural"
3     url = "https://api.d-id.com/talks"
4     payload = {
5         "script": {
6             "type": "text",
7             "subtitles": "false",
8             "provider": {
9                 "type": "microsoft",
10                "voice_id": voice_id
11            },
12            "ssml": "false",
13            "input": product_info
14        },
15        "config": {
16            "fluent": "false",
17            "pad_audio": "0.0",
18            "stitch": "true"
19        },
20        "source_url": "https://create-images-results.d-id.com/DefaultPresenters/santa_f_ai/image.jpeg"
21    }
22    headers = {
23        "accept": "application/json",
24        "content-type": "application/json",
25        "authorization": authorization
26    }
27    response = requests.post(url, json=payload, headers=headers)
28    response_data = response.json()
29    video_id = response_data.get('id')
30    return video_id

```

Hình 3.7. Hàm gửi yêu cầu GET đến D-ID API

### 3.5 Ứng dụng thư viện Cachetools để lưu cache và xuất video

Để tối ưu hiệu suất của ứng dụng, thư viện Cachetools được sử dụng để lưu cache các kết quả của API D-ID. Cache sẽ được lưu trữ trong bộ nhớ cục bộ của máy tính, giúp giảm thiểu số lần cần gửi yêu cầu đến API.

Cachetools là một thư viện Python cung cấp các hàm và lớp để tạo và quản lý cache. Thư viện này cung cấp nhiều loại cache khác nhau, bao gồm:

- **LRU cache:** Loại cache này sẽ xóa các mục cũ nhất khi bộ nhớ cache đầy.
- **LFU cache:** Loại cache này sẽ xóa các mục ít được truy cập nhất khi bộ nhớ cache đầy để giải phóng bộ nhớ.
- **FIFO cache:** Loại cache này sẽ xóa các mục được thêm vào sớm nhất khi bộ nhớ cache đầy.

Ứng dụng sử dụng loại cache LRU với thời gian hết hạn là 30 phút. Điều này có nghĩa là các kết quả của API D-ID sẽ được lưu cache trong bộ nhớ cục bộ trong tối đa 30 phút. Sau 30 phút, các kết quả này sẽ bị xóa để giải phóng bộ nhớ.

Để sử dụng Cachetools trong ứng dụng, tiến hành bổ sung thư viện này vào tệp tin requirements.txt. Sau đó, có thể sử dụng các hàm và lớp của thư viện này để tạo và quản lý cache.

```

1 def create_video(self, product_info):
2     voice_id = "vi-VN-NamMinhNeural" if self.voice_combobox.currentText() == "Nam" else "vi-VN-HoaiMyNeural"
3     url = "https://api.d-id.com/talks"
4
5     # Lấy kết quả từ cache
6     video_id = cachetools.LRUCache(maxsize=100, ttl=300).get(product_info, default=None)
7     if video_id is not None:
8         return video_id
9
10    # Nếu không có kết quả trong cache, gửi yêu cầu đến API D-ID
11    payload = {
12        "script": {
13            "type": "text",
14            "subtitles": "false",
15            "provider": {
16                "type": "microsoft",
17                "voice_id": voice_id
18            },
19            "ssml": "false",
20            "input": product_info
21        },
22        "config": {
23            "fluent": "false",
24            "pad_audio": "0.0",
25            "stitch": "true"
26        },
27        "source_url": "https://create-images-results.d-id.com/DefaultPresenters/santa_f_ai/image.jpeg"
28    }
29    headers = {
30        "accept": "application/json",
31        "content-type": "application/json",
32        "authorization": authorization
33    }
34    response = requests.post(url, json=payload, headers=headers)
35    response_data = response.json()
36    video_id = response_data.get('id')
37
38    # Lưu kết quả vào cache
39    cachetools.LRUCache(maxsize=100, ttl=300).set(product_info, video_id)
40
41    return video_id
42

```

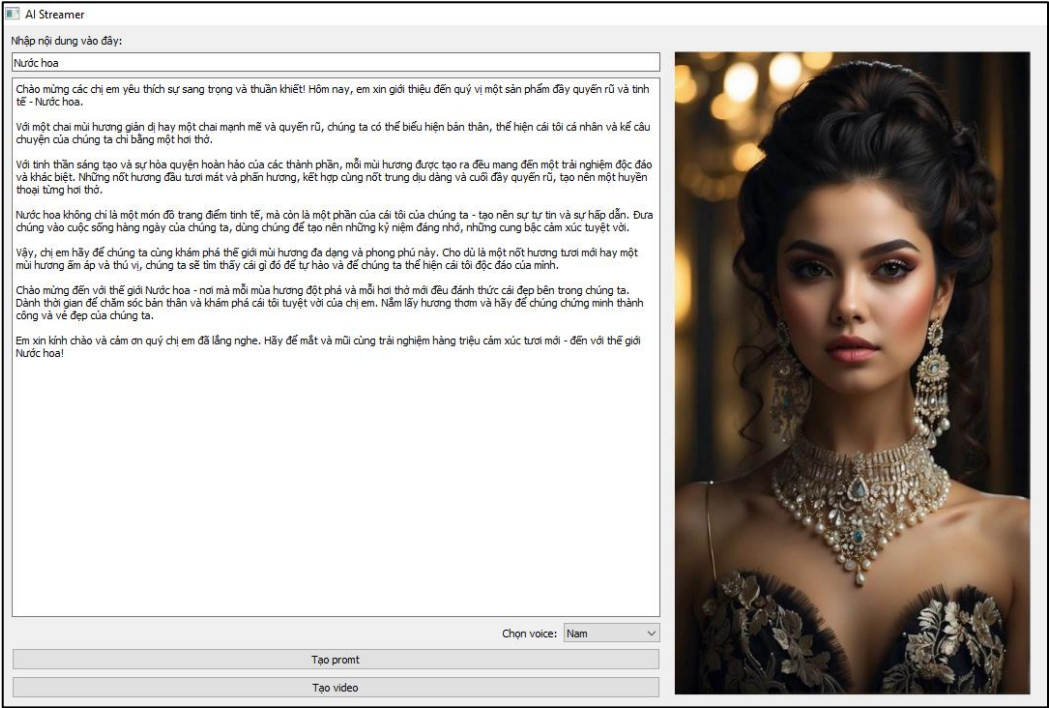
Hình 3. 8. Hàm `create_video()` sử dụng Cachetools

## CHƯƠNG 4: KẾT QUẢ NGHIÊN CỨU

### 4.1. Dữ liệu thử nghiệm

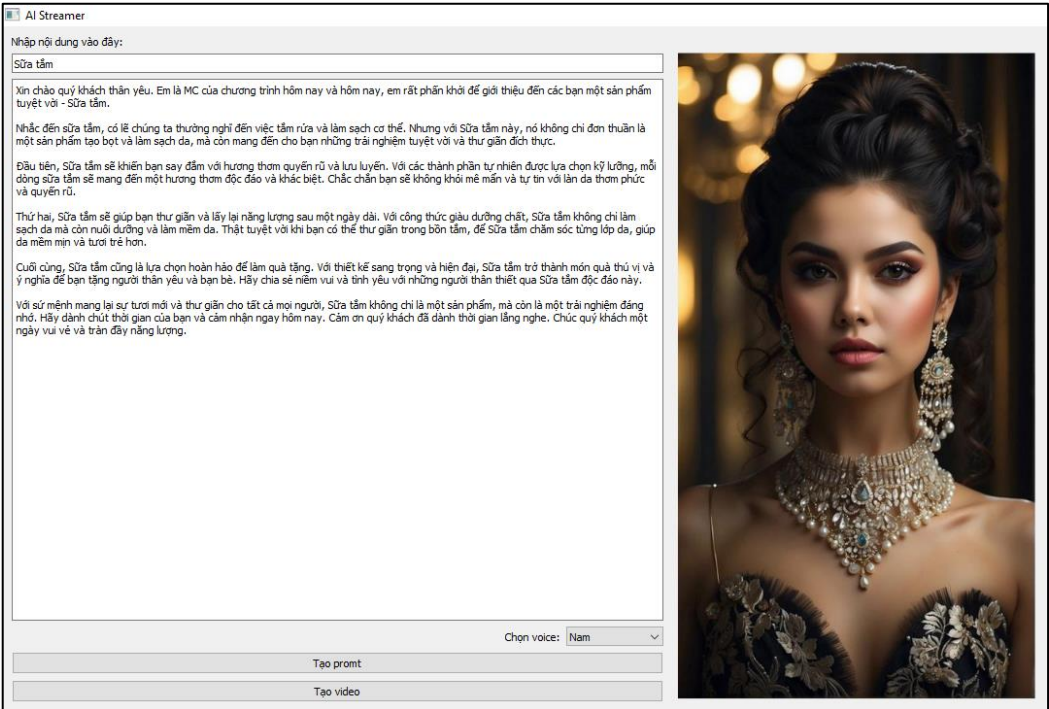
Trong nghiên cứu này, tập dữ liệu thử nghiệm được sử dụng bao gồm tên 3 sản phẩm tiêu biểu cùng với kết quả trả về từ các thông tin mô tả sản phẩm khác nhau.

- Dữ liệu thử nghiệm về sản phẩm: Nước hoa.



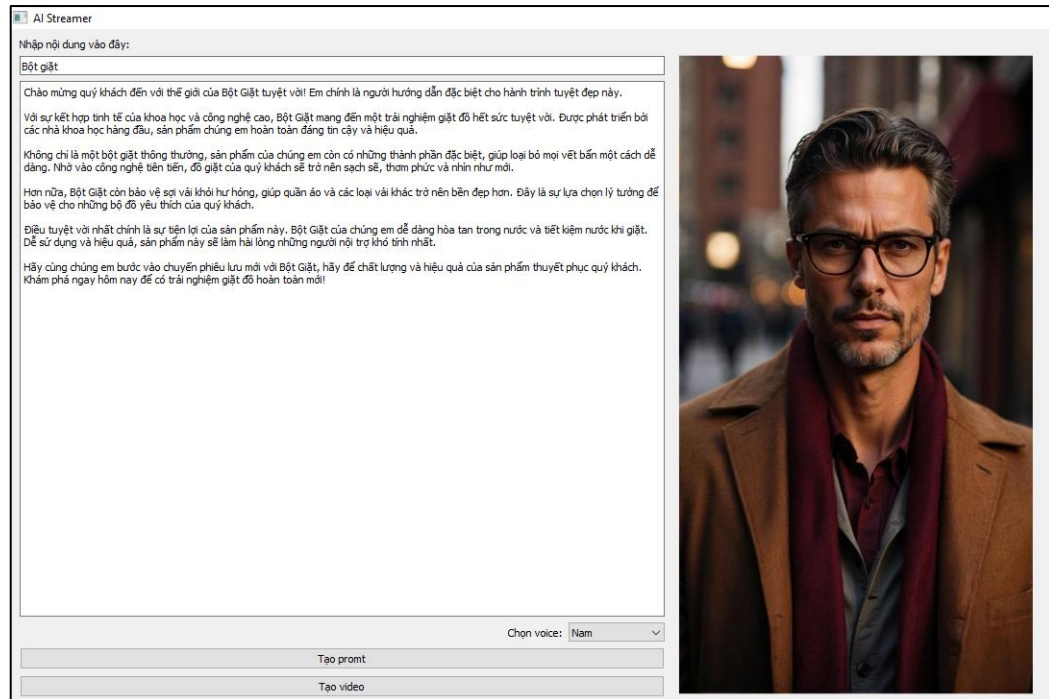
Hình 4.1. Dữ liệu thử nghiệm về sản phẩm nước hoa

- Dữ liệu thử nghiệm về sản phẩm: Sữa tắm.



Hình 4.2. Dữ liệu thử nghiệm về sản phẩm sữa tắm

- Dữ liệu thử nghiệm về sản phẩm: Bột giặt.



Hình 4.3. Dữ liệu thử nghiệm về sản phẩm bột giặt

## 4.2. Cài đặt môi trường

Để ứng dụng hoạt động một cách chính xác, người dùng cần thiết lập các môi trường cần thiết. Trong nghiên cứu này, các môi trường cần được cài đặt bao gồm:

- **Cài Đặt ffmpeg:** Người dùng cần cài đặt ffmpeg để hỗ trợ xử lý và mã hóa video.
- **Cài Đặt K-Lite Codec Pack:** K-Lite Codec Pack là một bộ codec hữu ích để giải mã và mã hóa các định dạng âm thanh và video khác nhau.
- **Cài Đặt Python 3.12.1:** Python là ngôn ngữ lập trình chính được sử dụng trong dự án. Người dùng cần cài đặt phiên bản Python 3.12.1.
- **Cài Đặt Visual Studio Code:** Visual Studio Code là một trình biên tập mã nguồn mở và linh hoạt, được sử dụng để phát triển ứng dụng.

Sau khi cài đặt các môi trường cần thiết, người dùng có thể thực hiện các bước sau:

**Bước 1:** Truy cập đường link để clone dự án: <https://github.com/ThanhTann/csn-da21ttb-duongthanhtan-aistreamer-python>

**Bước 2:** Mở dự án bằng Visual Studio Code.

**Bước 3:** Mở Terminal bằng cách nhấn tổ hợp Ctrl + ~.

**Bước 4:** Nhập:

- Set-ExecutionPolicy Unrestricted -Scope Process
- Set-ExecutionPolicy Unrestricted -Force

- Set-ExecutionPolicy -ExecutionPolicy RemoteSigned -Scope CurrentUser

**Bước 4:** Gõ các lệnh sau để cài đặt các thư viện Python:

- pip install PyQt5
- pip install requests
- pip install Cachetools

**Bước 5:** Sau khi cài đặt xong các môi trường cần thiết, người dùng có thể kiểm tra xem môi trường đã được cài đặt thành công hay chưa bằng cách mở Command Prompt và chạy lệnh:

- python -V

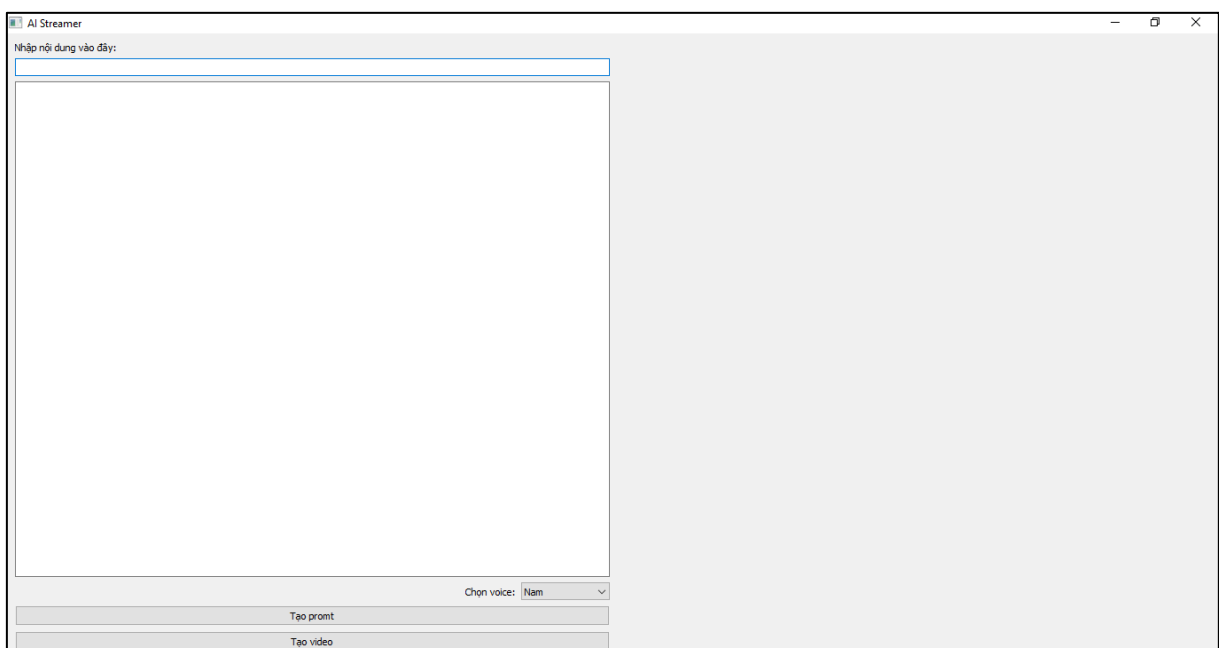
**Bước 6:** Khi môi trường đã được cài đặt thành công, người dùng có thể chạy chương trình bằng cách nhấn **Ctrl + F5**.

Ngoài ra, người dùng và các nhà phát triển cũng cần chú ý đến cấu hình máy tính tối thiểu để ứng dụng có thể hoạt động một cách ổn định:

- **Hệ điều hành:** Windows 10.
- **RAM:** 4GB.
- **Dung lượng ổ cứng (HDD hoặc SSD):** 1GB.
- **CPU:** i3 - 2375M (tối thiểu).

#### 4.3. Thiết kế giao diện

Trong nghiên cứu này, giao diện của ứng dụng được thiết kế với tông màu xám làm chủ đạo. Giao diện được bố trí đơn giản, thuận tiện cho người dùng thao tác.



Hình 4. 4. Giao diện ứng dụng AI Streamer



#### 4.4. Các chức năng của hệ thống

Ứng dụng có các chức năng theo đạt tả ứng dụng đã đề xuất. Trong nghiên cứu này, các chức năng của ứng dụng bao gồm:

- **Nhập văn bản mô tả sản phẩm:** Chức năng này cho phép người dùng nhập mô tả chi tiết về sản phẩm mà họ muốn quảng cáo. Người dùng có thể nhập thông tin về tên sản phẩm, tính năng nổi bật, giá cả, và bất kỳ thông tin nào khác mà họ muốn hiển thị trong video quảng cáo.
- **Chọn giọng đọc:** Người dùng có thể chọn giọng đọc cho video quảng cáo của mình. Có nhiều lựa chọn về giọng đọc, bao gồm cả giọng nam và nữ, với nhiều ngôn ngữ và giọng địa phương khác nhau. Điều này giúp video quảng cáo phù hợp hơn với khán giả mục tiêu.
- **Tạo video quảng cáo:** Sau khi đã nhập văn bản mô tả sản phẩm và chọn giọng đọc, người dùng có thể tạo video quảng cáo. Ứng dụng sẽ tự động tạo ra video dựa trên thông tin đã nhập, bao gồm cả hình ảnh, âm thanh, và hiệu ứng.
- **Xuất video ra file:** Cuối cùng, sau khi video quảng cáo đã được tạo, người dùng có thể xuất video ra file. Điều này cho phép người dùng dễ dàng chia sẻ video qua email, mạng xã hội, hoặc bất kỳ nền tảng nào khác.

Tất cả các chức năng này đều được thiết kế để đáp ứng nhu cầu quảng cáo sản phẩm của người dùng một cách hiệu quả và dễ dàng. Với ứng dụng này, việc tạo video quảng cáo chưa bao giờ dễ dàng hơn.

## CHƯƠNG 5: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

### 5.1. Kết luận

Qua quá trình thử nghiệm trên dữ liệu đa dạng, ứng dụng đã chứng minh được khả năng hoạt động ổn định và tạo ra những video quảng cáo chất lượng cao.

Với các chức năng như: nhập văn bản mô tả sản phẩm, lựa chọn giọng đọc, chọn hình ảnh đại diện, tạo và xuất video ra file, ứng dụng không chỉ đơn giản hóa quy trình tạo video mà còn mang lại nhiều lợi ích đáng kể cho doanh nghiệp. Việc tiết kiệm thời gian và chi phí, cùng khả năng tùy chỉnh cao, giúp người dùng dễ dàng thao tác hơn.

Điểm mạnh của ứng dụng là sự tích hợp công nghệ AI, tạo ra những video sống động và thu hút. Đồng thời, việc cung cấp nhiều tùy chọn cho người dùng và khả năng sử dụng trên mọi quy mô doanh nghiệp, làm cho ứng dụng trở thành một công cụ linh hoạt và mạnh mẽ.

Tóm lại, ứng dụng không chỉ là một công cụ hữu ích cho việc quảng cáo sản phẩm mà còn mở ra kỷ nguyên mới trong lĩnh vực tiếp cận và thu hút đối tượng khách hàng. Khả năng tương tác và sáng tạo của ứng dụng đặt ra tiêu chuẩn mới cho quảng cáo sản phẩm trực tuyến.

### 5.2. Hạn chế

Mặc dù ứng dụng đã đạt được nhiều kết quả đáng kể, nhưng trong quá trình thực hiện thì ứng dụng vẫn còn một số hạn chế cần được khắc phục để nâng cao trải nghiệm người dùng. Cụ thể như sau:

**Hiệu suất của ứng dụng:** Một trong những hạn chế lớn nhất hiện tại là hiệu suất của ứng dụng, đặc biệt là khi xử lý các video có độ dài lớn. Điều này có thể gây ra sự bất tiện cho người dùng khi họ cần tạo ra các video quảng cáo dài hơn. Việc tối ưu hóa hiệu suất của ứng dụng sẽ là một trong những ưu tiên hàng đầu trong quá trình phát triển tiếp theo.

**Giao diện của ứng dụng:** Giao diện hiện tại của ứng dụng còn đơn giản và cần được tối ưu hóa hơn nữa để tạo ra một trải nghiệm người dùng tốt hơn. Việc cải thiện giao diện sẽ giúp người dùng dễ dàng tìm hiểu và sử dụng ứng dụng, đồng thời cũng tạo ra một ấn tượng tích cực đối với người dùng.

**Tính năng nâng cao:** Hiện tại, ứng dụng chưa hỗ trợ nhiều tính năng nâng cao, ví dụ như thêm hình động, hiệu ứng âm thanh,... Việc bổ sung các tính năng này sẽ giúp người dùng tạo ra các video quảng cáo phong phú và đa dạng hơn.



Em hi vọng rằng, với sự cải tiến và phát triển không ngừng, ứng dụng sẽ trở nên hoàn thiện hơn và đáp ứng tốt hơn nhu cầu của người dùng trong tương lai.

### **5.3. Phương hướng phát triển**

Dựa trên những kết quả đã đạt được và những hạn chế hiện tại, em có đã xác định một số hướng phát triển tiềm năng cho ứng dụng tạo video quảng cáo sản phẩm trong tương lai. Điều này không chỉ giúp cải thiện chất lượng và hiệu suất của ứng dụng, mà còn mở rộng khả năng tiếp cận và tương tác với người dùng. Cụ thể, em sẽ tập trung vào các phương hướng sau:

- Tối ưu hóa hiệu suất của ứng dụng bằng cách sử dụng các kỹ thuật xử lý dữ liệu hiệu quả hơn.
- Cải thiện giao diện của ứng dụng bằng cách thêm các tính năng mới, tối ưu hóa bố cục,...
- Thêm các tính năng nâng cao cho ứng dụng, ví dụ như thêm hình động, hiệu ứng âm thanh,...
- Thêm tính năng tạo video quảng cáo theo phong cách khác nhau, ví dụ như video hoạt hình, video stop-motion,...
- Thêm tính năng tích hợp với các mạng xã hội, giúp người dùng dễ dàng chia sẻ video quảng cáo lên các mạng xã hội.
- Thêm tính năng phân tích hiệu quả của video quảng cáo, giúp người dùng đánh giá hiệu quả của các chiến dịch quảng cáo.

Việc phát triển ứng dụng tạo video quảng cáo sản phẩm có thể mang lại nhiều lợi ích cho người dùng, cũng như tính thuận tiện và dễ sử dụng.

## DANH MỤC TÀI LIỆU THAM KHẢO

- [1] “OpenAI Platform,” OpenAI. [Online]. Available: <https://platform.openai.com>. [Accessed: 08-Jan-2024].
- [2] “Introduction - OpenAI API Introduction,” OpenAI. [Online]. Available: <https://openai.com/research/>. [Accessed: 08-Jan-2024].
- [3] J. Doe, “How to Write API Documentation: Best Practices and Examples,” API Documentation, 01-Jan-2024. [Online]. Available: <https://apidocumentation.com/best-practices/>. [Accessed: 08-Jan-2024].
- [4] “How to write an API reference,” The MDN Web Docs project. [Online]. Available: <https://bom.so/kE1c3C>. [Accessed: 08-Jan-2024].
- [5] “AI FAQ About D-ID’s AI Video & Avatar Generator,” D-ID. [Online]. Available: <https://d-id.com/faq/>. [Accessed: 08-Jan-2024].
- [6] “Artificial Intelligence For Dummies (2nd Edition)” by Luca Massaron, John Mueller. [Accessed: 08-Jan-2024].
- [7] “Artificial Intelligence Basics: A Non-Technical Introduction” by Tom Taulli. [Accessed: 08-Jan-2024].
- [8] “Python: Beginner’s Guide to Artificial Intelligence” by Denis Rothman, Matthew Lamons, Rahul Kumar. [Accessed: 08-Jan-2024].

## PHỤ LỤC

### - Bảng thống kê thời gian phản hồi của mô hình ChatGPT 3.5:

Tên sản phẩm	Độ dài văn bản	Thời gian phản hồi	Token
Sữa tắm	500 từ	75 giây	1000
Nước hoa	500 từ	64 giây	1000
Quần jean	500 từ	79 giây	1000
Máy tính xách tay	500 từ	85 giây	1000
Mỹ phẩm	500 từ	91 giây	1000
Nước uống đóng chai	500 từ	97 giây	1000
Điện thoại thông minh	500 từ	85 giây	1000

### - Bảng thống kê thời gian phản hồi của mô hình ChatGPT 4.0:

Tên sản phẩm	Độ dài văn bản	Thời gian phản hồi	Token
Sữa tắm	500 từ	55 giây	1000
Nước hoa	500 từ	49 giây	1000
Quần jean	500 từ	59 giây	1000
Máy tính xách tay	500 từ	65 giây	1000
Mỹ phẩm	500 từ	76 giây	1000
Nước uống đóng chai	500 từ	57 giây	1000
Điện thoại thông minh	500 từ	67 giây	1000

### - Bảng thống kê thời gian tạo video của mô hình D-ID:

Tên sản phẩm	Độ dài văn bản	Thời gian tạo	Credit
Sữa tắm	500 từ	60 giây	4
Nước hoa	500 từ	65 giây	4.3
Quần jean	500 từ	70 giây	4.6
Máy tính xách tay	500 từ	75 giây	5
Mỹ phẩm	500 từ	80 giây	5.3
Nước uống đóng chai	500 từ	85 giây	5.6
Điện thoại thông minh	500 từ	90 giây	6