

# Hệ thống nhận diện sinh viên ra vào ký túc xá

## I. Tổng quan hệ thống

**Tên hệ thống:** Hệ thống nhận diện khuôn mặt sinh viên ra vào ký túc xá (KTX).

Phát triển hệ thống xử lý hình ảnh thời gian thực để nhận diện sinh viên ra/vào ký túc xá. Hệ thống cần có khả năng:

- Nhận diện khuôn mặt từ camera theo thời gian thực.
- So sánh và xác thực khuôn mặt với cơ sở dữ liệu.
- Ghi nhận lịch sử vào/ra của sinh viên.
- Hỗ trợ quản trị hệ thống: thêm/xóa/sửa người dùng.
- Được đóng gói toàn bộ bằng Docker, đẩy lên Docker Hub, sẵn sàng triển khai bằng Docker Compose.

## II. Kiến trúc hệ thống

Hệ thống chia thành 4 thành phần chính:

### 1. Frontend Giao Diện Người Dùng (User Interface - UI):

- Hiển thị video thời gian thực từ camera.
- Ghi nhận khuôn mặt nhận diện được + trạng thái (ra/vào).
- Hiển thị MSV, tên sinh viên, thời gian, phòng, trạng thái xác thực.

### 2. Frontend Admin:

- Dashboard: Quản lý sinh viên, ghi nhận ra/vào, lịch sử ra/vào, training dữ liệu.
- Chức năng : Thêm sinh viên và sửa thông tin sinh viên ( bắt buộc sinh viên phải xác thực khuôn mặt ), xóa sinh viên, quản lý lịch sử ra/ vào của sinh viên.

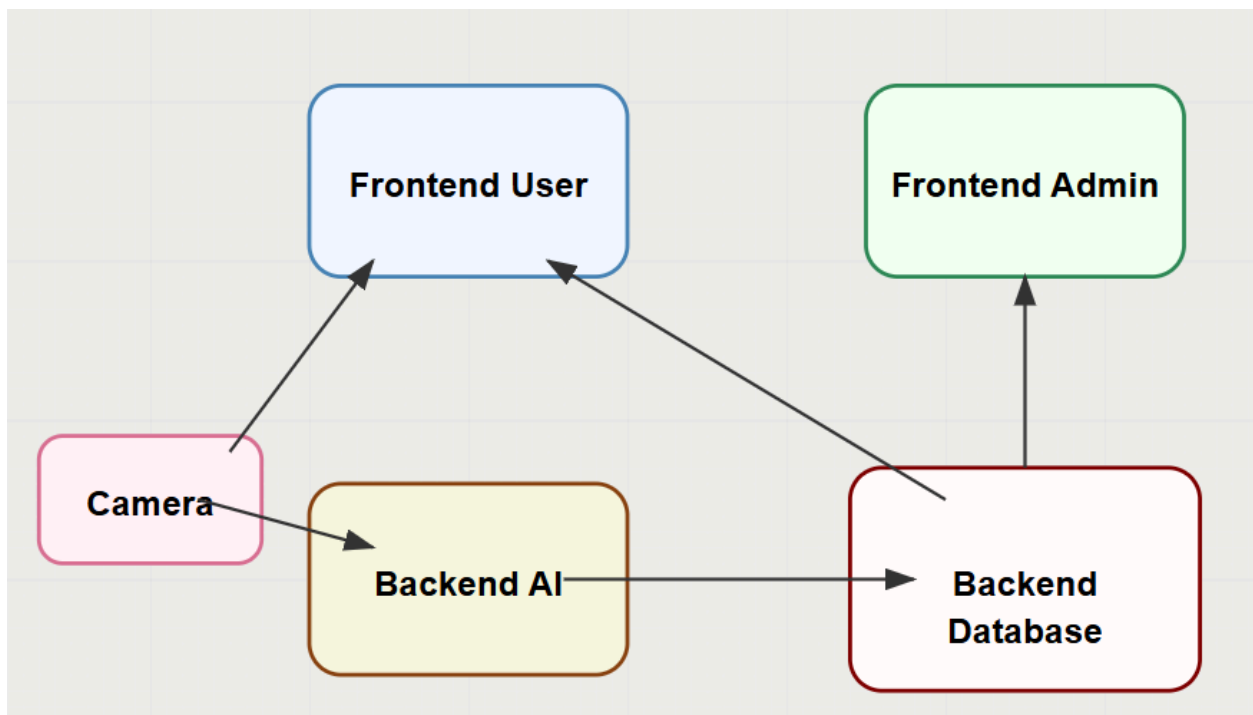
### 3. Backend Xử Lý Hình Ảnh (AI Server):

- Nhận dữ liệu ảnh/video từ camera.
- Dự đoán khuôn mặt bằng mô hình AI.
- Trả kết quả nhận dạng về frontend + lưu vào backend cơ sở dữ liệu.

#### 4. Backend Dữ Liệu:

- Ghi lại thông tin vào/ra ktx, thông tin người dùng, ảnh khuôn mặt của sinh viên.
- Cơ sở dữ liệu: MySQL (thông tin sinh viên).

### III. Sơ đồ kiến trúc tổng thể



### IV. Thiết Kế Thành Phần Cụ Thể

#### 1. Frontend Người Dùng

Các công nghệ sử dụng: HTML, CSS, JavaScript, Nginx

- Hiển thị giao diện camera, thông tin sinh viên vào/ra.

- Hiển thị: ảnh khuôn mặt trích xuất, tên sinh viên, mã số sinh viên, thời gian vào/ra sau khi sinh viên nhận diện.
- Kết nối với hệ thống để nhận kết quả từ AI server thời gian thực.

## 2. Frontend Admin

Các công nghệ sử dụng: HTML, CSS, JavaScript, Nginx

- Trang quản lý sinh viên:
  - Thêm/xoá/cập nhật thông tin sinh viên tại mục quản lý sinh viên.
  - Yêu cầu sinh viên chụp ảnh mới để xác thực khuôn mặt khi thêm sinh viên.

## 3. Backend AI

Các công nghệ sử dụng: Flask, DeepFace, TensorFlow, OpenCV

- Nhận input từ camera.
- Xử lý ảnh:
  - Tiền xử lý → trích đặc trưng → so sánh vector.
- Xác thực độ giống (cosine similarity > threshold).
- Gửi kết quả (mã SV, họ tên, thời gian, phòng) về frontend và DB.

## 4. Backend Dữ Liệu

MySQL:

- Bảng `access_logs`: Ghi nhận lịch sử ra/ vào của sinh viên.  
Sau khi sinh viên nhận diện ra/ vào kết quả sẽ được lưu vào bảng với các thông tin thời gian ra/vào, họ và tên, MSV, phòng.
- Bảng `admin_accounts`: Lưu trữ thông tin tài khoản mật khẩu của admin.
- Bảng `students`: Thông tin sinh viên đang ở trong KTX bao gồm MSV, Họ và tên, Phòng, Lớp, Trạng thái. Khi sinh viên ra khỏi KTX sẽ xóa sinh viên đi.

## V. Đóng gói qua Docker

- **Mục tiêu:** Đóng gói toàn bộ hệ thống (frontend, backend, AI model, database) để dễ triển khai, chạy nhất quán và di động giữa các môi trường.
- **Công nghệ sử dụng:**
  - **Docker** để tạo các container riêng biệt cho từng thành phần.
  - **Docker Compose** để cấu hình và khởi chạy nhiều container cùng lúc.
- **Thành phần được đóng gói:**
  - `frontend` (giao diện người dùng và/hoặc admin)
  - `backend` (xử lý AI, API, dữ liệu)
  - `database` (MySQL Server)
  - `nginx` (nếu dùng để reverse proxy hoặc load balancing)

## VI. Use Case hệ thống

### UC-01: Nhận Diện Sinh Viên

**Nhận diện bằng camera xác thực:**

1. **Mục tiêu:** Nhận diện sinh viên ra/vào KTX.
2. **Actor chính:** Sinh viên
3. **Tiền điều kiện:** Hệ thống hoạt động, người dùng đã đăng ký khuôn mặt.
4. **Luồng chính:**
  - a. Sinh viên đứng trước camera.
  - b. Hệ thống liên tục thu thập ảnh khuôn mặt gửi đến Backend AI.
  - c. Backend AI phát hiện, xử lý, trích xuất đặc trưng khuôn mặt, so sánh với dữ liệu đã đăng ký database.
  - d. Nếu nhận diện chính xác, hệ thống sẽ hiển thị thông tin sinh viên.
  - e. Hệ thống ghi nhận sinh viên vào/ra.
5. **Luồng phụ :**
  - **Lỗi nhận diện khuôn mặt:**

- Không phát hiện: AI không thấy khuôn mặt trong ảnh.
- Nhiều khuôn mặt: AI phát hiện nhiều hơn một khuôn mặt.
- Không khớp: AI thấy mặt nhưng không khớp với hồ sơ nào trong DB.

6. **Hậu điều kiện:** Thông tin nhận diện được ghi nhận, giao diện hiển thị trạng thái.

## **UC-02: Quản Lý Sinh Viên (Admin):**

### **1. Mục tiêu:**

Cung cấp công cụ cho Admin (quản trị viên) quản lý sinh viên ra/vào KTX thông qua xác thực bằng khuôn mặt:

- Đảm bảo chỉ những sinh viên đã đăng ký được phép vào/ra.
- Theo dõi lịch sử ra/vào của từng sinh viên.
- Dễ dàng quản lý và xử lý sự cố nhận diện.

### **2. Chức Năng Quản Lý Chi Tiết:**

#### **2.1 Đăng ký dữ liệu sinh viên (Thêm sinh viên)**

- **Admin nhập thông tin sinh viên:**
  - Mã sinh viên
  - Họ tên
  - Ngày sinh, giới tính
  - Lớp, ngành học
  - Số điện thoại, email
  - Ảnh chụp khuôn mặt
- **Lưu trữ dữ liệu:**
  - Ảnh gốc lưu trong cơ sở dữ liệu
  - Đặc trưng khuôn mặt (embedding) trích xuất lưu trữ để so sánh nhanh.

#### **2.2 Quản lý lịch sử ra/vào**

- Hệ thống ghi nhận mỗi lần nhận diện thành công:

- **Thời gian vào/ra**
- **Kết quả xác thực** (thành công/thất bại)
- Admin có thể:
  - Tra cứu lịch sử của từng sinh viên (lọc theo thời gian, trạng thái).
  - Xuất báo cáo thống kê số lần ra/vào theo ngày, tuần, tháng.

### 2.3 Quản lý dữ liệu khuôn mặt

- Cho phép cập nhật ảnh khuôn mặt khi sinh viên thay đổi ngoại hình (tóc, kính, phẫu thuật).
- Xóa dữ liệu khi sinh viên thôi KTX hoặc ra trường.

### 2.4 Giám sát thời gian thực

- Hiển thị trạng thái nhận diện trên giao diện dashboard:
  - Hình ảnh hiện tại từ camera.
  - Thông tin sinh viên vừa nhận diện.
  - Trạng thái (thành công/thất bại) với lý do cụ thể.
- Cho phép Admin can thiệp:
  - Đánh dấu nhận diện thủ công (nếu cần).
  - Ghi chú sự cố (ví dụ: camera lỗi, ánh sáng kém).

## 3. Luồng Chính – Nhận diện và ghi nhận

- Sinh viên nhận diện qua camera .
- Camera thu thập hình ảnh gửi về Backend AI liên tục (streaming hoặc ảnh chụp định kỳ).
- Backend AI xử lý:
  - Phát hiện khuôn mặt từ ảnh.
  - Trích xuất đặc trưng khuôn mặt.
  - So sánh với embedding trong database sinh viên đã đăng ký.
- Nếu khớp với một sinh viên:

- Hiển thị thông tin sinh viên (msv, tên, phòng, trạng thái).
- Ghi nhận thời gian vào/ra vào database lịch sử.
- Cập nhật dashboard thời gian thực.
- Nếu không khớp hoặc lỗi:
  - Hiển thị thông báo lỗi trên dashboard.

#### **4. Luồng Phụ – Xử lý lỗi nhận diện**

- Không phát hiện khuôn mặt:
  - Thông báo trên dashboard (Khuôn mặt không nhận diện được).
- Phát hiện nhiều khuôn mặt:
  - Thông báo trên dashboard.
- Không khớp với dữ liệu:
  - Thông báo "Khuôn mặt không nhận diện được".

#### **5. Hậu điều kiện:**

- Lưu lại lịch sử nhận diện vào hệ thống.
- Cập nhật giao diện hiển thị trạng thái (thành công/thất bại).
- Báo cáo đầy đủ cho Admin.