**BỘ CÔNG THƯƠNG**

**TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP TP. HCM**

**KHOA CÔNG NGHỆ THÔNG TIN**



**BÁO CÁO**

**Môn: ỨNG DỤNG CÔNG NGHỆ MỚI**

**Đề tài: TÌM HIỂU THƯ VIỆN FACE API, XÂY DỰNG PHẦN MỀM NHẬN DẠNG KHUÔN MẶT NGƯỜI**

***Thành viên nhóm:***

+ Trương Trung Thiên – 14026541

+ Bùi Nguyên Thanh Tùng – 14028031

+ Phạm Tân – 14023881

***Giáo viên bộ môn:*** Tôn Long Phước – Trương Khắc Tùng

*Thành phố Hồ Chí Minh tháng 11, 2017*

**LỜI CẢM ƠN**

Chúng em xin gửi lời cám ơn chân thành tới thầy Tôn Long Phước và thầy Trương Khắc Tùng đã trực tiếp hướng dẫn, góp nhiều ý kiến chuyên môn,… để giúp chúng em hoàn thành báo cáo đồ án môn Ứng dụng công nghệ mới. Đồ án tuy không phải quá lớn nhưng với sự hướng dẫn của các thầy chúng em đã được tiếp cận với những công nghệ mới hiện nay và ứng dụng chúng trong đồ án. Đây sẽ là hành trang cho chúng em trên những chặng đường sắp tới.

Đồng thời chúng em cũng xin cám ơn các quý thầy cô trong khoa Công nghệ thông tin – Trường Đại học Công nghiệp TPHCM và bạn bè đã tận tình giúp đỡ, quan tâm, cũng như hướng dẫn thêm trong quá trình hoàn thành đồ án.

Trong quá trình làm báo cáo của chúng em rất có thể không tránh khỏi sự sai sót nên chúng em rất mong nhận được sự xem xét đóng góp ý kiến của thầy cô để khi thực hiện những đồ án sau được hoàn chỉnh hơn, chúng em xin chân thành cảm ơn.

TPHCM, ngày 5 tháng 11 năm 2017

Nhóm thực hiện

**NHẬN XÉT VÀ ĐÁNH GIÁ CỦA GIANG VIÊN HƯỚNG DẪN**……………………………………………………………………………………….  
……………………………………………………………………………………….  
……………………………………………………………………………………….  
……………………………………………………………………………………….  
……………………………………………………………………………………….  
……………………………………………………………………………………….  
……………………………………………………………………………………….  
……………………………………………………………………………………….  
……………………………………………………………………………………….  
……………………………………………………………………………………….  
……………………………………………………………………………………….  
……………………………………………………………………………………….  
……………………………………………………………………………………….  
……………………………………………………………………………………….  
……………………………………………………………………………………….  
……………………………………………………………………………………….  
……………………………………………………………………………………….  
……………………………………………………………………………………….  
……………………………………………………………………………………….  
……………………………………………………………………………………….

TP Hồ Chí Minh, ngày…tháng… năm 2017

Giảng viên hướng dẫn

**NHẬN XÉT VÀ ĐÁNH GIÁ CỦA GIANG VIÊN HƯỚNG DẪN**……………………………………………………………………………………….  
……………………………………………………………………………………….  
……………………………………………………………………………………….  
……………………………………………………………………………………….  
……………………………………………………………………………………….  
……………………………………………………………………………………….  
……………………………………………………………………………………….  
……………………………………………………………………………………….  
……………………………………………………………………………………….  
……………………………………………………………………………………….  
……………………………………………………………………………………….  
……………………………………………………………………………………….  
……………………………………………………………………………………….  
……………………………………………………………………………………….  
……………………………………………………………………………………….  
……………………………………………………………………………………….  
……………………………………………………………………………………….  
……………………………………………………………………………………….  
……………………………………………………………………………………….

TP Hồ Chí Minh, ngày…tháng… năm 2017

Giảng viên hướng dẫn

**QUÁ TRÌNH THỰC HIỆN**

|  |
| --- |
| Tên đề tài: Tìm hiểu thư viện Face API, xây dựng phần mềm nhận dạng khuôn mặt người trong kho ảnh |
| Giáo viên hướng dẫn: thầy Tôn Long Phước và thầy Trương Khắc Tùng |
| Thời gian thực hiện: 06/09/2017 đến 05/11/2017 |
| Sinh viên thực hiện:  + Bùi Nguyên Thanh Tùng – 14028031  + Trương Trung Thiên – 14026541  + Phạm Tân – 14023881 |
| Loại đề tài: Lĩnh vực Machine Learning |
| Nội dung đề tài: Tìm hiểu thư viện Face API. Đồng thời tìm hiểu cách sử dụng các bộ API để đưa vào trong ứng dụng, qua đó tạo ra một ứng dụng nhận dạng khuôn mặt người trong kho ảnh |

**\*Bảng chi tiết công việc trong nhóm:**

|  |  |  |
| --- | --- | --- |
|  | ***Công việc*** | ***Thành viên thực hiện*** |
| **Tuần 01**  (Từ: 06/09/2017  Đến: 10/09/2017) | Lập danh sách nhóm và nhận đề tài | Bùi Nguyên Thanh Tùng  Trương Trung Thiên  Phạm Tân |
| **Tuần 02**  (Từ: 11/09/2017  Đến: 17/09/2017) | Tìm hiểu đề tài, xác định mục đích của đề tài, đồng thời tìm kiếm không gian làm đề tài, tài liệu tham khảo đề tài và phân công công việc | Bùi Nguyên Thanh Tùng  Trương Trung Thiên  Phạm Tân |
| **Tuần 03**  (Từ: 18/09/2017  Đến: 24/09/2017 | Đọc tải liệu và tìm hiểu cách sử dụng bộ API và thư viện Face API, cụ thể là Face Detection | Bùi Nguyên Thanh Tùng  Trương Trung Thiên  Phạm Tân |
| **Tuần 04**  (Từ: 25/09/2017  Đến: 01/10/2017) | Tìm cách sử dụng Face API Detection và Face API Verification trên nền tảng Java. Đồng thời thiết kế giao diện demo. | Bùi Nguyên Thanh Tùng  Trương Trung Thiên  Phạm Tân |
| **Tuần 05**  (Từ: 02/10/2017  Đến: 08/10/2017) | Tìm cách sử dụng Face API Detection và Face API Verification trên nền tảng Java. | Bùi Nguyên Thanh Tùng  Trương Trung Thiên  Phạm Tân |
| **Tuần 06**  (Từ: 09/10/2017  Đến: 15/10/2017) | Đưa Face API Detection và Face API Verification vào giao diện đã thiết kế.  Tìm cách vẽ khung định vị khuôn mặt | Bùi Nguyên Thanh Tùng  Trương Trung Thiên  Phạm Tân |
| **Tuần 07**  (Từ: 16/10/2017  Đến: 22/10/2017) | Tìm cách vẽ khung định vị khuôn mặt.  Viết báo cáo | Bùi Nguyên Thanh Tùng  Trương Trung Thiên  Phạm Tân |
| **Tuần 08**  (Từ: 23/10/2017  Đến: 29/10/2017) | Chỉnh sửa giao diện, trích xuất dữ liệu từ kết quả các API đưa lên giao diện để thông báo người dùng  Tìm hiểu Face API Identification  Viết báo cáo | Bùi Nguyên Thanh Tùng  Trương Trung Thiên  Phạm Tân |
| **Tuần 09**  (Từ: 30/10/2017  Đến: 05/11/2017) | Tìm cách sử dụng Face API Identification  Test các chức năng  Hoàn thành báo cáo | Bùi Nguyên Thanh Tùng  Trương Trung Thiên  Phạm Tân |
| **Tuần 10**  (Từ: 06/11/2017  Đến: 12/11/2017) | Chỉnh sửa báo cáo  Chuẩn bị thuyết trình  Tiếp tục test chức năng | Bùi Nguyên Thanh Tùng  Trương Trung Thiên  Phạm Tân |

MỤC LỤC

[CHƯƠNG I. TỔNG QUAN 9](#_Toc498248945)

[1.1 Bối cảnh và nhu cầu thực tế 9](#_Toc498248946)

[1.2 Nội dụng đồ án 10](#_Toc498248947)

[CHƯƠNG II: GIỚI THIỆU MICROSOFT COGNITIVE SERVICES 11](#_Toc498248949)

[2.1 Sơ lược Microsoft Cognitive Service: 11](#_Toc498248950)

[2.2 Các bộ API trong Microsoft Cognitive Services 12](#_Toc498248951)

[2.2.1 Vision API: 12](#_Toc498248952)

[2.2.2 Speech API 14](#_Toc498248953)

[2.2.3 Language API: 15](#_Toc498248954)

[2.2.4 Knowledge API: 17](#_Toc498248955)

[2.2.5 Search API: 18](#_Toc498248956)

[CHƯƠNG III. GIỚI THIỆU FACE API 21](#_Toc498248957)

[3.1 REST API là gì? 21](#_Toc498248958)

[3.2 Giới thiệu về JSON 22](#_Toc498248959)

[3.2.1 Tổng quan về JSON 22](#_Toc498248960)

[3.2.2 Một số đặc điểm của JSON 23](#_Toc498248961)

[3.3 Giới thiệu Face API 23](#_Toc498248962)

[3.3.1 Face detection 24](#_Toc498248963)

[3.3.2 Face recognition 25](#_Toc498248964)

[3.3.3 So sánh Face API của Microsoft với các hãng khác: 27](#_Toc498248965)

[CHƯƠNG IV. XÂY DỰNG CHƯƠNG TRÌNH 28](#_Toc498248966)

[4.1 Yêu cầu ứng dụng: 28](#_Toc498248967)

[4.1.1. Yêu cầu chức năng 28](#_Toc498248968)

[4.2 Sơ đồ phân cấp chức năng của ứng dụng 29](#_Toc498248969)

[4.3. Đặc tả yêu cầu: 29](#_Toc498248970)

[4.3.1 Giới thiệu chung: 29](#_Toc498248971)

[4.3.2 Mô tả tổng quan ứng dụng: 30](#_Toc498248972)

[4.3.3 Đặc tả yêu cầu chức năng (Functional) 32](#_Toc498248973)

[4.4 Cấu trúc và giao diện chương trình 36](#_Toc498248974)

[4.5 Giao diện của phần mềm 41](#_Toc498248975)

[TỔNG KẾT 43](#_Toc498248976)

[TÀI LIỆU THAM KHẢO 45](#_Toc498248977)

**CHƯƠNG I. TỔNG QUAN**

* 1. **Bối cảnh và nhu cầu thực tế**

Hiện nay với cuộc Cách mạng 4.0 ngành Công nghệ thông tin trở thành một lĩnh vực được phát triển mạnh. Những mảng công nghệ tân tiến như trí tuệ nhân tạo (AI), máy học (Machine Learning), cloud computing, internet of things (IoTs) và big data.v.v.. đang được đầu tư mạnh mẽ và dần trở thành một xu thế trong những phần mềm hiện đại sau này.

Trong những mảng công nghệ tân tiến trên, mảng Máy học (Machine Learning) đang phát triển mạnh mẽ và được rất nhiều công ty lớn đầu tư như Microsoft, Kairos, Amazon, Google, Facebook,…Các công ty này tạo ra những thư viện API để hỗ trợ các lập trình viên trong việc viết ra những phần mềm phục vụ cho những nhu cầu của xã hội như nhận diện khuôn mặt hay vật thể trong các bức ảnh hay các video, tìm kiếm các trang web, hình ảnh về một thông tin nào đó hay thậm chí nhận diện cả giọng nói…

Với mong muốn tăng trải nghiệm ứng dụng của người giúp họ sử dụng chúng một cách thông minh hơn và linh hoạt hơn, các lập trình viên cần tận dụng các thư viện API đã được các công ty lớn phát triển. Trong các API có Face API thực hiện chức năng nhận diện khuôn mặt trong một bức ảnh. Hiện tại chức năng này đã được tất cả các công ty lớn làm về Machine Learning phát triển mạnh mẽ. Chức năng nhận diện khuôn mặt hiện nay không chỉ dùng cho việc chụp ảnh, nó còn có mặt trong vấn đề bảo mật hay dùng để tìm kiếm một người nào đó trong một bức ảnh được chụp bất kỳ.

Để có thể thực hiện đề tài này, cần nắm vững ngôn ngữ lập trình Java, hiểu rõ cách sử dụng các hàm API, cách làm việc với mã nguồn mở, quy trình phát triển một phần mềm,…Nhóm đã lựa chọn sử dụng thư viện Face API của Microsoft để xây dựng phần mềm với sự hướng dẫn của giáo viên bộ môn.

* 1. **Nội dụng đồ án**

Chương I. Tổng quan: có bối cảnh và nhu cầu thực tế khi thực hiện đề tài và nội dung đồ án.

Chương II. Giới thiệu Microsoft Cognitive Services

Chương III. Giới thiệu Face API

Chương IV. Xây dựng chương trình

# **CHƯƠNG II: GIỚI THIỆU MICROSOFT COGNITIVE SERVICES**

## 2.1 Sơ lược Microsoft Cognitive Service:

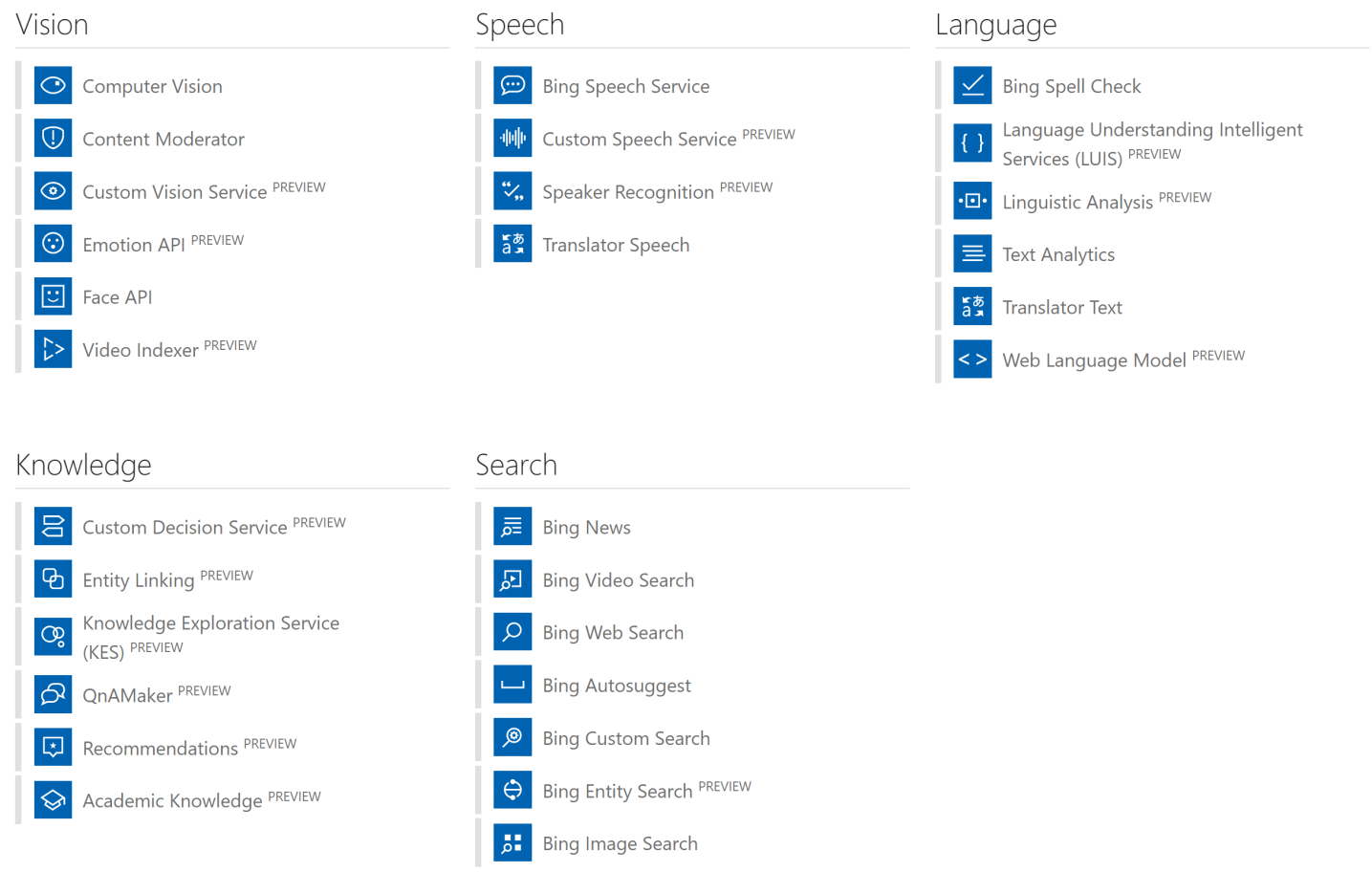
Với xu hướng tăng cường các tính năng thông minh sử dụng các thuật toán phức tạp hay tích hợp các công nghệ tân tiến như trí tuệ nhân tạo vào trong ứng dụng, giúp tăng trải nghiệm của người sử dụng bằng khả năng giải quyết các vấn đề của người dùng một cách thông minh và linh hoạt hơn. Tuy nhiên việc viết ra các thuật toán phức tạp không phải là chuyện một sớm một chiều và không phải lập trình viên nào cũng có thể làm được. Hiểu được điều đó, phía Microsoft đã đưa ra một dịch vụ để thu hẹp khoảng cách giữa các lập trình viên với việc tích hợp các công nghệ tân tiến trong ứng dụng của mình, nó được gọi là Microsoft Cognitive Services.

Microsoft Cognitve Services cung cấp các dịch vụ trí tuệ nhân tạo thông minh dưới dạng các bộ API, cho phép các lập trình viên ở mọi cấp độ đều có thể sử dụng chúng để tạo ra các ứng dụng thông minh một cách dễ dàng hơn.

Các API của Microsoft Cognitive Services được viết dưới dạng REST API do vậy lập trình viên có thể tích hợp các API này trên nhiều nền tảng khác nhau như iOS, Android, hay Windows, chỉ cần có kết nối Internet.

## 2.2 Các bộ API trong Microsoft Cognitive Services

Tính đến thời điểm viết báo cáo này, phía Microsoft đã đưa vào Cognitive Services tổng cộng 29 bộ API được chia thành 5 nhóm, bao gồm: Vision, Language, Knowledge, Speech và Search.

*Hình 1. Các bộ API trong Microsoft Cognitive Services*

### 2.2.1 Vision API:

Là nhóm API liên quan đến xử lý hình ảnh, trong nhóm Vision API này hiện Microsoft cung cấp 6 API sau:

+ Computer Vision: API này cho phép trích xuất những thông tin có giá trị từ bức ảnh của bạn với khả năng xác định được kiểu đối tượng trong ảnh (là bánh mỳ, con chó, con mèo hay cây cối, …) hay nếu là người thì API này cũng xác định được giới tính của nhân vật trong ảnh. Ngoài ra, API này cũng hỗ trợ nhận diện được những nhận vật nổi tiếng hay trích xuất chữ có trong bức hình của bạn.

+ Face API: Đây là API cho phép phát hiện khuôn mặt có trong bức hình của bạn. Ngoài ra, API này cũng trả về các thuộc tính của khuôn mặt như tuổi, giới tính, độ rạng ngời của nụ cười hay thậm chí là chiều dài của tóc, … Ngoài phát hiện khuôn mặt, Face API còn cho phép so sánh 2 khuôn mặt có phải là của cùng một người hay không.

+ Emotion API: API này cho phép xác định tâm trạng của người có trong bức hình xem họ đang vui, đang buồn hay đang giận dữ.

+ Video API: API này là một tập hợp các thuật toán xử lý video tân tiến của Microsoft. Với Video API, các nhà phát triển có thể tích hợp các tính năng chỉnh sửa video bao gồm chống rung, phát hiện khuôn mặt người, phát hiện chuyển động hay tạo video thumbnail.

+ Custom Vision Service: API dùng để tạo một bộ phân loại ảnh. Chúng ta sẽ upload ảnh lên thông qua REST API để phân loại và huấn luyện.

+ Video Indexer: API này giúp xử lý các video để trích xuất ra các thông tin để chuyển đổi giọng nói từ ngôn ngữ này thành ngôn ngữ khác, theo dõi nhận diện khuôn mặt có trong video, nhận diện văn bản trong video, tách tiếng ồn khỏi tiếng nói,…

### 2.2.2 Speech API

Là nhóm API liên quan đến xử lý âm thanh với công nghệ xử lý đến từ Bing. Trong nhóm này hiện Microsoft cung cấp 4 API:

**+** Bing Speech API: API này cho phép trích xuất một tập tin âm thanh sang dạng chữ, chuyển đổi định dạng chữ sang âm thanh (tức là đọc chữ) hay đoán ý của một câu nói.

+ Custom Recognition Intelligent Service (CRIS): CRIS cho phép chúng ta có thể tùy biến language model và acoustic model sao cho phù hợp với ứng dụng hoặc người dùng.

+ Speaker Recognition API: API này cho phép nhận dạng giọng nói của người nói trong một tập tin âm thanh. API này bao gồm 2 thành phần: speaker verification và speaker identification, tạm dịch tương ứng là xác nhận người nói và xác định người nói. **Speaker Verification** cho phép xác nhận và xác thực người dùng bằng giọng nói của họ. **Speaker Identification** có thể xác định được người đang nói trong một tập tin âm thanh dựa trên một tập dữ liệu các người nói tiềm năng. Tính năng này cũng có thể được sử dụng để xác thực người dùng bằng giọng nói. Tuy nhiên thay vì phải đọc chính xác một đoạn văn bản cố định như Speaker Verification thì khi sử dụng tính năng Speaker Identification này, người dùng có thể đọc một đoạn văn bản bất kỳ, API sẽ phân tích và đối chiếu với tập dữ liệu giọng nói của người dùng để so sánh và xác thực.

+ Translator Speech: API này giúp chuyển từ một cuộc đối thoại của một ngôn ngữ này thành một văn bản của một ngôn ngữ khác.

### 2.2.3 Language API:

Là nhóm API liên quan đến xử lý ngôn ngữ. Nhóm này hiện được Microsoft cung cấp 6 API:

+ Bing Spell Check API: API này cho phép phát hiện và sửa các lỗi chính tả có trong một đoạn văn bản mà bạn cung cấp. API còn có khả năng phát hiện từ lóng, sửa lỗi tên riêng hay sửa các từ đồng âm, …

+ Web Language Model API: API này giúp hỗ trợ xử lý ngôn ngữ tự nhiên, với khả năng chèn khoảng cách vào một đoạn văn bản được viết liền nhau như hashtag hay đường dẫn.

+ Linguistic Analysis API: API này giúp bạn hiểu sâu hơn văn bản của mình. Nó sẽ giúp phân tích cú pháp của ngôn ngữ tự nhiên để dễ dàng xác định được các thực thể (danh từ) hay các hành động (động từ) có trong văn bản. Việc xử lý văn bản này có thể hữu ích cho các công việc phân tích như phân tích tâm lý.

+ Language Understanding Intelligent Service (LUIS): LUIS cho phép lập trình viên xây dựng các model hiểu được ngôn ngữ tự nhiên cũng như hiểu được các câu lệnh riêng được thiết kế riêng cho ứng dụng của bạn.

+ Text Analytics API: API này giúp xác định các ẩn ý, từ khóa, chủ đề hay ngôn ngữ được sử dụng có trong một đoạn văn bản.

+ Translator Text: API này đã được Microsoft phát triển trước đó với Statistical Machine Translation (SMT) nhưng hiệu quả của nó chưa cao. Để cải thiện độ chính xác, phía Microssoft quyết định dùng Deep Neural Networks (DNN).

### 2.2.4 Knowledge API:

Là nhóm API liên quan đến tri thức. Nhóm Knowledge API này hiện bao gồm 6 API:

+ Academic Knowledge API: API này cho phép lập trình viên xây dựng những giải pháp tìm kiếm tài liệu học thuật với tính năng Interpret, trả về kết quả gợi ý cho từ khóa mà người dùng nhập vào dựa vào nguồn dữ liệu phong phú từ hệ thống [Microsoft Academic Graph](https://www.microsoft.com/en-us/research/project/microsoft-academic-graph/) (MAG).

+ Knowledge Exploration Service API: API này cho phép lập trình viên xây dựng những giải pháp tìm kiếm sử dụng ngôn ngữ tự nhiên bằng cách dịch ngôn ngữ tự nhiên mà người dùng nhập vào sang các biểu thức truy vấn có cấu trúc mà máy tính có thể dễ dàng hiểu và xử lý được.

+ Entity Linking Intelligence Service API: Với một đoạn văn bản, API này sẽ nhận dạng và xác định từng thực thể (entity) có trong đoạn văn dựa vào ngữ cảnh của đoạn văn đó và sẽ liên kết những entity này tới Wikipedia.

+ Recommendations: API này cho phép xây dựng các giải pháp khuyến nghị cho người dùng.

+ Custom Decision Service: API này giúp bạn tạo ra các hệ thống thông minh dựa trên điện toán đám mây để đưa quyết định với kinh nghiệm. Hệ thống kết hợp phản hồi của người dùng vào các quyết định trong thời gian thực và đáp ứng các xu hướng đang nổi lên và giải quyết các câu chuyện chỉ trong vài phút.

+ QnA Maker: Với API này chúng ta có thể tạo ra một ứng dụng trả lời câu hỏi một cách thông minh, thậm chí là tự động đặt câu hỏi tiếp theo. Việc hỏi và trả lời sẽ trở nên một cách tự động, đồng thời có ứng dụng trí tuệ nhân tạo để những câu trả lời hay câu hỏi có liên quan nhau.

### 2.2.5 Search API:

Search API cung cấp các giải pháp tìm kiếm sử dụng sức mạnh của Bing.com cũng như từ nhiều đối tác như AOL, Apple, Amazon, Yahoo, …. Nhóm Search API bao gồm 7 API cho các tác vụ tìm kiếm trang web, hình ảnh, video hay tin tức.

+ Bing Web Search API: Đây là API chủ lực của gói Search API. Chỉ với một cú pháp lệnh gọi đến API này, lập trình viên có thể lấy được các kết quả trả về cho trang web, hình ảnh, video hay tin tức tương ứng. Nó khá tương tự với việc bạn tìm kiếm trên các công cụ tìm kiếm như Bing.com hay Google.com. Ngoài ra, lập trình viên cũng sẽ nhận được những tính năng mạnh mẽ từ công cụ tìm kiếm Bing Search như ranking kết quả tìm kiếm, phân loại kết quả tìm kiếm theo vùng, …

+ Bing Autosuggest API: API này cho phép lập trình viên có thể xây dựng tính năng đề xuất các từ khóa tìm kiếm liên quan kể cả khi từ khóa tìm kiếm chưa được điền đầy đủ.

+ Bing Image Search API: API này cho phép lập trình viên có thể tìm kiếm các hình ảnh tương ứng với từ khóa nhập vào. Ngoài trả về đường dẫn của hình ảnh, API này cũng trả về các metadata hữu ích như kích thước ảnh, màu chủ đạo của ảnh, …

+ Bing Video Search API: API này cho phép lập trình viên có thể tìm kiếm các video tương ứng với từ khóa nhập vào. Ngoài ra, API này cũng trả về các metadata hữu ích khác như tên nhà sản xuất, định dạng mã hóa, ảnh thumbnail, …

+ Bing News Search API: API này cho phép lập trình viên có thể tìm kiếm các tin tức, bài báo tương ứng với từ khóa nhập vào. Ngoài ra, API cũng trả về các metadata hữu ích khác như thể loại, thông tin nhà xuất bản, ngày xuất bản, tác giả,…

+ Bing Custom Search: API này giúp người dùng chỉ tìm kiếm ở những trang web liên quan tới chủ đề người đó quan tâm. Đồng thời, chúng ta có thể loại những trang web không liên quan.

+ Bing Entity Search: API này cho phép lập trình viên tìm kiếm các thực thể (entity) . Các thực thể ở đây bao gồm nhà hàng, khách sạn, bệnh viện,… và xác định vị trí của chúng ở đâu ? Có ở gần bạn hay không ?

+ Bing Web Search: API này giúp lập trình viên tìm kiếm những trang web bao gồm hình ảnh, tin tức về từ khóa liên quan. Nó không chỉ trả về hình ảnh như Bing Image Search hay tin tức như Bing News Search.

# **CHƯƠNG III. GIỚI THIỆU FACE API**

## 3.1 REST API là gì?

Những khái niệm đầu tiên về REST (REpresentational State Transfer) được đưa ra vào năm 2000 trong luận văn tiến sĩ của Roy Thomas Fielding (đồng sáng lập giao thức HTTP). Trong luận văn ông giới thiệu khá chi tiết về các ràng buộc, quy ước cũng như cách thức thực hiện với hệ thống để có được một hệ thống REST.

Chúng ta có thể hiểu đơn giản nó là một bộ các ràng buộc và quy ước, khi áp dụng đầy đủ vào hệ thống của bạn thì ta có 1 hệ thống REST.

Hệ thống hoạt động theo mô hình client-server, trong đó server là tập hợp các service nhỏ lắng nghe các request từ client. Với từng request khác nhau thì có thể một hoặc nhiều service xử lý.

Stateless (phi trạng thái): Đơn giản server và client không lưu trạng thái của nhau nên mỗi request lên server thì client phải đóng gói thông tin đầy đủ để phía server hiểu được. Điều này giúp hệ thống của bạn dễ phát triển, bảo trì, mở rộng vì không cần tốn công CRUD trạng thái của client. Hệ thống phát triển theo hướng này có ưu điểm nhưng cũng có khuyết điểm là gia tăng lượng thông tin cần truyền tải giữa client và server.

Khả năng caching: Các response có thể lấy ra từ cache. Bằng cách cache các response, server giảm tải việc xử lý request, còn client cũng nhận được thông tin nhanh hơn. Ở đây ta đặt một cache vào giữa: client - cache - server.

Chuẩn hóa các interface: Đây là một trong những đặc tính quan trọng của hệ thống REST. Bằng cách tạo ra các quy ước chuẩn để giao tiếp giữa các thành phần trong hệ thống, bạn đã đơn giản hóa việc client có thể tương tác với server. Các quy ước này áp dụng cho toàn bộ các service giúp cho người sử dụng hệ thống của bạn dễ dùng hơn. Dễ hiểu hơn trên hệ thống bạn đặt ra một chuẩn API để người dùng dù là mobile, web đều có thể kết nối vào được. Hệ thống REST có yếu điểm ở đây vì khi chuẩn hóa rồi ta không thế tối ưu từng kết nối.

Phân lớp hệ thống: trong hệ thống REST bạn chia tách các thành phần hệ thống theo từng lớp, mỗi lớp chỉ sử dụng lớp ở dưới nó và giao tiếp với lớp ở ngay trên nó mà thôi. Điều này giúp bạn giảm độ phức tạp của hệ thống,giúp các thành phần tách biệt nhau từ đó dễ dàng mở rộng từng thành phần.

## 3.2 Giới thiệu về JSON

### 3.2.1 Tổng quan về JSON

JSON là viết tắt của Javascript Object Notation nó là một chuẩn để định dạng dữ liệu, hầu hết các ngôn ngữ lập trình hiện nay đều có thể đọc được, bạn có thể sử dụng lưu nó vào một file, hoặc một bản ghi trong cơ sở dữ liệu rất dễ dàng.

Định dạng JSON sử dụng cặp key-value cho mỗi thông tin dữ liệu, như vậy mỗi thông tin dữ liệu cũng tương đương như tên trường và giá trị của nó ở một bản ghi nào đó trong cơ sở dữ liệu.

### 3.2.2 Một số đặc điểm của JSON

Chuỗi JSON được bao bọc bởi cặp dấu ngoặc nhọn {}

Các key và value của JSON bắt buộc phải đặt trong cặp dấu nháy kép "" nếu bạn đặt nó trong dấu nháy đơn thì đây không phải là một chuỗi JSON đúng chuẩn. Nên trường hợp trong value của bạn có chứa dấu nháy kép thì hãy dùng dấu (\) để khóa dấu.

Nếu có nhiều dữ liệu thì ta dùng đấu phẩy "," để ngăn cách.

Các key của JSON bạn nên đặt tên giống như quy tắc đặt tên bao gồm: chữ cái, chữ số, dấu gạch dưới, không bao gồm khoảng trắng và các ký tự đặc biệt, ký tự đầu tiên không phải là số.

## 3.3 Giới thiệu Face API

Face API là một giải pháp cloud-base cung cấp cho người dùng hàng loạt các thuật toán nâng cao về trí tuệ nhân tạo, Face API được chia thành 2 chức năng chính bao gồm:

### 3.3.1 Face detection

Chức năng này cho phép người dùng nhập vào một ảnh từ bộ sưu tập hoặc một URL, sau đó thư viện này có khả năng nhận diên lên đến 64 khuôn mặt người trong một bức ảnh với độ chính xác cao về vị trí khuôn mặt trong bức ảnh.

Face rectangle (top, left, width, height) chính là vị trí hay tọa độ khuôn mặt trong ảnh có M x N pixel và cũng là giá trị trã về cho mỗi khuôn mặt. Ngoài ra chức năng này còn có thể trã về một số thuộc tính như: tuổi, giới tính, màu tóc, độ rạng ngời của nụ cười, cảm xúc (emotion), có mang kính hay không...

Một số lưu ý khi sử dụng Face Detection:

**+** Mỗi một request sẽ trã về một FaceId để đinh danh khuôn mặt đã request và có thời hạn trong vòng 24 giờ tức sau 24 giờ thì microsoft sẽ không lưu dữ liệu về khuôn mặt đó nữa

+ Định dạng ảnh đầu vào chấp nhận là: JPEG, PNG, GIF (khung hình đầu tiên) và BMP, khích thước ảnh không nhỏ hơn 1KB và không lớn hơn 4MB**.**

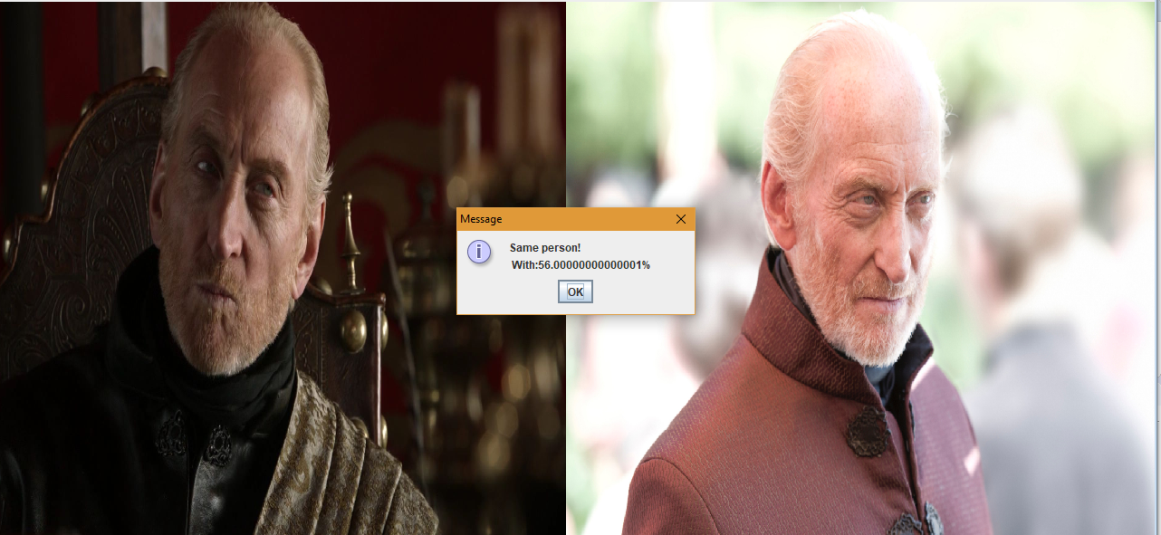
+ Mỗi khuôn mặt trong ảnh có kích thước trong khoảng 36x36 đến 4096x4096 pixel nếu vượt ngưỡng sẽ không nhận diện được**.**

+ Trong một ảnh không vượt quá 64 khuôn mặt.

### 3.3.2 Face recognition

Face recognition được ứng dụng rộng rãi trong nhiều lĩnh vực như: bảo mật, robotics, mobile app...Chia làm 4 chức năng chính: face verification, finding similar faces, face grouping, and person identification.

Face verification: Kiểm tra (chứng thực) 2 khuôn mặt có phải là cùng một người hay không.

  
*Hình 2. Ví dụ về face verificatiom*

Finding similar faces: thực hiện chức năng detect và tìm kiếm tập list face đã lưu trước đây để kiểm tra xem những ảnh nào giống với ảnh đã yêu cầu.

Face identification: Tìm kiếm khuôn mặt và xác đinh tên với dữ liệu về một số khuôn mặt đã biết trước( training). Để làm được chức năng Identify cần phải có một person group. Trong person group có person, mỗi person có nhiều khuôn mặt. Sau cùng train person group để có thể identify.

  
*Hình 3. Ví dụ về face identification*

Face grouping: Nhóm những ảnh là cùng một người vào một nhóm

  
*Hình 4. Ví dụ về face grouping*

### 3.3.3 So sánh Face API của Microsoft với các hãng khác:

*****Hình 5. Bảng so sánh Face API giữa một số hãng*

Dựa vào bảng so sánh trên có thể thấy hãng mà hỗ trợ Face API nhiều tính năng nhất là Kairos. Vậy tại sao lại chọn Microsoft để thực hiện đồ án ? Lý do để chọn Microsoft đơn giản vì nó dễ sử dụng, dễ tìm hiểu khi dùng để thực hiện những nghiên cứu đề tài.

# **CHƯƠNG IV. XÂY DỰNG CHƯƠNG TRÌNH**

## 4.1 Yêu cầu ứng dụng:

### 4.1.1. Yêu cầu chức năng

- Định vị (detect) khuôn mặt trong ảnh và dựa vào dư liệu khuôn mặt để xác định thông tin về người đó như giới tính, tuổi, màu tóc…

**-** So sánh 2 khuôn mặt có trong 2 ảnh khác nhau để xác định có phải cùng một người hay không, chức năng này gọi là Verify.

- Nhập ảnh có định dạng JPG, BMP, GIF, JPEG…

- Nhập ảnh chụp từ webcam máy tính và nhận diện được khuôn mặt.

**4.1.2 Yêu cầu phi chức năng**

- Ứng dụng có thể đọc nhiều định dạng ảnh để nhận diện khuôn mặt

- Giao diện thuận tiện, dễ sử dụng

## 4.2 Sơ đồ phân cấp chức năng của ứng dụng

## 4.3. Đặc tả yêu cầu:

### 4.3.1 Giới thiệu chung:

a) Mục đích:

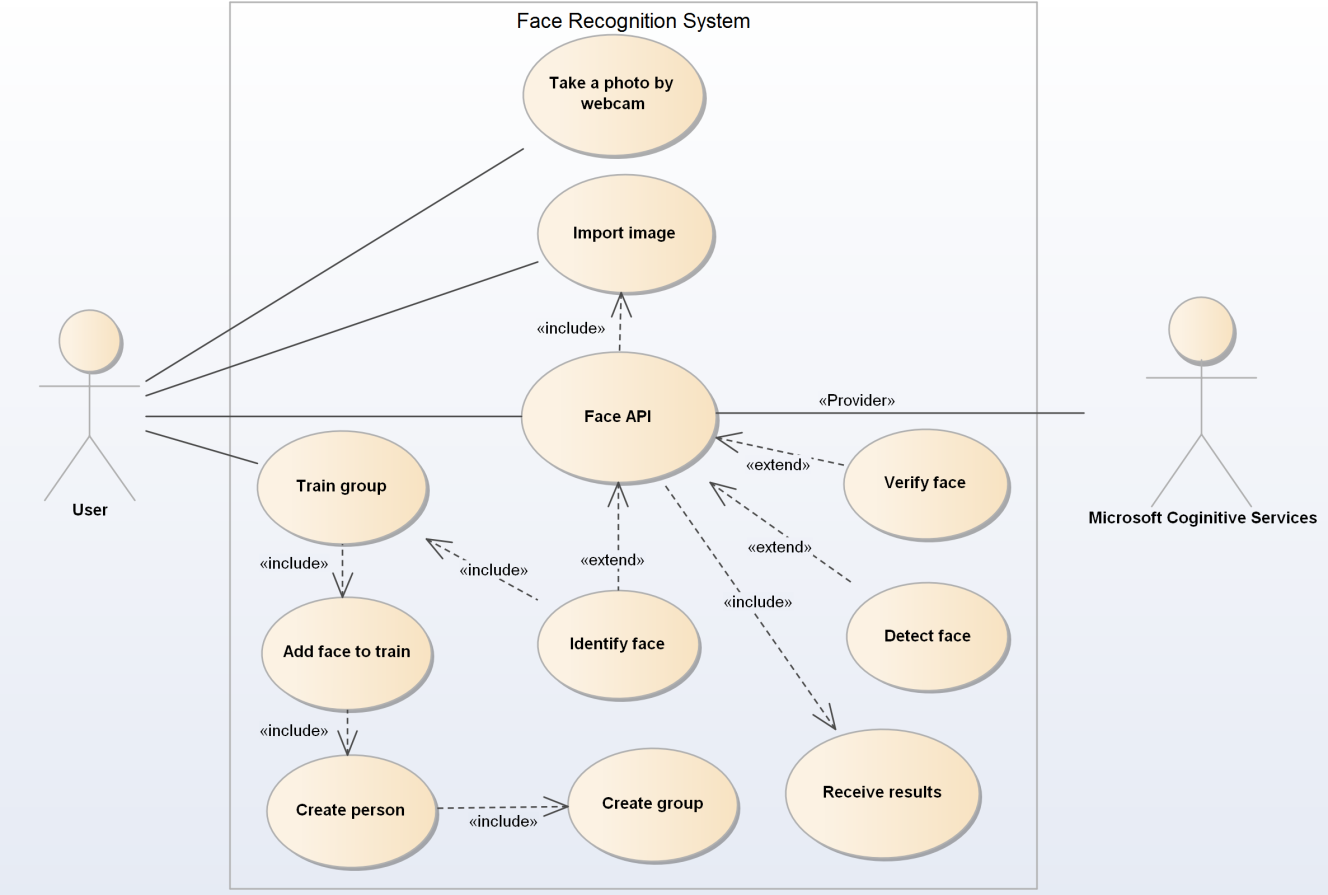
Phần mềm có chức năng chính nhận diện khuôn mặt, định vị khuôn mặt, so sánh hai khuôn mặt có cùng một người hay không và train cho ứng dụng nhận dạng một khuôn mặt.

b) Phạm vi:

Những file ảnh mà máy tính đọc được như JPG, JPEG, PNG…

### 4.3.2 Mô tả tổng quan ứng dụng:

a) Mô hình use case:

 *Hình 6. Lược đồ use case*

b) Danh sách các tác nhân và mô tả

|  |  |  |
| --- | --- | --- |
| Tác nhân | Mô tả tác nhân | Ghi chú |
| User | Đưa file ảnh vào và nhận kết quả sau khi định vị khuôn mặt hoặc so sánh hai khuôn mặt hoặc nhận dạng khuôn mặt đó là ai |  |
| Microsoft Cognitive Services | Lấy link ảnh định vị khuôn mặt hoặc so sánh hai khuôn mặt và trả về kết quả cho User |  |

c) Danh sách Use case và mô tả

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Tên Use case | Mô tả ngắn gọn Use case | Chức năng | Ghi chú |
| UC001 | Import image | User nhập hình ảnh vào ứng dụng để thực hiện những chức năng của ứng dụng như định vị khuôn mặt, so sánh hai khuôn mặt hay nhận dạng khuôn mặt | Nhập ảnh vào ứng dụng |  |
| UC002 | Face API | User sẽ sử dụng những chức năng của ứng dụng như định vị khuôn mặt, so sánh hai khuôn mặt hay nhận dạng khuôn mặt. Tất cả chức năng này đều từ hàm Face API của Microsoft Cognitve Services cung cấp. Sau đó Users sẽ nhận kết quả. | So sánh hai khuôn mặt có cùng một người hay không |  |
| UC003 | Train group |  | Train cho ứng dụng nhận dạng một gương mặt |  |
| UC004 | Take a photo by webcam | Dùng webcam trên máy tính và thực hiện chụp ảnh để dùng nhận diện khuôn mặt | Lưu ảnh ở định dạng jpg, bmp, png,… |  |

d) Các điều kiện phụ thuộc

- Face API từ Microsoft Cognitive Services

- Có mạng Internet

- Eclipse

- OpenCV

### 4.3.3 Đặc tả yêu cầu chức năng (Functional)

a) Mô tả use case UC001

|  |  |  |
| --- | --- | --- |
| **Use case: UC001\_Import image** | | |
| Mục đích: | Nhập ảnh vào ứng dụng | |
| Mô tả: | User nhập ảnh vào ứng dụng rồi thực hiện các chức năng như định vị khuôn mặt, so sánh hai khuôn mặt hay nhận dạng khuôn mặt. | |
| Tác nhân: | User | |
| Điều kiện trước: | File đưa vào ứng dụng lúc đầu phải là file ảnh. Nếu không phải file ảnh thì sẽ không đọc được và hiện thông báo. | |
| Điều kiện sau: | Ảnh được load lên giao diện của ứng dụng | |
| Luồng sự kiện chính (Basic flows) | Thứ tự các bước | Hành động |
|  | 1 | User nhập ảnh vào ứng dụng |
|  | 2 | Ảnh được load lên màn hình |
| Luồng sự kiện phụ (Alternative Flows): | 1.1 | User nhập vào không phải file ảnh, sẽ không hiện ảnh và xuất ra thông báo lỗi. |
|  | 1.2 | Nếu ở giao diện phần Detect Face thì nút nhập ảnh là nút “Detect” |
|  | 1.3 | Nếu ở giao diện phần Verification thì nút nhập ảnh là 2 nút “Browse1” và “Browse2” |
|  | 1.4 | Nếu ở giao diện phần Identification thì nút nhập là nút “Indentify Name” |

b) Mô tả use case UC002

|  |  |  |
| --- | --- | --- |
| **Use case: UC002\_Face API** | | |
| Mục đích: | Thực hiện chức năng của ứng dụng như detect face, verify face và identify face | |
| Mô tả: | User sau khi nhập ảnh vào | |
| Tác nhân: | User, Microsoft Cognitive Services | |
| Điều kiện trước: | Phải nhập ảnh trước đó vào ứng dụng rồi | |
| Điều kiện sau: | Xuất ra kết quả tùy vào mỗi chức năng | |
| Luồng sự kiện chính (Basic flows) | Thứ tự các bước | Hành động |
|  | 1 | User thực hiện một trong ba chức năng: detect face, verify face, identify face mà Microsoft Cognitve Service đã cung cấp trong Face API. |
|  | 2 | User nhận lại kết quả theo từng chức năng |
| Luồng sự kiện phụ (Alternative Flows): | 2.1 | Nếu sử dụng chức năng detect face thì kết quả nhận được là vị trí khuôn mặt kèm theo một thuộc tính liên quan đến khuôn mặt |
|  | 2.2 | Nếu sử dụng chức năng verify face thì kết quả nhận được là kết quả so sánh 2 khuôn mặt có cùng một người không và kèm theo phần trăm giống nhau giữa 2 khuôn mặt. |
|  | 2.3 | Nếu sử dụng chức năng identify face thì kết quả nhận được là nhận dạng được khuôn mặt đó là ai. |

c) Mô tả use case UC003

|  |  |  |
| --- | --- | --- |
| **Use case: UC003\_Train group** | | |
| Mục đích: | Train cho ứng dụng nhận dạng một gương mặt | |
| Mô tả: | User sẽ tạo group, gán tên mặt người đó, nhập ảnh có mặt người đó vào train cho ứng dụng nhận dạng gương mặt đó.. | |
| Tác nhân: | User | |
| Điều kiện trước: | Phải có ảnh dùng để train | |
| Điều kiện sau: | Nhập ảnh vào sẽ nhận dạng đó là ai | |
| Luồng sự kiện chính (Basic flows) | Thứ tự các bước | Hành động |
|  | 1 | User tạo group |
|  | 2 | User đặt tên face trong group vừa tạo |
|  | 3 | User nhập vào ảnh để train group |
|  | 4 | User nhấn nút “Train Group” để train cho ứng dụng nhận dạng khuôn mặt trong ảnh |

d) Mô tả use case UC004

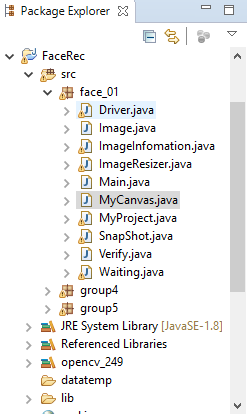
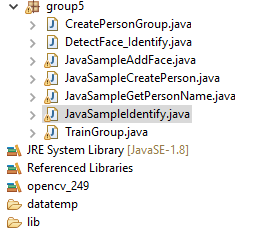
|  |  |  |
| --- | --- | --- |
| **Use case: UC004\_ Take a photo by webcam** | | |
| Mục đích: | Dùng webcam trên máy tính và thực hiện chụp ảnh để dùng nhận diện khuôn mặt | |
| Mô tả: | User sẽ sử dụng webcam trên máy tính và thực hiện chụp ảnh để dùng nhận diện khuôn mặt nào đó | |
| Tác nhân: | User | |
| Điều kiện trước: | Webcam máy tính hoạt động | |
| Điều kiện sau: | Ảnh sau khi chụp được lưu lại | |
| Luồng sự kiện chính (Basic flows) | Thứ tự các bước | Hành động |
|  | 1 | Nhấn nút “Webcam” để khởi động chức năng chụp ảnh từ webcam của máy tính |
|  | 2 | Hiện ra giao diện để dùng webcam, nhấn nút “Star” để dùng webcam |
|  | 3 | Nhấn nút “Capture” để chụp và lưu ảnh |
| Luồng sự kiện phụ (Alternative Flows): | 3.1 | Có thể nhấn nút “Stop” đề dừng màn hình webcam, xong nhấn nút Capture để chụp và lưu ảnh lại |

## 

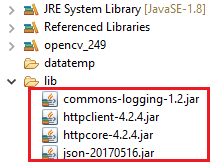
## 4.4 Cấu trúc và giao diện chương trình

Chương trình chia làm 3 chức năng chính: là Detect face, verify face và identify face.

Ngôn ngữ hiện thực: Java

   
*Hình 7 và 8. Các package và các lớp chức năng trong ứng dụng*

Để sử dụng được face API phải có 4 thư viện hỗ trợ cho việc request bao gồm: bộ giao thức hppt và bộ thư viện xử lý file JSON.



*Hình 9. Các thư viện đã thêm vào*

Để thực hiện những công việc trên chúng ta cần chuẩn bị: một subcription key và một requestURL do Microsoft cung cấp

Thực hiện:

**public** **static** **void** main(String[] args)

{

HttpClient httpclient = **new** DefaultHttpClient();

**try**

{

URIBuilder builder = **new** URIBuilder(***uriBase***);

// Request parameters. All of them are optional.

builder.setParameter("returnFaceId", "true");

// builder.setParameter("returnFaceLandmarks", "false");

// builder.setParameter("returnFaceAttributes", "age,gender,headPose,smile,facialHair,glasses,emotion,hair,makeup,occlusion,accessories,blur,exposure,noise");

// Prepare the URI for the REST API call.

URI uri = builder.build();

HttpPost request = **new** HttpPost(uri);

// Request headers.

request.setHeader("Content-Type", "application/octet-stream");

request.setHeader("Ocp-Apim-Subscription-Key", ***subscriptionKey***);

// Request body.

// StringEntity reqEntity = new StringEntity("{\"url\":\"https://upload.wikimedia.org/wikipedia/commons/c/c3/RH\_Louise\_Lillian\_Gish.jpg\"}");

FileInputStream fis=**new** FileInputStream("C:\\data.jpg");

InputStreamEntity reqEntity=**new** InputStreamEntity(fis,-1);

request.setEntity(reqEntity);

// Execute the REST API call and get the response entity.

HttpResponse response = httpclient.execute(request);

HttpEntity entity = response.getEntity();

**if** (entity != **null**)

{

// Format and display the JSON response.

System.***out***.println("REST Response:\n");

String jsonString = EntityUtils.*toString*(entity).trim();

**if** (jsonString.charAt(0) == '[') {

JSONArray jsonArray = **new** JSONArray(jsonString);

System.***out***.println(jsonArray.toString());

}

**else** **if** (jsonString.charAt(0) == '{') {

JSONObject jsonObject = **new** JSONObject(jsonString);

System.***out***.println(jsonObject.toString());

} **else** {

System.out.println(jsonString);

}

}

}

**catch** (Exception e)

{

System.***out***.println(e.getMessage());

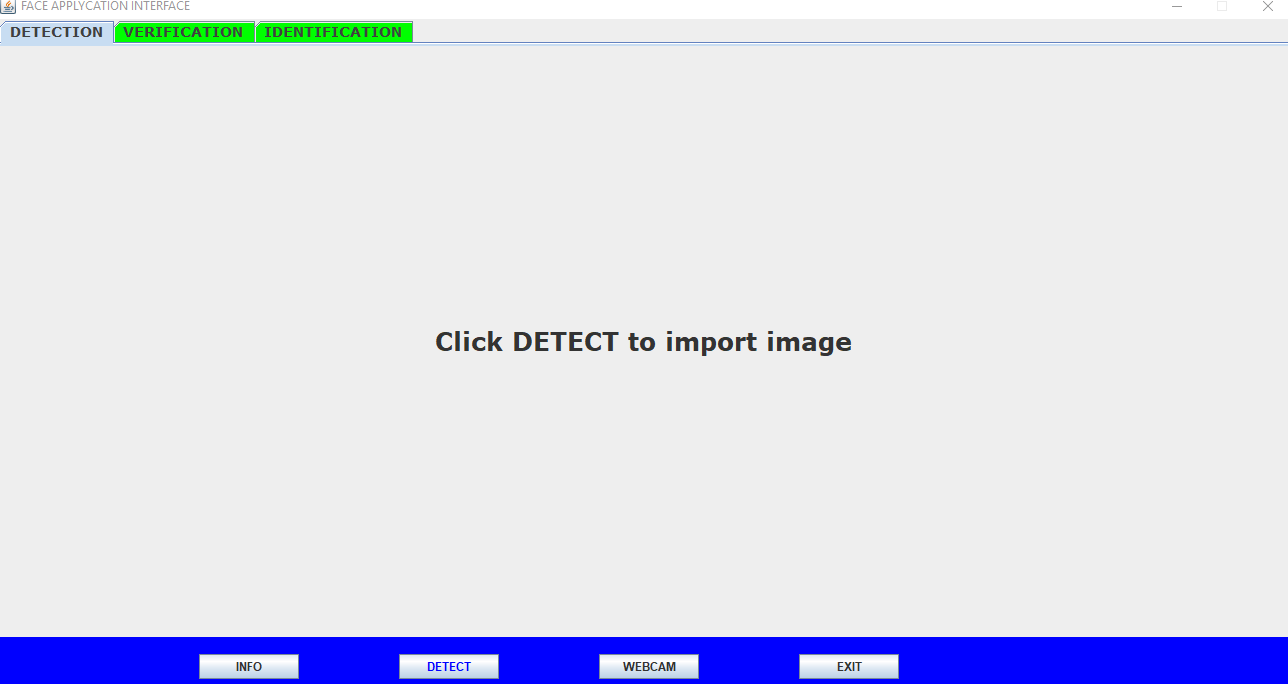
//System.out.println("Err");

}

}

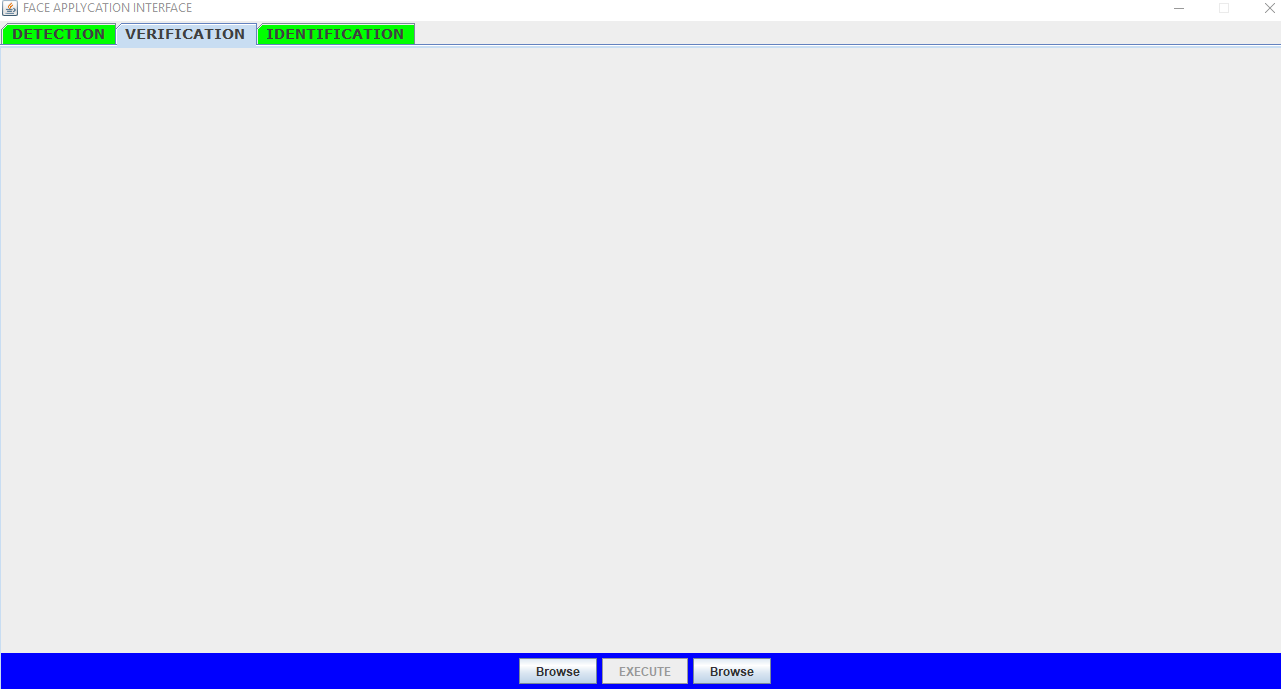
## 4.5 Giao diện của phần mềm

**Giao diện chính** (Giao diện của chức năng detect face)



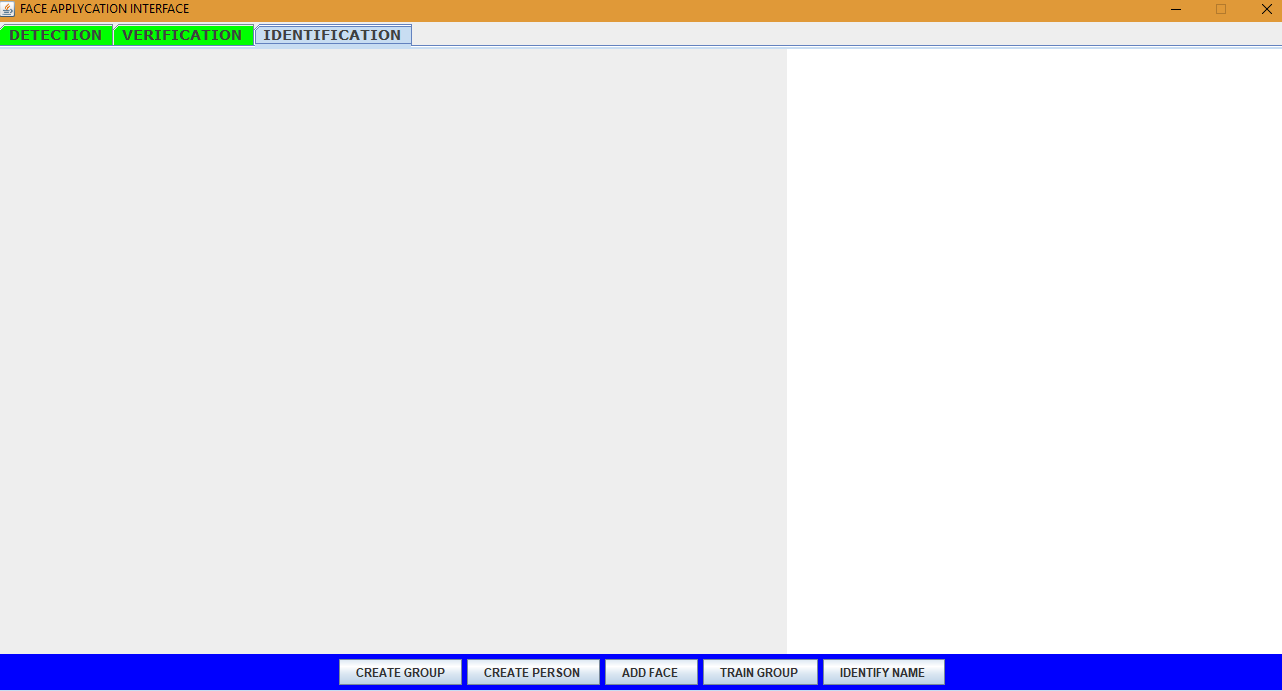
*Hình 10. Giao diện chính của phần mềm (Giao diện phần detect face)*

**Giao diện của chức năng verify face**



*Hình 11. Giao diện chức năng verify face*

**Giao diện chức năng identify face:**

*****Hình 12. Giao diện chức năng identify face*

# **TỔNG KẾT**

**MỘT SỐ KẾT QUẢ ĐẠT ĐƯỢC:**

* Học được cách thức viết một chương trình bằng ngôn ngữ lập trình Java.
* Hiểu thêm cách sử dùng bộ API.
* Biết cách làm việc với mã nguồn mở và tiếp cận công nghệ mới.
* Nâng cao kỹ năng lập trình, thuyết trình, viết báo cáo và làm việc nhóm.
* Về phần ứng dụng:
  + Chức năng định vị khuôn mặt đã hoạt động
  + Chức năng so sánh hai khuôn mặt đã hoạt động
  + Chức năng nhận dạng khuôn mặt đó là ai
  + Nhận dạng được nhiều loại ảnh khác nhau từ thư viện trong máy tính như: bitmap, jpg, png, …
  + Đối với ảnh có chất lượng tốt, độ chính xác trung bình lên tới 90% tới 95%.
  + Chụp ảnh bằng webcam máy tính rồi nhận dạng.

**HẠN CHẾ:**

* Ứng dụng không nhận dạng được các hình ảnh có kích thước quá nhỏ hoặc bị vỡ vì quá lớn.
* Giao diện ứng dụng còn thô sơ chưa được đầu tư để gây sự chú ý cho người dùng.
* Sắp xếp các chức năng còn gây chút bất tiện cho người dùng.
* Ứng dụng còn phải phụ thuộc vào máy tính.

**HƯỚNG PHÁT TRIỂN:**

* Thiết kế giao diện tốt hơn.
* Sắp xếp chức năng giúp người dùng cảm thấy thuận tiện.
* Cố gắng chạy được chương trình trên di động.

- Tích hợp vào phần cứng để tạo ra một số thiết bị thông minh.

- Triển khai ứng dụng trên nền tảng di động: Android, iOS…

# **TÀI LIỆU THAM KHẢO**

*Tài liệu web:*

<http://blog.lionpham.com/2016/08/27/microsoft-cognitive-services-api/>

<https://techtalk.vn/tim-hieu-ve-microsoft-cognitive-services-suc-manh-cua-machine-learning.html>

<https://toidicodedao.com/2016/07/25/microsoft-cognitive-service-api/>

<https://medium.com/@ageitgey/machine-learning-is-fun-part-4-modern-face-recognition-with-deep-learning-c3cffc121d78>