**TRƯỜNG CAO ĐẲNG NGHỀ ĐÀ NẴNG**

**KHOA CÔNG NGHỆ THÔNG TIN**

**---o0o---**

Logo, company name

Description automatically generated

**BÁO CÁO ĐỒ ÁN TỐT NGHIỆP**

**ĐỀ TÀI:**

**XÂY DỰNG ỨNG DỤNG HỖ TRỢ NHẬN DẠNG HỌC SINH, SINH VIÊN TRƯỜNG CAO ĐẲNG NGHỀ ĐÀ NẴNG**

**Giảng viên hướng dẫn : Hồ Viết Hà**

**Ngành : Công nghệ thông tin**

**Lớp : 20CNTT1A**

**Sinh viên thực hiện**

**Ngô Duy Hưng - Võ Thanh Nhật - Lê Lợi**

ĐÀ NẴNG – 2023

**LỜI CẢM ƠN**

Để hoàn thành tốt đồ án này, lời đầu tiên em xin cảm ơn Ban Giám Hiệu trường Cao Đẳng Nghề Đà Nẵng đã tạo điều kiện cho em được học tập tại trường, và đặc biệt em xin gửi lời cảm ơn chân thành đến thầy Hồ Viết Hà - hiệu trưởng trường Cao Đẳng Nghề Đà Nẵng đã trang bị cho em những kiến thức chuyên môn, quan tâm và tận tình hướng dẫn, giúp đỡ em hoàn thành một cách tốt nhất trong đợt làm đồ án này.

Bên cạnh đó để hoàn thành tốt đồ án này em cũng nhận được nhiều sự giúp đỡ những lời động viên quý báu của bạn bè, các anh chị em xin chân thành cảm ơn.

Tuy nhiên do thời gian hạn hẹp, cũng như lượng kiến thức còn hạn chế nên bài báo cáo của em sẽ khó tránh khỏi những thiếu sót. Em rất mong nhận được sự thông cảm và sự chỉ bảo tận tình của quý Thầy Cô và các bạn để thực hiện tốt đồ án này.

Một lần nữa em xin chân thành cảm ơn.

*Đà Nẵng, ngày…tháng…năm 2022*

Sinh viên thực hiện

Ngô Duy Hưng  
 Lê Lợi

Võ Thanh Nhật

# LỜI CAM ĐOAN

Chúng tôi xin cam đoan bài đồ án tốt nghiệp là do chính tôi thực hiện, số liệu và kết quả nghiên cứu trong báo cáo này là trung thực. Các thông tin trích dẫn trong đồ án đã được chỉ rõ nguồn gốc rõ ràng và được phép công bố.

Xin chân thành cảm ơn !

Đà Nẵng, ngày…tháng…năm 2022

Sinh viên thực hiện

Ngô Duy Hưng  
 Lê Lợi

Võ Thanh Nhật

**MỤC LỤC**

[LỜI CẢM ƠN 1](#_Toc14521)

[LỜI CAM ĐOAN 2](#_Toc9388)

[DANH MỤC HÌNH ẢNH 4](#_Toc23250)

[DANH MỤC CÁC TỪ VIẾT TẮT 6](#_Toc6993)

[CHƯƠNG 1: MỞ ĐẦU 8](#_Toc32303)

[1. Tên đề tài 8](#_Toc16628)

[2. Lí do chọn đề tài 8](#_Toc27962)

[3. Mục tiêu đề tài 8](#_Toc9877)

[CHƯƠNG 2: CƠ SỞ LÝ THUYẾT 9](#_Toc11309)

[1. Giới thiệu công nghệ nhận dạng khuôn mặt: 9](#_Toc477)

[2. Các framework được sử dụng: 10](#_Toc6935)

[3. Ngôn ngữ và thư viện được sử dụng: 14](#_Toc22398)

[CHƯƠNG 3: KHẢO SÁT VÀ PHÂN TÍCH THIẾT KẾ HỆ THỐNG 18](#_Toc17722)

[1. Khảo sát thực trạng và yêu cầu bài toán 18](#_Toc8611)

[2. Phân tích hệ thống: 18](#_Toc12351)

[CHƯƠNG 4: XÂY DỰNG ỨNG DỤNG 37](#_Toc6868)

[1. Chuẩn bị: 37](#_Toc21003)

[2. Giao diện: 37](#_Toc22772)

[CHƯƠNG 5: KẾT LUẬN 41](#_Toc19005)

[1. Kết quả đạt được: 41](#_Toc329)

[2. Hướng phát triển: 41](#_Toc9372)

**DANH MỤC CÁC TỪ VIẾT TẮT**

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Từ viết tắt** | **Tên đầy đủ** | **Giải thích** |
| 1 | HTML | HyperText Markup Language | Ngôn ngữ đánh dấu siêu văn bản |
| 2 | CSS | Cascading Style Sheets | Miêu tả trình bày tài liệu viết bằng ngôn ngữ HTML |
| 3 | DFD | Data Flow Diagram | Chỉ ra cách thông tin chuyển vận từ một tiến trình hoặc từ chức năng này trong hệ thống |
| 4 | XHTML | Extensible HyperText Markup Language | Ngôn ngữ Đánh dấu Siêu văn bản Mở rộng |
| 5 | [W3C](http://www.w3c.org/) | World Wide Web Consortium | tổ chức tiêu chuẩn quốc tế chính cho World Wide Web |
| 6 | SVG | Scalable Vector Graphics | Ngôn ngữ đánh dấu (Markup Language) XML. Dùng để mô tả các hình ảnh đồ họa vector 2 chiều, tĩnh và hoạt hình |
| 7 | SETL | Set Language | [Ngôn ngữ lập trình được phát triển tại NYU (Viện Toán Courant, Hoa Kỳ)](https://vi.wikipedia.org/wiki/Hoa_K%E1%BB%B3) |
| 8 | CNN | Convolutional Neural Network | Một mạng tích chập. là một lớp của mạng thần kinh sâu (deep neural network), Áp dụng phổ biến nhất để phân tích hình ảnh trực quan |

# CHƯƠNG 1: MỞ ĐẦU

1. **Tên đề tài**

Xây dựng ứng dụng hỗ trợ nhận dạng học sinh sinh viên trường Cao Đẳng Nghề Đà Nẵng

1. **Lí do chọn đề tài**

Theo nhu cầu phòng chống trộm và bảo vệ an toàn cho sinh viên và học sinh. Hiện nay trên thế giới, trong các môi trường cộng đồng, đang áp dụng các mô hình nhận diện các biển số xe, nhân viên, chống trộm. Các công ty lớn như FPT,…, các bãi đỗ xe, các công trình nhà máy, xí nghiệp, các nút giao thông công cộng.

Bởi lý do đó, chúng tôi đã lên ý tưởng để xây dựng một ứng dụng hỗ trợ nhận diện học sinh sinh viên trường Cao Đẳng Nghề Đà Nẵng. Để đề bảo vệ an toàn cho học sinh, sinh viên khi ra vào nhà trường.

1. **Mục tiêu đề tài**

Với mong muốn giúp nhà trường tăng cường bảo an trong và ngoài trường. Chúng tôi đã đưa ra những mục tiêu như sau:

- Nhận dạng khuôn mặt học sinh, sinh viên trong danh sách sinh viên của trường.

- Nhận dạng được người lạ ra vào trường.

- Lưu thông tin nhận dạng như giờ, ngày, tháng, năm

- Thực hiện thao tác xóa sinh viên trên hệ thống nhận dạng

- Thực hiện thao tác xóa danh sách nhận dạng trong danh sách nhận dạng.

# CHƯƠNG 2: CƠ SỞ LÝ THUYẾT

## Giới thiệu công nghệ nhận dạng khuôn mặt:

1. **OpenCV:** Chúng tôi sử dụng OpenCV để xử lý ảnh. OpenCV viết tắt cho Open Source Computer Vision Library. OpenCV là thư viện nguồn mở hàng đầu cho Computer Vision và Machine Learning, và hiện có thêm tính năng tăng tốc GPU cho các hoạt động theo real-time.
2. **Mediapipe:** Về tổng quan, [MediaPipe](https://github.com/google/mediapipe) là tập hợp của một loạt các giải pháp Machine Learning đa nền tảng, có thể can thiệp được và cực kỳ lightweight. Với đầu vào là một ảnh hoặc một video, nhiệm vụ của chúng ta là tìm ra vị trí và đóng hộp (bounding box) những khuôn mặt con người xuất hiện trên đấy, cũng như đánh dấu các điểm quan trọng (*MediaPipe* sử dụng 5-landmarks) trên khuôn mặt đó. *MediaPipe Face Detection* sử dụng mạng *BlazeFace* làm nền tảng nhưng thay đổi backbones. Ngoài ra, thuật toán *NMS (non-maximum suppression)* cũng được thay thế bởi một chiến thuật khác, giúp thời gian xử lý giảm đáng kể.
3. **Face recognition:** Face recognition nhận dạng và thao tác khuôn mặt từ Python hoặc từ dòng lệnh với thư viện nhận dạng khuôn mặt đơn giản nhất thế giới. Được xây dựng bằng cách sử dụng tính năng nhận dạng khuôn mặt tiên tiến nhất của [dlib](http://dlib.net/) được xây dựng bằng học sâu. Mô hình có độ chính xác 99,38% trên các [Khuôn mặt được dán nhãn trong điểm chuẩn Wild](http://vis-www.cs.umass.edu/lfw/) .  
   Face recognition sẽ trích xuất đặc trưng khuôn mặt và đưa ra mã hóa dữ liệu khuôn mặt dạng CNN
4. **CNN:**

*CNN* là một trong những mô hình Deep Learning tiên tiến. Nó giúp cho chúng ta xây dựng được những hệ thống thông minh với độ chính xác cao như hiện nay. Mạng CNN là một tập hợp các lớp Convolution chồng lên nhau và sử dụng các hàm nonlinear activation như ReLU và tanh để kích hoạt các trọng số trong các node. Mỗi một lớp sau khi thông qua các hàm kích hoạt sẽ tạo ra các thông tin trừu tượng hơn cho các lớp tiếp theo.

Trong quá trình huấn luyện mạng (traning) CNN tự động học các giá trị qua các lớp filter dựa vào cách thức mà bạn thực hiện. Ví dụ trong tác vụ phân lớp ảnh, CNNs sẽ cố gắng tìm ra thông số tối ưu cho các filter tương ứng theo thứ tự raw pixel > edges > shapes > facial > high-level features. Layer cuối cùng được dùng để phân lớp ảnh.

1. **Các framework được sử dụng:**

Chúng tôi sử dụng Django để làm giao diện và các chức năng quản trị cho ứng dụng hỗ trợ nhận dạng khuôn mặt.

1. **Khái niệm Django:**

**Django**là một **Framework lập trình Web** bậc cao, mã nguồn mở được viết bằng **Ngôn ngữ lập trình Python**.

1. **Mô hình Django:**

Django sử dụng mô hình MVT (Model- View- Template) thay vì sử dụng mô hình MVC (Model- View- Controller).

Mô hình MVT được sử dụng trong khi tạo một khối ứng dụng tương tác với người dùng. Mô hình này bao gồm code HTML với Django Template Language (DTL). Bất cứ khi nào người dùng người request, nó xử lý request của người dùng đó bằng Model, View và Template. Nó hoạt động như một Controller để kiểm tra xem nó có khả dụng hay không bằng cách ánh xạ URL và nếu URL ánh xạ thành công thì View sẽ bắt đầu tương tác với Model và gửi lại Template cho người dùng dưới dạng respone.

1. **Các thành phần cơ bản của Django:**

Dự án Django tạo ra một tập hợp các cài đặt bao gồm cấu hình cơ sở dữ liệu và các tùy chọn cụ thể cũng như các cài đặt cụ thể của ứng dụng mặc định trong dự án.

Các thành phần của Django gồm có:

* Manage.py: Cho phép bạn tương tác với dự án Django theo các cách khác nhau
* \_\_init.py\_\_: Nói với trình thông dịch python là thư mục nên được coi là một python package. Tập tin này chủ yếu là trống.
* Setting.py: Tập tin cấu hình
* Urls.py: Bao gồm tất cả khai báo URL cho dự án Django và mục lục của trang web Django.
* Wsgi.py: Đây là lối vào cho các máy chủ web tương thích WSGI để phục vụ các dự án của bạn và deploy với WSGI.

Nếu bạn muốn kiểm tra, dự án của bạn có đang chạy không, bạn có thể kiểm tra nó bằng lệnh sau:

Python manage.py runserver

1. **Cài đặt Django:**

* Sử dụng pip để cài đặt Django. Mở cửa sổ cmd, từ cửa sổ cmd gõ lệnh:

pip install django

**Note** : Tải Python trước nhá , không có python thì không có Django mà xài.

* Kiểm tra Django đã được cài đặt .Từ cửa sổ cmd, chạy python, sau đó kiểm tra phiên bản Django đã cài đặt với các lệnh:

>> import django

>> print(django.get\_version())

**Tạo mới project với Django**

* Từ cửa sổ cmd, tạo mới project bằng lệnh:

django-admin startproject mysite

* Project mới sẽ được tạo ra trong thư mục mysite với các file như sau:

mysite/

manage.py

mysite/

\_\_init\_\_.py

settings.py

urls.py

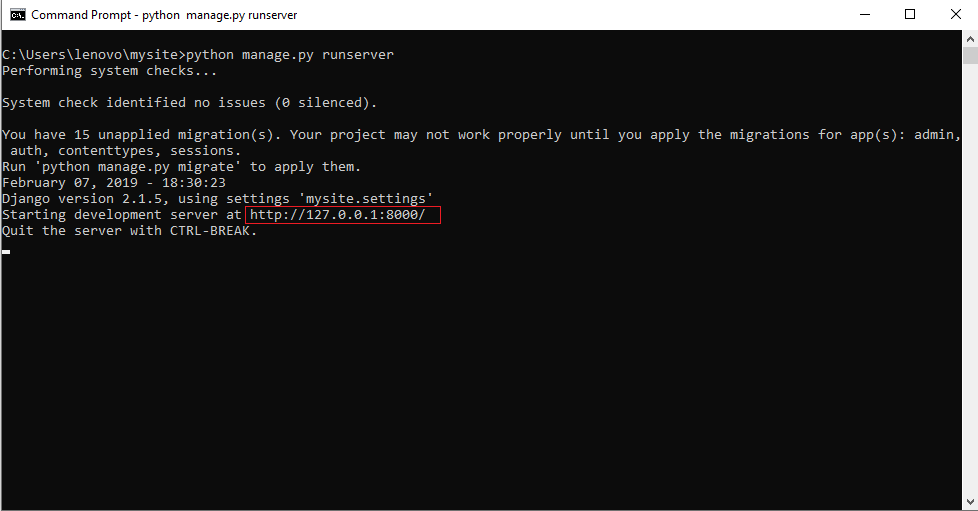
wsgi.py

* Chạy thử sever để coi ok chưa nhá !
* Từ cửa sổ cmd, di chuyển vào trong thư mục project (mysite) và khởi động server với lệnh:

cd mysite

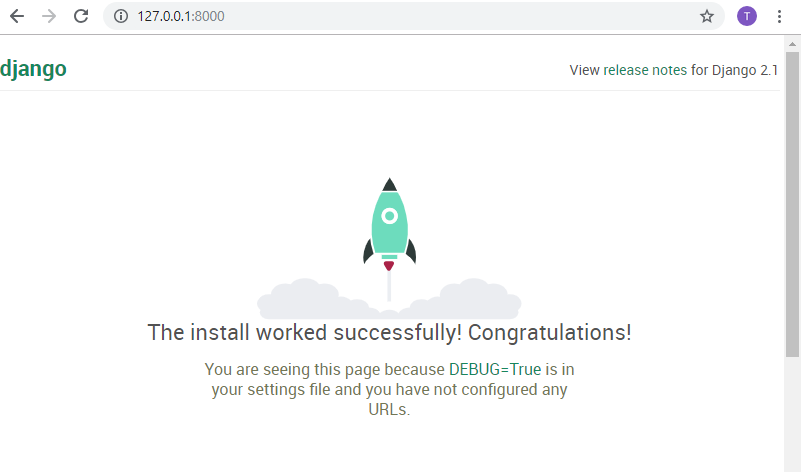
python manage.py runserver

* Kết quả:



Hình ảnh . Kết quả cài đặt DjangoKết quả cài đặt Django(1)

Theo mặc định, server sẽ được chạy tại địa chỉ [http://127.0.0.1:8000](https://www.google.com/url?q=http://127.0.0.1:8000&sa=D&ust=1557762779922000)



Hình ảnh . Kết quả cài đặt DjangoKết quả cài đặt Django(2)

Để thay đổi địa chỉ ip và cổng mà server sẽ chạy, có thể dùng lệnh:

python manage.py runserver 80

Nó đổi port server sang port 8080

Vài ví dụ về port server:

python manage.py runserver 8080 # nghe tại cổng 8080

python manage.py runserver 0.0.0.0:8080 # nghe tại cổng 8080, tất cả ip

python manage.py runserver 0:8080 # nghe tại cổng 8080, tất cả ip

1. **Ngôn ngữ và thư viện được sử dụng:**

HTML, CSS, Python, Bootstrap

1. **Giới thiệu****:**

*- HTML:* HTML (viết tắt của từ *HyperText Markup Language*, hay là "Ngôn ngữ Đánh dấu Siêu văn bản") là một [ngôn ngữ đánh dấu](https://vi.wikipedia.org/wiki/Ng%C3%B4n_ng%E1%BB%AF_%C4%91%C3%A1nh_d%E1%BA%A5u) được thiết kế ra để tạo nên các [trang web](https://vi.wikipedia.org/wiki/Website) trên [*World Wide Web*](https://vi.wikipedia.org/wiki/World_Wide_Web). Nó có thể được trợ giúp bởi các công nghệ như [*CSS*](https://vi.wikipedia.org/wiki/CSS) và các ngôn ngữ kịch bản giống như [*JavaScript*](https://vi.wikipedia.org/wiki/JavaScript).

*- CSS:* Trong [tin học](https://vi.wikipedia.org/wiki/Tin_h%E1%BB%8Dc), các tập tin định kiểu theo tầng – dịch từ [tiếng Anh](https://vi.wikipedia.org/wiki/Ti%E1%BA%BFng_Anh) là *Cascading Style Sheets* (CSS) – được dùng để miêu tả cách trình bày các tài liệu viết bằng ngôn ngữ [*HTML*](https://vi.wikipedia.org/wiki/HTML)và [*XHTML*](https://vi.wikipedia.org/wiki/XHTML). Ngoài ra ngôn ngữ định kiểu theo tầng cũng có thể dùng cho [*XML*](https://vi.wikipedia.org/wiki/XML), [*SVG*](https://vi.wikipedia.org/wiki/SVG), [*XUL*](https://vi.wikipedia.org/w/index.php?title=XUL&action=edit&redlink=1). Các đặc điểm kỹ thuật của CSS được duy trì bởi [World Wide Web Consortium](https://vi.wikipedia.org/wiki/W3C) ([W3C](http://www.w3c.org/)). Thay vì đặt các thẻ quy định kiểu dáng cho văn bản [HTML](https://vi.wikipedia.org/wiki/HTML) (hoặc XHTML) ngay trong [nội dung](https://vi.wikipedia.org/w/index.php?title=N%E1%BB%99i_dung&action=edit&redlink=1) của nó, bạn nên sử dụng CSS.

*- Bootstrap:* Bootstrap là một [*framework*](https://vi.wikipedia.org/w/index.php?title=Framework&action=edit&redlink=1)[CSS](https://vi.wikipedia.org/wiki/CSS) [miễn phí sử dụng mã nguồn mở](https://vi.wikipedia.org/wiki/Ph%E1%BA%A7n_m%E1%BB%81m_t%E1%BB%B1_do_ngu%E1%BB%93n_m%E1%BB%9F) dùng cho mục đích phát triển web front-end cho [thiết bị di động](https://vi.wikipedia.org/wiki/Thi%E1%BA%BFt_b%E1%BB%8B_di_%C4%91%E1%BB%99ng). Bên cạnh những mẫu thiết kế [kiểu chữ](https://vi.wikipedia.org/wiki/Thi%E1%BA%BFt_k%E1%BA%BF_web), biểu mẫu, nút hay thanh điều hướng bằng ngôn ngữ [*HTML*](https://vi.wikipedia.org/wiki/HTML), *CSS* và [*JavaScript*](https://vi.wikipedia.org/wiki/JavaScript), framework này còn bao gồm một số thành phần giao diện khác.

- *Python:* Python là một [ngôn ngữ lập trình](https://vi.wikipedia.org/wiki/Ng%C3%B4n_ng%E1%BB%AF_l%E1%BA%ADp_tr%C3%ACnh) bậc cao cho các mục đích lập trình đa năng, do [*Guido van Rossum*](https://vi.wikipedia.org/w/index.php?title=Guido_van_Rossum&action=edit&redlink=1) tạo ra và lần đầu ra mắt vào năm [1991](https://vi.wikipedia.org/wiki/1990). Python được thiết kế với ưu điểm mạnh là dễ đọc, dễ học và dễ nhớ. Python là ngôn ngữ có hình thức rất sáng sủa, cấu trúc rõ ràng, thuận tiện cho người mới học lập trình và là ngôn ngữ lập trình dễ học; được dùng rộng rãi trong phát triển [trí tuệ nhân tạo](https://vi.wikipedia.org/wiki/Tr%C3%AD_tu%E1%BB%87_nh%C3%A2n_t%E1%BA%A1o). Cấu trúc của *Python* còn cho phép người sử dụng viết mã lệnh với số lần gõ phím tối thiểu. Vào tháng 7 năm 2018, *van Rossum* đã từ chức lãnh đạo trong cộng đồng ngôn ngữ *Python* sau 30 năm làm việc.

1. Lịch sử ra đời:

- *HTML:* Năm 1980, nhà vật lý [*Tim* Berners-Lee](https://vi.wikipedia.org/wiki/Tim_Berners-Lee), một nhà thầu tại [*CERN*](https://vi.wikipedia.org/wiki/CERN), đã đề xuất và tạo mẫu [*ENQUIRE*](https://vi.wikipedia.org/w/index.php?title=ENQUIRE&action=edit&redlink=1), một hệ thống cho các nhà nghiên cứu *CERN* sử dụng và chia sẻ tài liệu. Năm 1989, *Berners-Lee* đã viết một bản ghi nhớ đề xuất một hệ thống [siêu văn bản](https://vi.wikipedia.org/wiki/Si%C3%AAu_v%C4%83n_b%E1%BA%A3n) dựa trên [Internet](https://vi.wikipedia.org/wiki/Internet). *Berners-Lee* xác định rõ *HTML* và viết phần mềm trình duyệt và máy chủ vào cuối năm 1990. Năm đó, *Berners-Lee* và kỹ sư hệ thống dữ liệu *CERN Robert Cailliau* đã hợp tác để cùng yêu cầu tài trợ, nhưng dự án không được *CERN* chính thức thông qua. Trong ghi chú cá nhân của mình từ năm 1990, ông đã liệt kê"một số trong nhiều lĩnh vực mà siêu văn bản được sử dụng" và đặt một cuốn bách khoa toàn thư lên hàng đầu.

- *CSS*:  
 CSS lần đầu tiên được đề xuất bởi *Håkon Wium* Lie vào ngày 10 tháng 10 năm 1994. Vào thời điểm đó, Lie đang làm việc với *Tim Berners-Lee* tại *CERN*. Một số ngôn ngữ định kiểu khác cho web đã được đề xuất cùng lúc và các cuộc thảo luận về danh sách gửi thư cộng đồng cũng như đồng thuận trong *World Wide Web Consortium* dẫn đến *CSS W3C* đầu tiên (CSS1) được phát hành vào năm 1996. Đặc biệt, một đề nghị của *Bert Bos* có ảnh hưởng hơn đó là ông trở thành đồng tác giả của *CSS1* và được coi là đồng sáng tạo *CSS*.

- *Bootstrap:*

Bootstrap ra mắt lần đầu tiên vào ngày 19/8/2011 trên mạng xã hội *GitHub*. Ban đầu, Bootstrap được thiết kế như một mã nguồn mở và có tên gọi là *Twitter Blueprint*. Ngày 31/1/2012, các nhà sáng lập phát hành phiên bản thứ 2 của *Bootstrap*. So với phiên bản cũ, phiên bản số 2 được bổ sung thêm 12 cột giúp người dùng có thể dễ dàng tùy chỉnh và thiết kế với nhiều kích thước màn hình khác nhau. Đến ngày 19/8/2013, phiên bản thứ 3 của *Bootstrap* được ra đời. Đây là phiên bản đầu tiên có giao diện dành cho điện thoại thông minh. Chính nhờ sự thay đổi này đã giúp *Bootstrap* tạo nên sự đột phá và đứng thứ 1 trên Bảng xếp hạng các dự án của *GitHub*. Vào tháng 10/2014, các nhà phát triển tiếp tục nâng cấp và công bố phiên bản thứ 4 của Bootstrap. Tháng 8/2015, *Bootstrap* 4 giới thiệu phiên bản *Alpha* đầu tiên.

- *Python:*

Python đã được [*Guido van Rossum*](https://vi.wikipedia.org/w/index.php?title=Guido_van_Rossum&action=edit&redlink=1) thai nghén vào những năm 1980 tại Trung tâm Toán học – Tin học ([*Centrum Wiskunde & Informatica*](https://vi.wikipedia.org/w/index.php?title=Centrum_Wiskunde_&_Informatica&action=edit&redlink=1), *CWI*) ở [Hà Lan](https://vi.wikipedia.org/wiki/H%C3%A0_Lan) như là một ngôn ngữ kế tục ngôn ngữ [*ABC*](https://vi.wikipedia.org/w/index.php?title=ABC_(ng%C3%B4n_ng%E1%BB%AF_l%E1%BA%ADp_tr%C3%ACnh)&action=edit&redlink=1)– một ngôn ngữ được lấy cảm hứng từ [*SETL*](https://vi.wikipedia.org/w/index.php?title=SETL&action=edit&redlink=1), có khả năng [xử lí ngoại lệ](https://vi.wikipedia.org/w/index.php?title=X%E1%BB%AD_l%C3%AD_ngo%E1%BA%A1i_l%E1%BB%87&action=edit&redlink=1) và giao tiếp với hệ điều hành [*Amoeba*](https://vi.wikipedia.org/w/index.php?title=Amoeba_(h%E1%BB%87_%C4%91i%E1%BB%81u_h%C3%A0nh)&action=edit&redlink=1). Nó bất đầu được triển khai vào tháng Mười Hai năm 1989. *Van Rossum* đã tự mình gánh vác trách nhiệm cho dự án, với vai trò là nhà phát triển chính, cho đến ngày 12 tháng Bảy năm 2018, khi ông thông báo rằng ông sẽ rời bỏ trách nhiệm của ông và cả danh hiệu "[Nhà độc tài nhân từ cho cuộc sống](https://vi.wikipedia.org/w/index.php?title=Nh%C3%A0_%C4%91%E1%BB%99c_t%C3%A0i_nh%C3%A2n_t%E1%BB%AB_cho_cu%E1%BB%99c_s%E1%BB%91ng&action=edit&redlink=1)" của *Python*, một danh hiệu mà cộng đồng *Python* đã trao tặng cho ông vì sự tận tuỵ lâu dài của ông với vai trò là người ra quyết định chính cho dự án. Vào tháng Một năm 2019, các nhà phát triển phần lõi *Python* đã bầu ra một "Hội đồng Chèo lái" gồm năm thành viên để dẫn dắt dự án.

**CHƯƠNG 3: KHẢO SÁT VÀ PHÂN TÍCH THIẾT KẾ HỆ THỐNG**

## Khảo sát thực trạng và yêu cầu bài toán

1. **Khảo sát thực trạng:**

Hiện nay tại Trường Cao Đẳng Nghề Đà Nẵng, số lượng sinh viên tại trường rất nhiều. Hằng ngày, lưu lượng học sinh, sinh viên ra vào trường rất nhiều, khiến cho việc kiểm soát trở nên khó khăn.

Hiện tại, việc quản lý chỉ được giao cho các cán bộ giáo viên và bảo vệ. Việc này không đảm bảo cho việc kiểm soát hết tất cả lưu lượng người ra vào trường.

Hoạt động này diễn ra như sau:

- Học sinh đi vào trường không có sự kiểm soát.

- Bảo vệ chỉ kiểm soát được một phần nhỏ lượng người đi vào trường.

- Các thầy cô đi lấy danh sách điểm danh từng lớp.

Những hoạt động trên cho thấy rằng, khả năng quản lý, kiểm soát bằng sức người không có hiệu quả cao. Vì vậy, cần có một công nghệ có thể hỗ trợ kiểm soát được lưu lượng người ra vào trường. Qua đó, giúp cho người quản lý dể dàng kiểm soát các hoạt động ra vào.

1. **Yêu cầu bài toán:**

Nhà trường yêu cầu xây dựng một ứng dụng hỗ trợ nhận dạng học sinh, sinh viên. Ứng dụng là công cụ hỗ trợ cho nhà trường trong việc quản lý, cũng như kiểm soát học sinh, sinh viên và người lạ. Ứng dụng phải hoạt động ổn định, độ chính xác cao, không xảy ra lỗi không đáng có trong quá trình sử dụng.

1. **Phân tích hệ thống:**
2. **Chức năng hệ thống**

➢ Hệ thống quản trị:

- Quản lí nhận diện sinh viên

- Quản lí nhận diện người lạ

- Quản lý sinh viên

- Quản lý lớp

- Quản lý khoa

➢ Chức năng của người dùng khác:

- Xem trình chiếu camera nhận dạng

- Xem danh sách nhận dạng trong ngày

- Xem danh sách người lạ trong ngày

1. **Yêu cầu chức năng:**

* Quản trị viên:
  + Quản lý toàn bộ các thông tin về nhận dạng sinh viên và người lạ.
  + Quản lý các thông tin về sinh viên
  + Quản lý các thông tin về khoa
  + Quản lý các thông tin về lớp
* Giao diện người dùng:
  + Không nên quá phức tạp.
  + Dung lượng file không quá lớn.
  + Thanh menu thật đơn giản.
  + Font chữ đơn giản, dễ nhìn, màu sắc hài hòa.
  + Website phải tương thích với các trình duyệt phổ biến.
  + Hiển thị được các thông tin về nhận dạng sinh viên.
  + Cho phép quản trị viên quản lí các thông tin về sinh viên, lớp và khoa.

1. **Yêu cầu phi chức năng:**

* Giao diện hệ thống phải dễ sử dụng, trực quan,bắt mắt, thân thiện với người dùng.
* Tính bảo mật và độ an toàn cao.
* Hệ thống phải xử lý nhanh chóng và chính xác.
* Tương thích với đa phần các trình duyệt web hiện tại

1. **Các tác nhân của hệ thống:**

**\*Tác nhân**

- Tác nhân người quản trị (Administrator): Người quản lý các chức năng quản trị hệ thống, quản lý danh sách sinh viên, danh sách lớp, danh sách khoa, danh sách nhận diện người lạ và sinh viên

- Tác nhân người dùng: Người dùng có thể: xem trình chiếu camera nhận dạng, xem danh sách nhận dạng trong ngày, xem danh sách người lạ trong ngày, đưa khuôn mặt vào camera để nhận diện

**\*Ca sử dụng**

*- Đăng nhập*

|  |  |
| --- | --- |
| **Hành động của tác nhân** | **Hành động của hệ thống** |
| 1. Người quản lý tiến hành đăng nhập tài khoản |  |
| 1. Người quản lý nhập đầy đủ thông tin tài khoản |  |
| 1. Người quản lý nhấn nút đăng nhập | 1. Hệ thống kiểm tra thông tin hợp lệ theo yêu cầu. Yêu cầu nhập lại nếu sai. |
|  | 1. Xử lý yêu cầ và xác nhận đăng nhập |
|  | 1. Kết thúc phiên đăng nhập |
|  | 1. Chuyển đến trang quản lý |

*- Thêm khoa:*

|  |  |
| --- | --- |
| **Hành động của tác nhân** | **Hành động của hệ thống** |
| 1. Người quản lý tiến hành thêm Khoa vào hệ thống |  |
| 1. Người quản lý nhập đầy đủ thông tin Khoa | 1. Xử lý yêu cầu và xác nhận thêm |
|  | 1. Cập nhật danh sách Khoa |

*- Sửa khoa:*

|  |  |
| --- | --- |
| **Hành động của tác nhân** | **Hành động của hệ thống** |
| 1. Người quản lý tiến hành sửa Khoa |  |
| 1. Người quản lý nhập thông tin Khoa | 1. Xử lý yêu cầu và xác nhận sửa |
|  | 1. Cập nhật danh sách Khoa |

*- Xóa khoa:*

|  |  |
| --- | --- |
| **Hành động của tác nhân** | **Hành động của hệ thống** |
| 1. Người quản lý tiến hành xóa Khoa | 1. Xử lý yêu cầu và xác nhận xóa |
|  | 1. Cập nhật danh sách Khoa |
|  | 1. Cập nhật danh sách sinh viên |

*- Thêm lớp:*

|  |  |
| --- | --- |
| **Hành động của tác nhân** | **Hành động của hệ thống** |
| Người quản lý tiến hành thêm lớp vào hệ thống |  |
| Người quản lý nhập đầy đủ thông tin lớp | Xử lý yêu cầu và xác nhận thêm |
|  | Cập nhật danh sách lớp |

*- Sửa lớp:*

|  |  |
| --- | --- |
| **Hành động của tác nhân** | **Hành động của hệ thống** |
| Người quản lý tiến hành sửa lớp |  |
| Người quản lý nhập thông tin lớp | Xử lý yêu cầu và xác nhận sửa |
|  | Cập nhật danh sách lớp |
|  | Cập nhật danh sách sinh viên |

*- Xóa lớp:*

|  |  |
| --- | --- |
| **Hành động của tác nhân** | **Hành động của hệ thống** |
| Người quản lý tiến hành xóa lớp | Xử lý yêu cầu và xác nhận xóa |
|  | Cập nhật danh sách lớp |
|  | Cập nhật danh sách sinh viên |

*- Thêm sinh viên:*

|  |  |
| --- | --- |
| **Hành động của tác nhân** | **Hành động của hệ thống** |
| 1. Người quản lý tiến hành thêm sinh viên |  |
| 1. Người quản lý nhập đầy đủ thông tin sinh viên | 1. Hệ thống kiểm tra thông tin hợp lệ theo yêu cầu. Yêu cầu nhập lại nếu không chuẩn. |
|  | 1. Xử lý yêu cầu và xác nhận thêm |
|  | 1. Cập nhật danh sách sinh viên |

*- Sửa sinh viên:*

|  |  |
| --- | --- |
| **Hành động của tác nhân** | **Hành động của hệ thống** |
| 1. Người quản lý tiến hành sửa sinh viên |  |
| 1. Người quản lý nhập đầy đủ thông tin sinh viên | 1. Hệ thống kiểm tra thông tin hợp lệ theo yêu cầu. Yêu cầu nhập lại nếu không chuẩn. |
|  | 1. Xử lý yêu cầu và xác nhận sửa |
|  | 1. Cập nhật danh sách sinh viên |

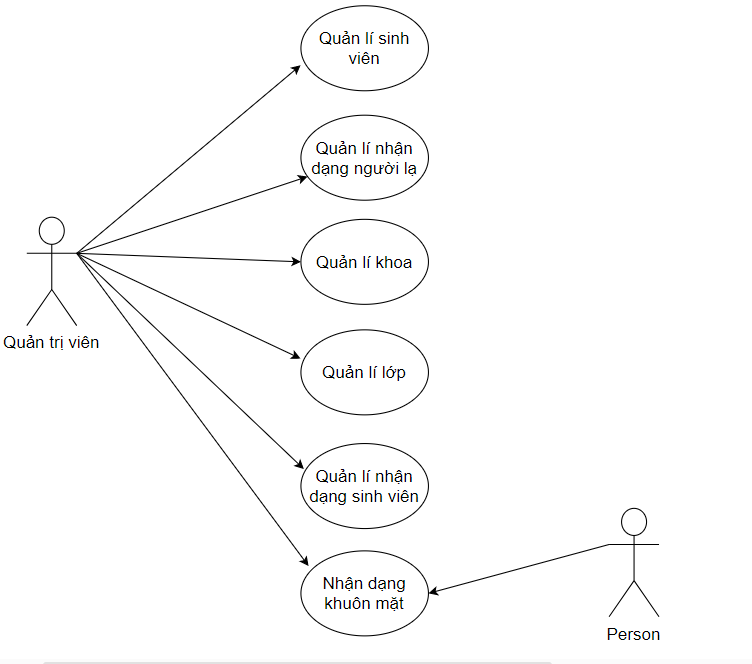
*- Xóa sinh viên*

|  |  |
| --- | --- |
| **Hành động của tác nhân** | **Hành động của hệ thống** |
| 1. Người quản lý tiến hành xóa sinh viên | 1. Xử lý yêu cầu và xác nhận xóa |
|  | 1. Cập nhật danh sách sinh viên |

**-** *Xóa danh sách nhận dạng:*

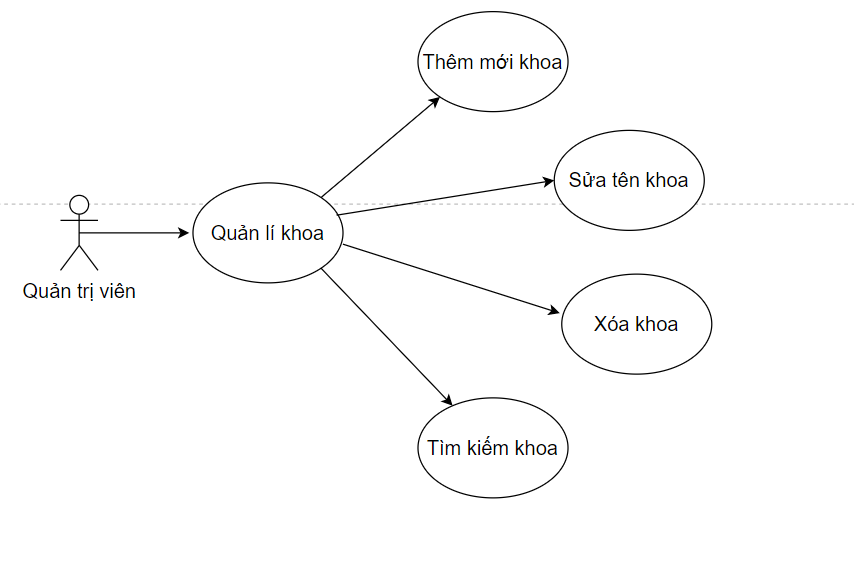
|  |  |
| --- | --- |
| **Hành động của tác nhân** | **Hành động của hệ thống** |
| 1. Người quản lý tiến hành xóa danh sách nhận dạng | 1. Xử lý yêu cầu và xác nhận xóa |
|  | 1. Cập nhật danh sách sinh viên |

1. **Biểu đồ Use Case:**
   * 1. *Biểu đồ Use Case tổng quát:*



Hình ảnh . Use case tổng quát

* + 1. *Use Case quản lý khoa:*



Hình ảnh . Use case quản lý khoa

* Điều kiện trước: Quản trị viên phải đăng nhập
* Dòng sự kiện chính:

**-** Người dùng chọn kiểu tác động: thêm, sửa, xóa, tìm kiếm thông tin về khoa

-Thêm khoa:

 Hệ thống yêu cầu người dùng cung cấp thông tin về khoa

 Người dùng nhập xong thông tin khoa sẽ bấm thêm mới.

 Hệ thống kiểm tra lại thông tin và thông báo thành công/thất bại cho người dùng.

**-** Sửa thông tin về khoa:

 Hệ thống yêu cầu người dùng cung cấp lại thông tin khoa.

 Người dùng nhập lại thông tin khoa sẽ sửa.

 Hệ thống kiểm tra lại thông tin và thông báo thành công/thất bại cho người dùng

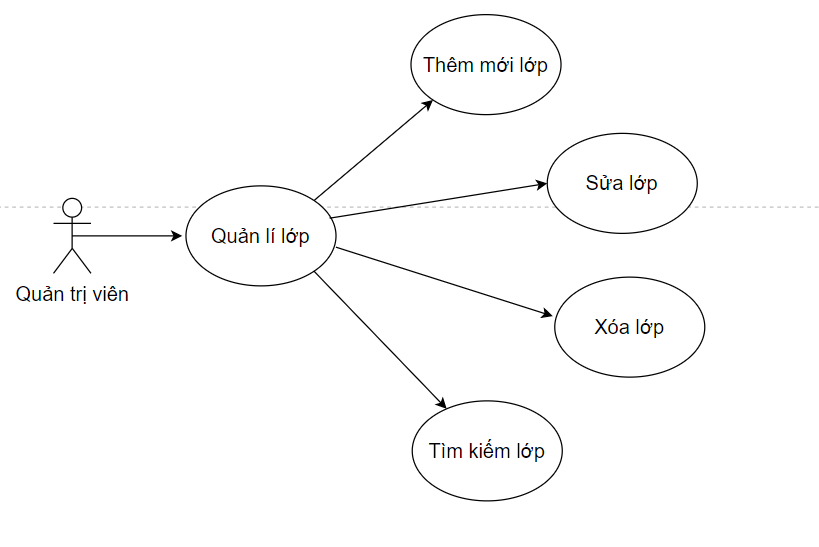
**-** Xóa thông tin khoa:

 Hệ thống hiển thị danh khoa để người dùng chọn.

 Người dùng chọn khoa cần xóa

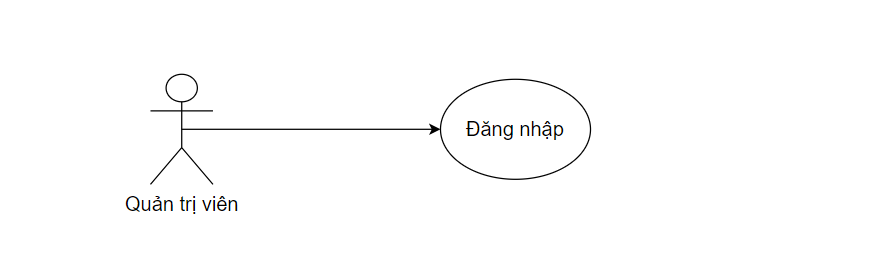
 Hệ thống kiểm tra lại thông tin và thông báo thành công/thất bại cho người dùng.

* + 1. Use Case quản lý lớp:



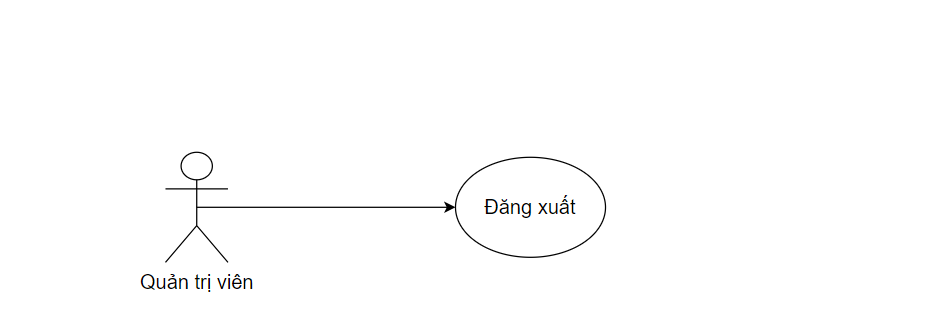
Hình ảnh . Use case quản lý lớp

* Tác nhân bao gồm (người dùng): Quản trị viên
* Mô tả: Use case này mô tả các bước người quản trị tác động vào hệ thống.
* Điều kiện trước: Quản trị viên phải đăng nhập
* Dòng sự kiện chính:
  + Người dùng chọn kiểu tác động: thêm, sửa, xóa, tìm kiếm thông tin về lớp
  + Thêm lớp:
    - Hệ thống yêu cầu người dùng cung cấp thông tin về lớp.
    - Người dùng nhập xong thông tin lớp sẽ bấm thêm mới.
    - Hệ thống kiểm tra lại thông tin và thông báo thành công/thất bại cho người dùng.
  + Sửa thông tin về lớp:
    - Hệ thống yêu cầu người dùng cung cấp lại thông tin lớp.
    - Người dùng nhập lại thông tin lớp sẽ sửa.
    - Hệ thống kiểm tra lại thông tin và thông báo thành công/thất bại cho người dùng.
  + Xóa thông tin lớp:
    - Hệ thống hiển thị danh lớp để người dùng chọn.
    - Người dùng chọn lớp cần xóa.
    - Hệ thống kiểm tra lại thông tin và thông báo thành công/thất bại cho người dùng.
    1. *Usecase đăng nhập:*



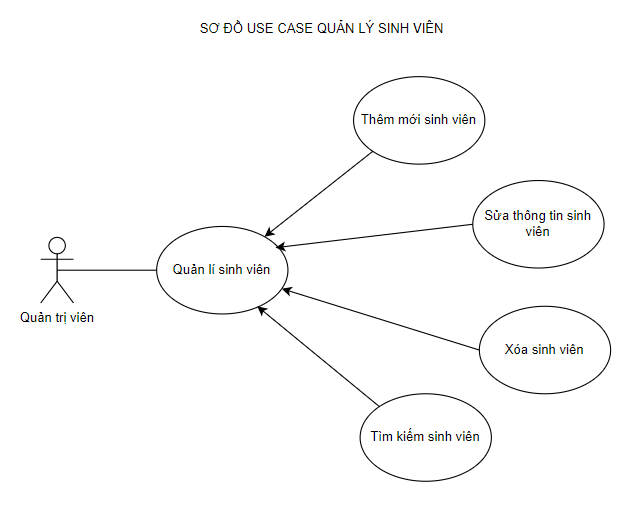
Hình ảnh . Use case đăng nhập

* Tác nhân bao gồm (người dùng): Quản trị viên
* Mô tả: Use case này mô tả các bước đăng nhập của người quản trị vào hệ thống.
* Điều kiện trước:
* Dòng sự kiện chính:
  + Hệ thống yêu cầu người dùng cung cấp thông tin đăng nhập gồm tên tài khoản và mật khẩu.
  + Người dùng nhập xong thông tin đăng nhập và click nút gửi.
  + Hệ thống kiểm tra lại thông tin đăng nhập và thông báo thành công/thất bại cho người dùng. Nếu đăng nhập thành công hệ thống dựa trên thông tin đăng nhập sẽ đồng thời phân quyền tùy theo loại người dùng. Nếu đăng nhập thất bại, hệ thống sẽ hiện thông báo cho người dùng và yêu cầu đăng nhập lại.
    1. *Use Case đăng xuất:*



Hình ảnh . Use case đăng xuất

* Tác nhân bao gồm (người dùng): Quản trị viên.
* Mô tả: Use case này mô tả các bước đăng xuất khỏi hệ thống của người dùng đã đăng nhập vào hệ thống.
* Điều kiện trước: người dùng đăng nhập.
* Dòng sự kiện chính:
  + Người dùng thực hiện chức năng đăng xuất khỏi hệ thống khi click vào nút đăng xuất.
  + Hệ thống đăng xuất tài khoản người dùng khỏi hệ thống.
    1. *Use Case quản lý sinh viên:*



Hình ảnh . Use case quản lý sinh viên

* Tác nhân bao gồm (người dùng): Quản trị viên
* Mô tả: Use case này mô tả các thao tác của người quản trị đối với dữ liệu sinh viên.
* Điều kiện trước: Tài khoản quản trị viên đã được đăng nhập
* Dòng sự kiện chính:
* Thêm sinh viên:

1. Quản trị viên nhập thông tin sinh viên vào ứng dụng

2. Hệ thống kiểm tra thông tin và hình ảnh hợp lệ, cho phép thêm sinh viên vào hệ thống, thông báo thêm thành công

* Sửa thông tin sinh viên:

1. Quản trị viên click nút chỉnh sửa tại hàng sinh viên muốn sửa

2. Quản trị viên nhập thông tin mới của sinh viên vào ứng dụng

3. Hệ thống kiểm tra thông tin và hình ảnh hợp lệ, cho phép cập nhật sinh viên vào hệ thống, thông báo cập nhật thành công

* Xóa dữ liệu sinh viên:

1. Quản trị viên click nút xóa tại hàng sinh viên muốn xóa

2. Hệ thống hỏi lại có đồng ý xóa hay không

3. Quản trị viên click đồng ý

4. Hệ thống xóa sinh viên khỏi cơ sở dữ liệu

* Tìm kiếm thông tin sinh viên

1. Quản trị viên click nhập thông tin đặc trưng sinh viên vào ứng dụng

2. Hệ thống kiểm tra rồi đưa ra kết quả

* Dòng sự kiện ngoại lệ:
* Thêm sinh viên:

2b. Hệ thống kiểm tra thông tin và hình ảnh không hợp lệ, hiển thị thông báo thêm không thành công và nguyên nhân

* Sửa thông tin sinh viên:

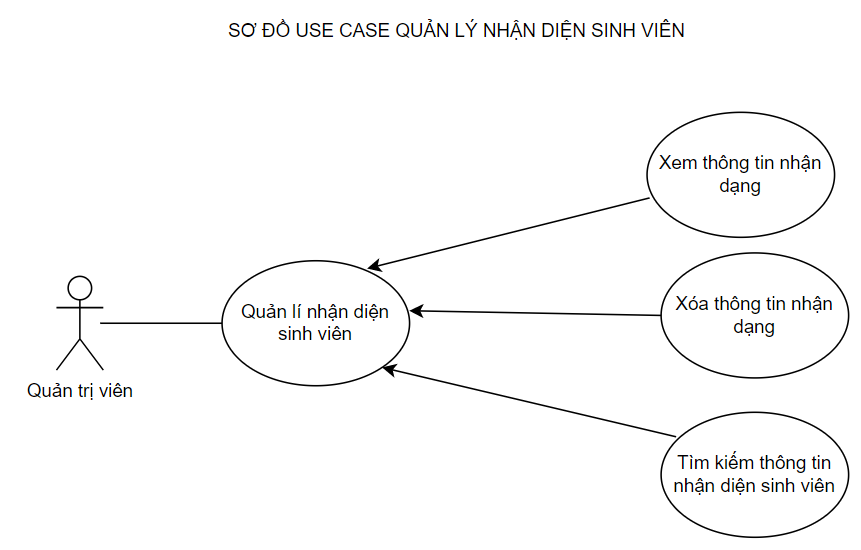
3b. Hệ thống kiểm tra thông tin và hình ảnh không hợp lệ, hiển hị thông báo cập nhật không thành công và nguyên nhân

* Xóa dữ liệu sinh viên:

3b. Quản trị viên click đóng

4b. Hệ thống hủy đi thông báo

* + 1. *Use Case nhận dạng sinh viên:*



Hình ảnh . Use case nhận dạng sinh viên

* Tác nhân bao gồm (người dùng): Quản trị viên
* Mô tả: Use case này mô tả các thao tác của người quản trị đối với dữ liệu nhận diện sinh viên.
* Điều kiện trước: Tài khoản quản trị viên đã được đăng nhập
* Dòng sự kiện chính:
* Tìm kiếm thông tin nhận diện sinh viên

1. Quản trị viên click nhập thông tin sinh viên cần tìm vào ứng dụng

2. Hệ thống xử lý tìm kiếm rồi đưa ra kết quả

* Xóa dữ liệu nhận diện sinh viên:

1. Quản trị viên click nút xóa tại hàng nhận diện sinh viên muốn xóa

2. Hệ thống hỏi lại có đồng ý xóa hay không

3. Quản trị viên click đồng ý

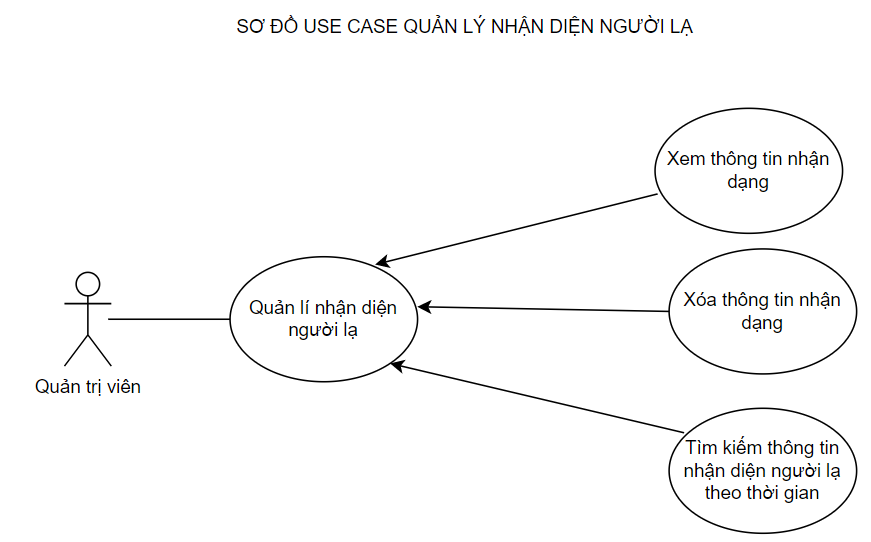
4. Hệ thống xóa sinh viên khỏi cơ sở dữ liệu

* Dòng sự kiện ngoại lệ:
* Xóa dữ liệu nhận diện sinh viên:

3b. Quản trị viên click đóng

4b. Hệ thống hủy đi thông báo

* + 1. *Use Case nhận diện người lạ:*



Hình ảnh . Use case nhận diện người lạ

* Tác nhân bao gồm (người dùng): Quản trị viên
* Mô tả: Use case này mô tả các thao tác của người quản trị đối với dữ liệu nhận diện sinh viên.
* Điều kiện trước: Tài khoản quản trị viên đã được đăng nhập
* Dòng sự kiện chính:
* Tìm kiếm thông tin nhận diện người lạ

1. Quản trị viên click nhập thời gian cần tìm vào ứng dụng

2. Hệ thống xử lý tìm kiếm rồi đưa ra kết quả

* Xóa dữ liệu nhận diện người lạ:

1. Quản trị viên click nút xóa tại hàng nhận diện người lạ muốn xóa

2. Hệ thống hỏi lại có đồng ý xóa hay không

3. Quản trị viên click đồng ý

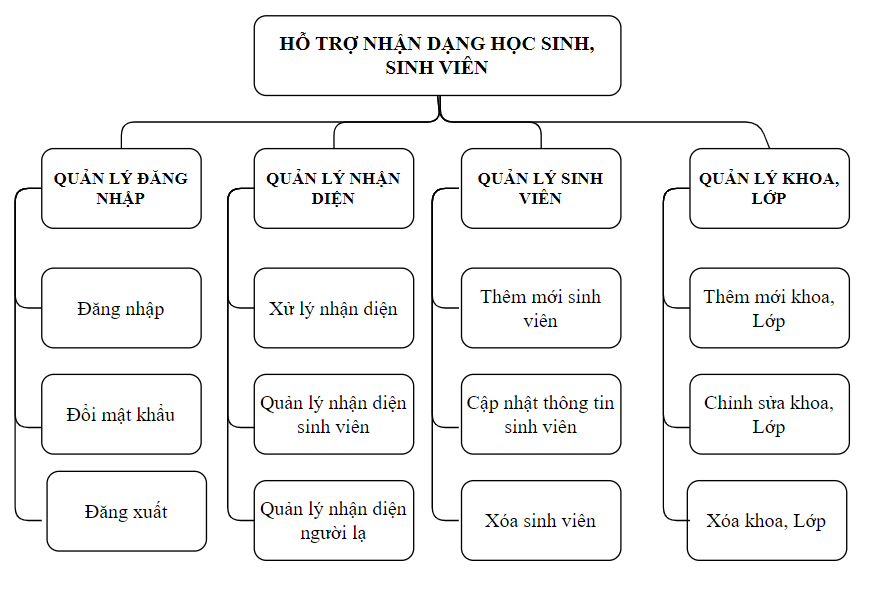
4. Hệ thống xóa nhận diện người lạ đó khỏi cơ sở dữ liệu

* + Dòng sự kiện ngoại lệ:
* Xóa dữ liệu nhận diện người lạ:

3b. Quản trị viên click đóng

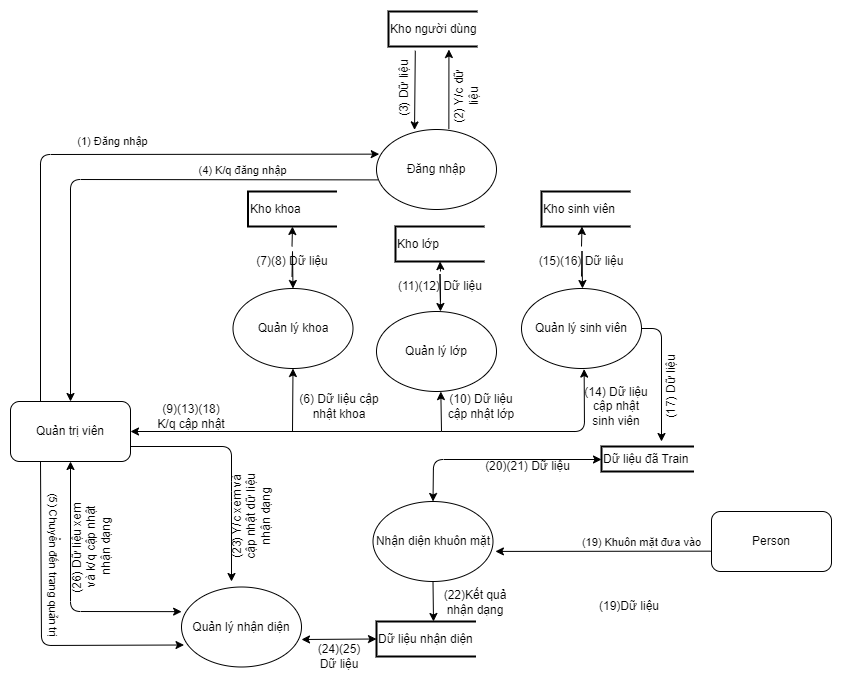
4b. Hệ thống hủy đi thông báo

1. **Sơ đồ ngữ cảnh:**



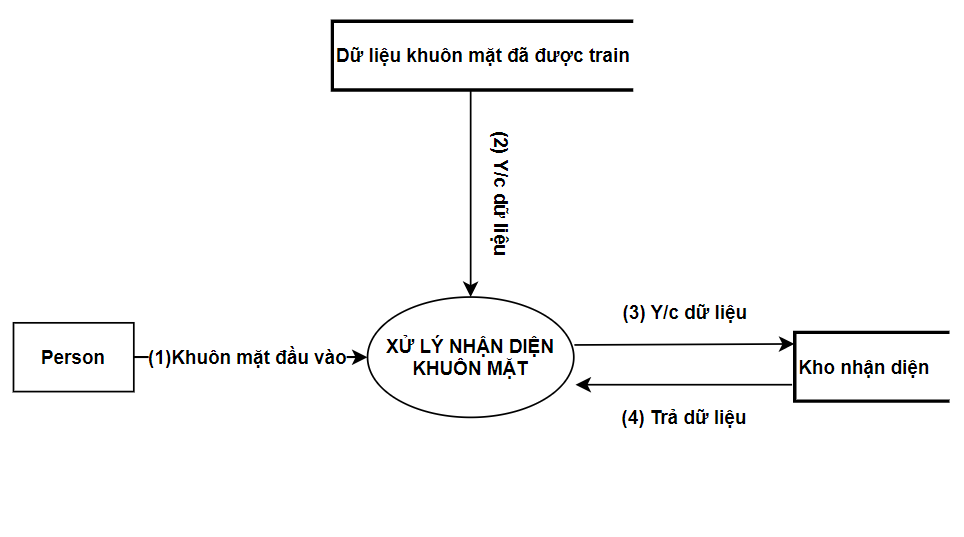
Hình ảnh . Sơ đồ ngữ cảnh

1. **Sơ đồ luồng dữ liệu DFD:**
2. *Sơ đồ DFD mức đỉnh:*

****

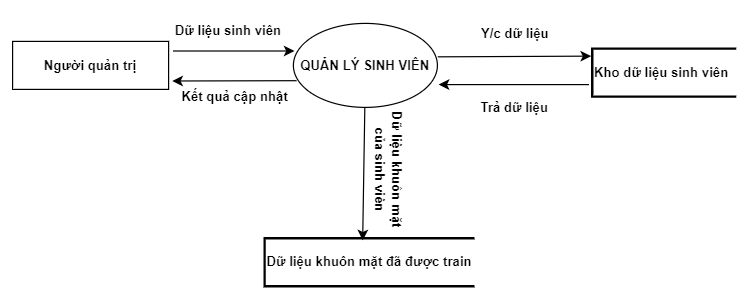
Hình ảnh . Sơ đồ DFD mức đỉnh

1. *Sơ đồ DFD mức dưới đỉnh (chức năng nhận diện khuôn mặt):*

****

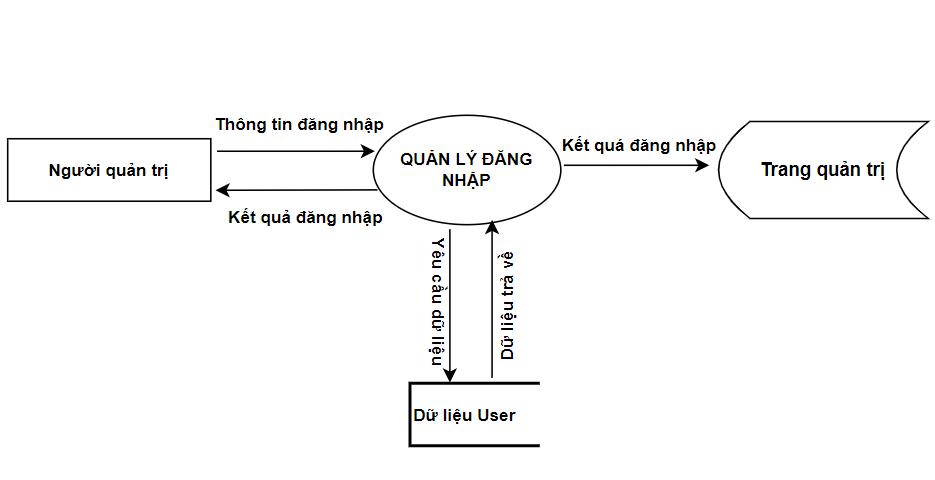
Hình ảnh . Sơ đồ DFD mức dưới đỉnh (nhận diện khuôn mặt)

1. *Sơ đồ DFD mức dưới đỉnh (chức năng quản lý sinh viên):*

****

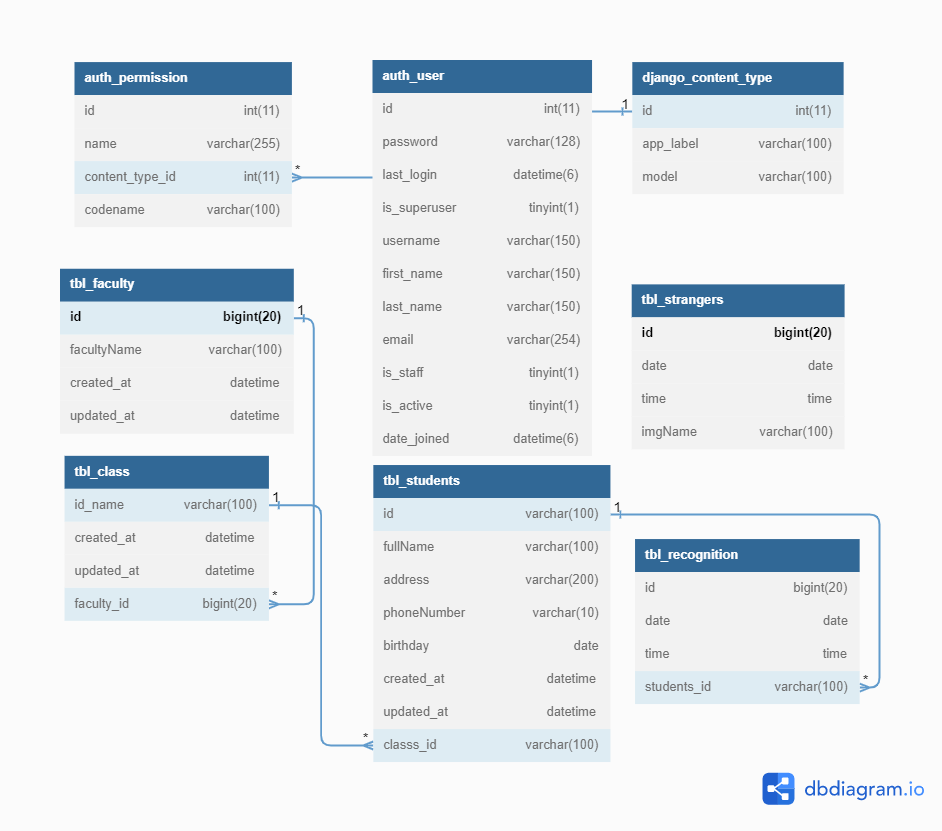
Hình ảnh . Sơ đồ DFD mức dưới đỉnh (quản lý sinh viên)

1. *Sơ đồ mức dưới đỉnh (chức năng quản lý đăng nhập):*

****

Hình ảnh . Sơ đồ DFD mức dưới đỉnh (quản lý đăng nhập)

1. **Thiết kế:**
2. **Danh sách chi tiết các bảng dữ liệu:**

****

Hình ảnh . Sơ đồ quan hệ và chi tiết các bảng dữ liệu

# CHƯƠNG 4: XÂY DỰNG ỨNG DỤNG

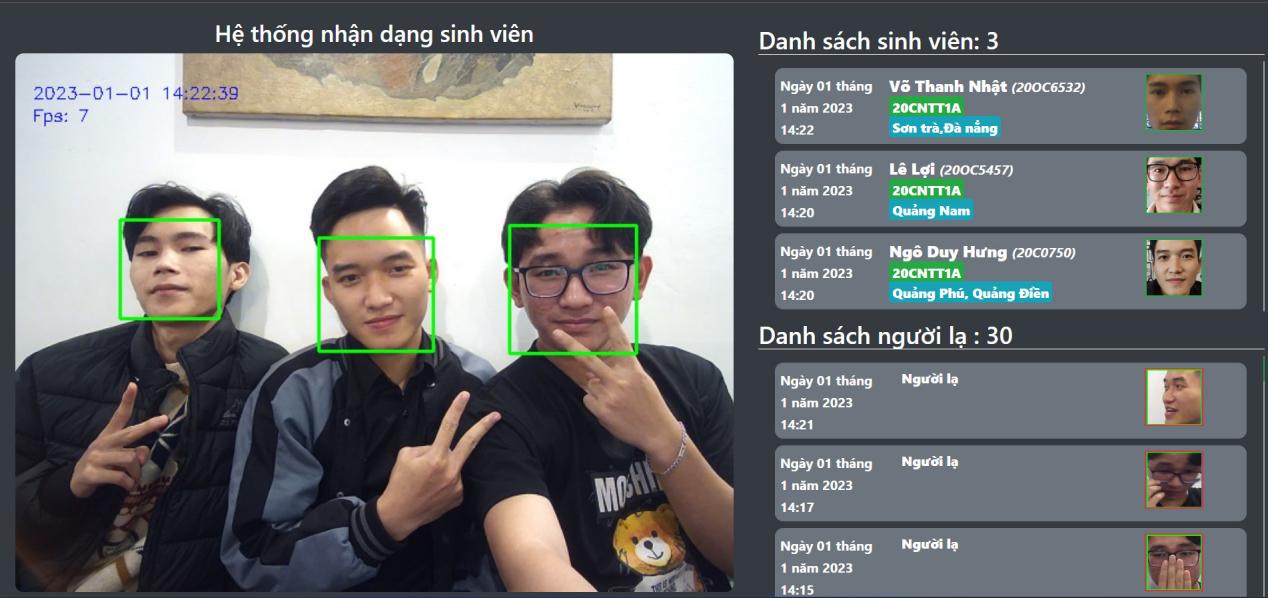
1. **Chuẩn bị:**

- Môi trường Python và Conda.

- Nắm chắc kiến thức về Framework Django

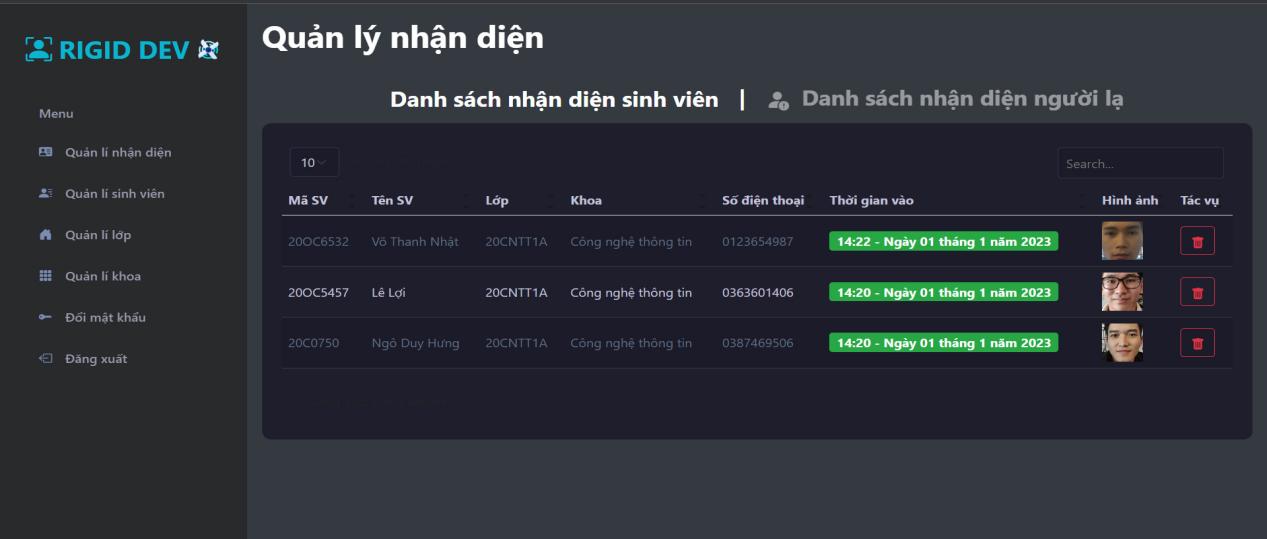
- Nắm chắc kiến thức về các ngôn ngữ giao diện HTML, CSS, Bootstrap, Javascript

1. **Giao diện:**
2. **Giao diện chính:**



Hình ảnh . Giao diện chính

1. **Giao diện quản trị:**
   * 1. *Giao diện quản lý nhận diện:*

****

Hình ảnh . Giao diện quản lý nhận diện (sinh viên)

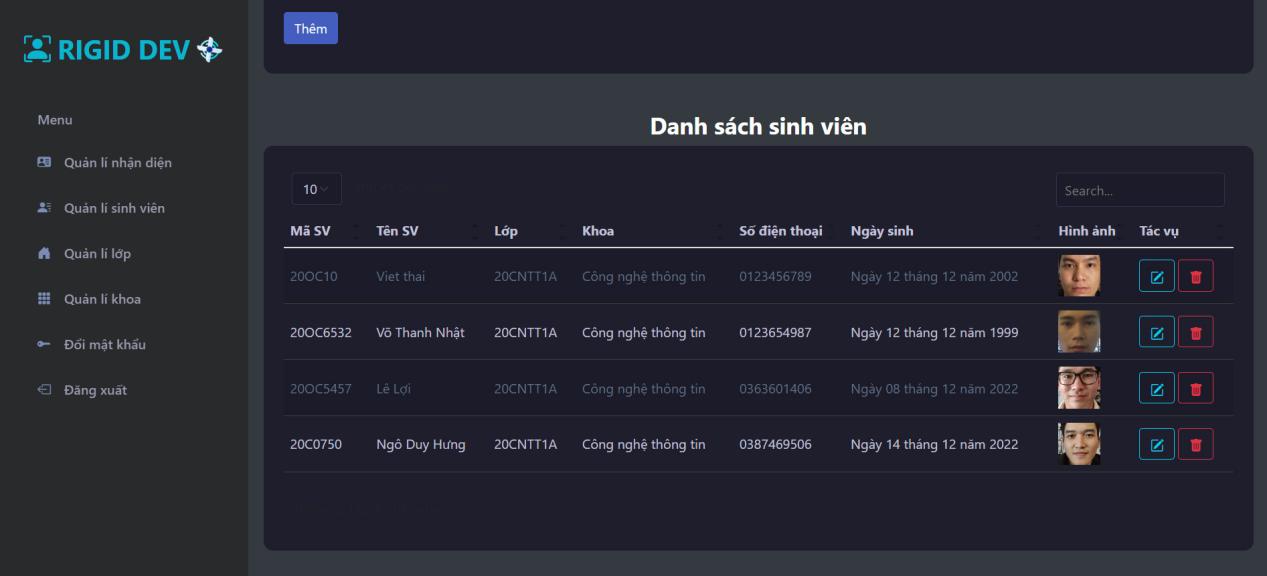
****

Hình ảnh . Giao diện quản lý nhận diện (người lạ)

* + 1. *Giao diện quản lý sinh viên:*

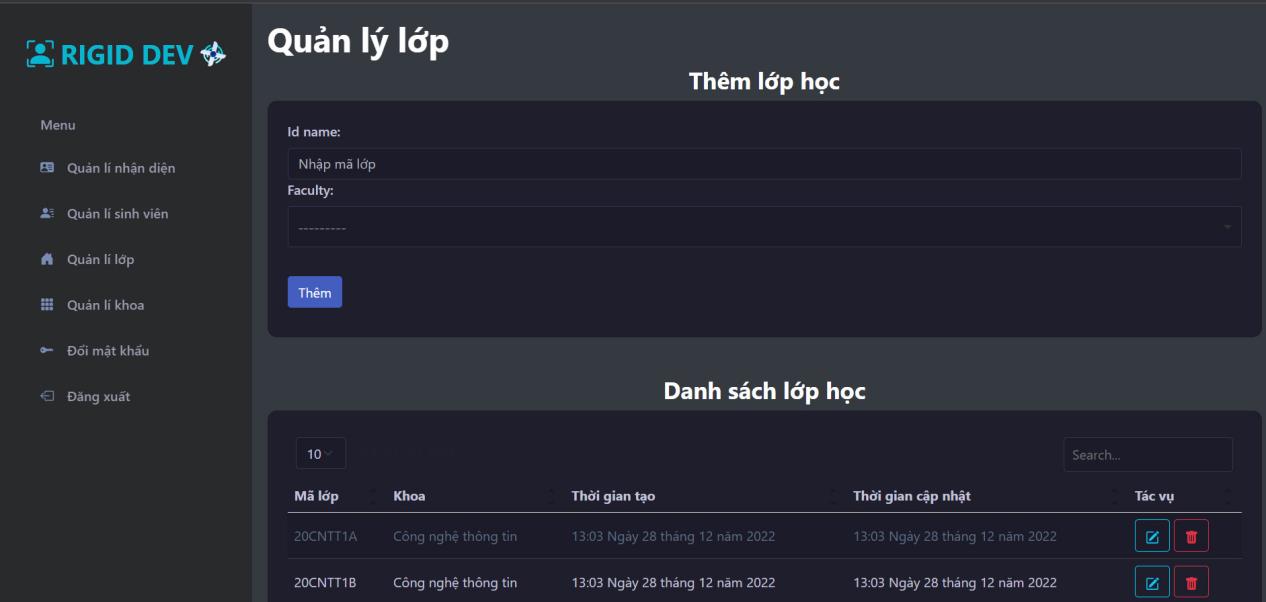
****

Hình ảnh . Giao diện quản lý sinh viên (thêm)

****

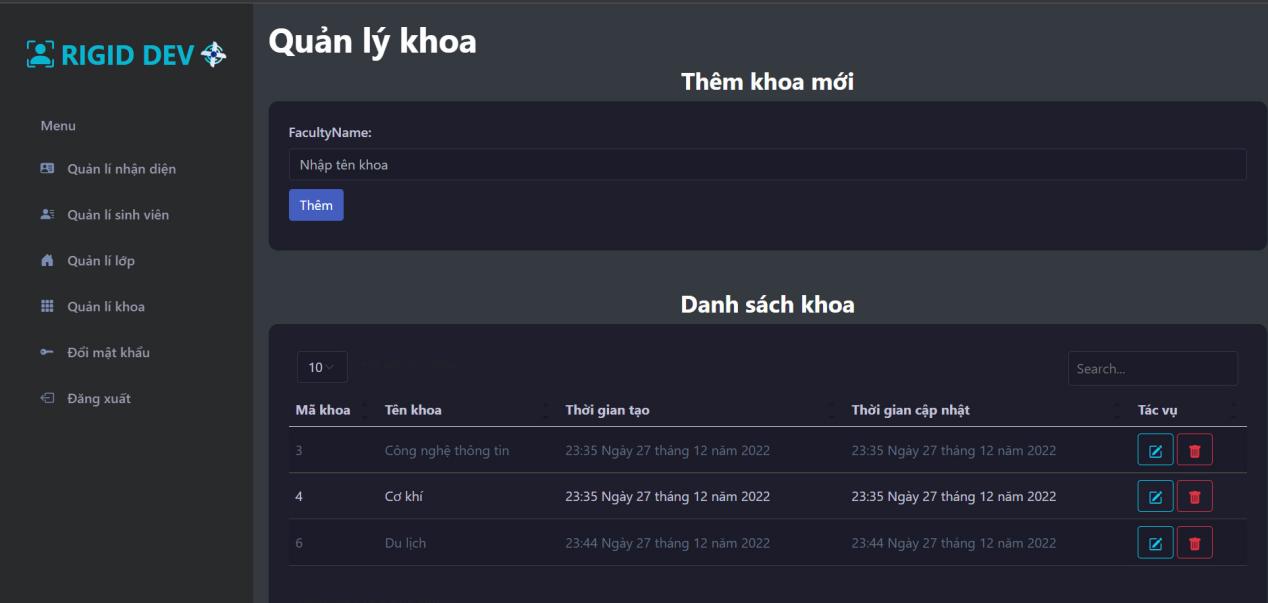
Hình ảnh . Giao diện quản lý sinh viên (danh sách)

* + 1. *Giao diện quản lý lớp:*

****

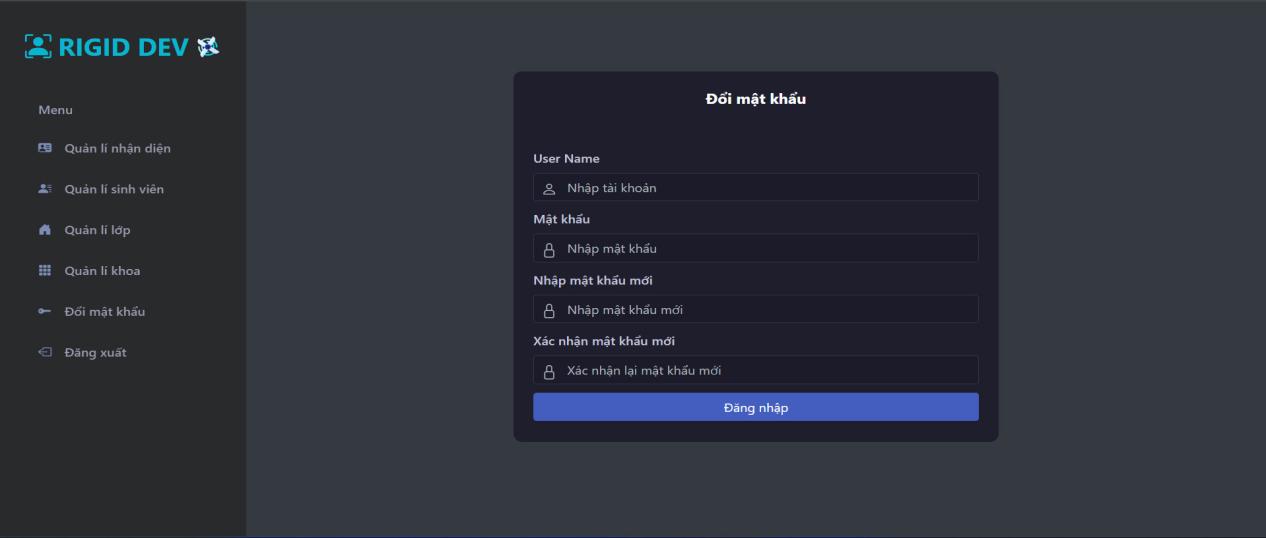
Hình ảnh . Giao diện quản lý lớp (thêm)

* + 1. *Giao diện quản lý khoa:*

****

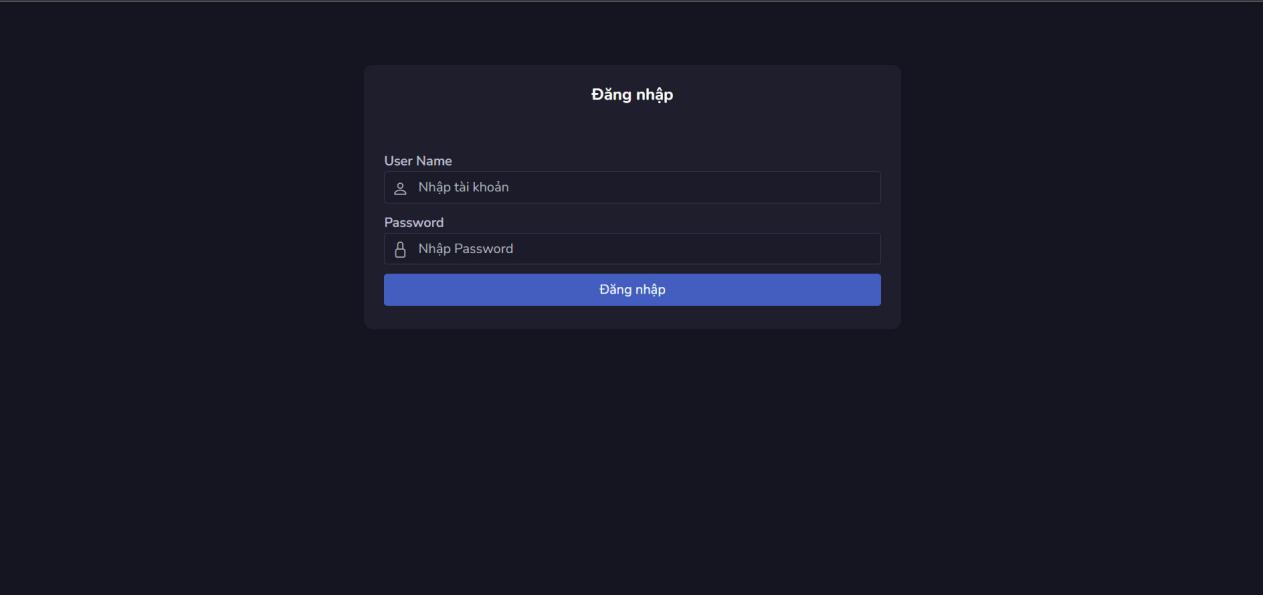
Hình ảnh . Giao diện quản lý lớp (danh sách)

* + 1. *Giao diện đổi mật khẩu:*

****

Hình ảnh . Giao diện đổi mật khẩu

* + 1. *Giao diện đăng xuất:*

****

Hình ảnh . Giao diện đăng xuất

**CHƯƠNG 5: KẾT LUẬN**

1. **Kết quả đạt được:**

**-** Về cơ bản đã hoàn thành được yêu cầu của bài toán, phân tích và thiết kế hệ thống nhận diện khuôn mặt.

- Đã hoàn thành đúng tiến độ yêu cầu.

- Về mặt giao diện, đơn giản, dễ sử dụng, không cầu kì.

- Về mặt người dùng, dễ tiếp cận, dễ hiểu.

- Về mặt quản trị, có các chức năng thêm sửa xóa cơ bản để trợ giúp trong việc quản lý nhận dạng.

1. **Hướng phát triển:**

**-** Hệ thống nhận dạng chính xác hơn.

- Đồng bộ dữ liệu với kho dữ liệu có sẵn.

- Tăng tốc độ nhận dạng.

- Tăng tầm nhận dạng.

- Phát triển thành app và có thể quản lý ở bất cứ đâu.

**TÀI LIỆU THAM KHẢO**

1. Thanh T. (2021, July 29). *OpenCV là gì? OpenCV được sử dụng để làm gì?* Tài Liệu Hướng Dẫn Và Tối Ưu Dịch Vụ. <https://wiki.tino.org/opencv-la-gi/>
2. Wikipedia contributors. (2022c, December 10). *HTML*. https://vi.wikipedia.org/wiki/HTML
3. Wikipedia contributors. (2022b, October 13). *CSS*. https://vi.wikipedia.org/wiki/CSS
4. Wikipedia contributors. (2022a, August 27). *Bootstrap*. https://vi.wikipedia.org/wiki/Bootstrap
5. Trần, Q. (2023, January 2). *MediaPipe: Live ML Solutions và ứng dụng vẽ bằng Hands Gestures*. Viblo. https://viblo.asia/p/mediapipe-live-ml-solutions-va-ung-dung-ve-bang-hands-gestures-gAm5ymOV5db
6. Wikipedia contributors. (2023, January 1). *Python (ngôn ngữ lập trình)*. https://vi.wikipedia.org/wiki/Python\_(ng%C3%B4n\_ng%E1%BB%AF\_l%E1%BA%ADp\_tr%C3%ACnh)
7. *Lập trình web với Django? Django là gì? Tại sao lại dùng Django Framework?* (n.d.). https://niithanoi.edu.vn/django-la-gi.html
8. Blog, T. (2022, July 5). *Thuật toán CNN là gì? Cấu trúc mạng Convolutional Neural Network*. TopDev. https://topdev.vn/blog/thuat-toan-cnn-convolutional-neural-network/