

**1. Employee Pictures (Hình ảnh sinh viên)**

* **Nguồn dữ liệu đầu vào**: Đây là nơi lưu trữ hình ảnh của sinh viên. Các hình ảnh này được tải lên và lưu trữ trong một bucket của Amazon S3 (được gọi là S3: Employee Pics).
* **Quá trình xử lý**: Những hình ảnh này được sử dụng để đăng ký khuôn mặt của sinh viên trong hệ thống để phục vụ cho quá trình nhận diện sau này.

**2. Visitor/Employee (Sinh viên hoặc Sinh viên)**

* **Nguồn dữ liệu đầu vào**: Đây là người sẽ được nhận diện khuôn mặt khi họ vào hệ thống.
* **Quá trình xử lý**: Hình ảnh của họ sẽ được tải lên từ ứng dụng frontend và so sánh với dữ liệu đã đăng ký trước đó.

**3. S3: Employee Pics (Bucket chứa hình ảnh sinh viên)**

* **Mục đích**: Đây là nơi lưu trữ các hình ảnh của sinh viên để đăng ký nhận diện khuôn mặt.

**4. S3: Visitor Pics (Bucket chứa hình ảnh sinh viên)**

* **Mục đích**: Đây là nơi lưu trữ các hình ảnh của sinh viên hoặc sinh viên khi họ đến. Hình ảnh này sẽ được tải lên khi người dùng tương tác với hệ thống thông qua ứng dụng frontend.

**5. React Frontend**

* **Mục đích**: Đây là giao diện người dùng (UI), nơi mà sinh viên hoặc sinh viên có thể tải lên hình ảnh của mình để hệ thống tiến hành nhận diện. Frontend này sẽ gửi yêu cầu đến API Gateway.

**6. API Gateway**

* **Mục đích**: Đây là một dịch vụ của AWS cho phép quản lý các API. Nó hoạt động như một "cổng" mà mọi yêu cầu từ frontend sẽ phải đi qua. API Gateway sẽ phân phối yêu cầu này tới Lambda function tương ứng để xử lý.

**7. Registration Lambda**

* **Mục đích**: Lambda function này được sử dụng để đăng ký hình ảnh của sinh viên. Nó sẽ nhận hình ảnh từ S3: Employee Pics và sau đó sử dụng dịch vụ Amazon Rekognition để phân tích và lưu trữ dữ liệu khuôn mặt trong DynamoDB.

**8. Auth Lambda**

* **Mục đích**: Lambda function này được sử dụng để xác thực hình ảnh của sinh viên hoặc sinh viên. Khi một hình ảnh được tải lên thông qua React Frontend, API Gateway sẽ gọi Auth Lambda. Auth Lambda sau đó sẽ kiểm tra hình ảnh này với dữ liệu đã lưu trữ trước đó trong DynamoDB thông qua Amazon Rekognition để xác định danh tính của người đó.

**9. Amazon Rekognition**

* **Mục đích**: Đây là dịch vụ của AWS chuyên về phân tích và nhận diện hình ảnh. Nó được sử dụng để so sánh và nhận diện khuôn mặt, cung cấp kết quả xác thực lại cho Lambda function.

**10. DynamoDB**

* **Mục đích**: Đây là cơ sở dữ liệu NoSQL của AWS, nơi lưu trữ thông tin về các khuôn mặt đã được đăng ký và các dữ liệu liên quan khác. Khi quá trình nhận diện diễn ra, dữ liệu sẽ được so sánh với các mục đã lưu trữ trong DynamoDB.

**Quy trình hoạt động tổng quát:**

1. **Đăng ký khuôn mặt sinh viên**:
   * Hình ảnh của sinh viên được tải lên S3: Employee Pics.
   * Registration Lambda lấy hình ảnh từ S3 và sử dụng Amazon Rekognition để phân tích và lưu trữ dữ liệu khuôn mặt trong DynamoDB.
2. **Nhận diện khuôn mặt**:
   * Sinh viên hoặc sinh viên tải lên hình ảnh của mình thông qua ứng dụng React Frontend.
   * Hình ảnh được gửi tới API Gateway, từ đó truyền tiếp tới Auth Lambda.
   * Auth Lambda sẽ sử dụng Amazon Rekognition để so sánh hình ảnh này với dữ liệu đã lưu trữ trong DynamoDB. Nếu khớp, hệ thống sẽ xác thực danh tính của người đó.

Các thành phần của hệ thống, bao gồm S3, Lambda với Trigger, DynamoDB, và Rekognition:

### ****1. Amazon S3 (Simple Storage Service)****

* **Lưu trữ hình ảnh**:
  + S3 là nơi lưu trữ tất cả các hình ảnh của sinh viên và sinh viên. Hình ảnh của sinh viên được lưu trữ trong một bucket riêng, và hình ảnh của sinh viên hoặc sinh viên mời (visitor) cũng được lưu trữ trong một bucket riêng khác.
  + Khi một sinh viên tham gia kỳ thi, ảnh của họ được lưu trữ trong S3 và được gán một tên tệp duy nhất, thường bao gồm ID của sinh viên.

### ****2. AWS Lambda và Trigger****

* **AWS Lambda** là dịch vụ tính toán không máy chủ cho phép bạn chạy mã mà không cần cung cấp hoặc quản lý máy chủ. Trong hệ thống của bạn, Lambda được sử dụng cho hai chức năng chính:
  1. **Registration Lambda**:
     + **Chức năng**: Đảm nhận việc lưu trữ thông tin ảnh của sinh viên hoặc sinh viên khi họ được đăng ký vào hệ thống. Lambda này được kích hoạt khi một hình ảnh mới được tải lên S3 (thông qua sự kiện **S3 Trigger**).
     + **Trigger**: Mỗi khi một hình ảnh mới được tải lên bucket S3, sự kiện này sẽ kích hoạt một Lambda function. Lambda sẽ nhận sự kiện này, lấy ảnh từ S3, và có thể thực hiện việc xử lý, chẳng hạn như lưu trữ thông tin vào DynamoDB hoặc gọi AWS Rekognition để phân tích khuôn mặt trong ảnh.
  2. **Auth Lambda**:
     + **Chức năng**: Xử lý các yêu cầu xác thực khi sinh viên hoặc sinh viên đăng nhập vào hệ thống hoặc tham gia kỳ thi. Lambda này có thể nhận một hình ảnh từ frontend (thông qua API Gateway) và sử dụng Rekognition để so sánh khuôn mặt trong ảnh với ảnh đã lưu trong S3.
     + **Trigger**: Lambda này được kích hoạt bởi một sự kiện từ API Gateway, khi có một yêu cầu đăng nhập hoặc tham gia từ frontend. Lambda sẽ lấy ảnh từ yêu cầu, so sánh với ảnh đã lưu trong S3 và trả về kết quả xác thực.

### ****3. DynamoDB****

* **DynamoDB** là dịch vụ cơ sở dữ liệu NoSQL do AWS cung cấp, được sử dụng để lưu trữ dữ liệu liên quan đến hệ thống, chẳng hạn như thông tin sinh viên, lịch thi của sinh viên, và kết quả của các lần nhận diện khuôn mặt.
  + **Lưu trữ thông tin đăng ký**: Khi một sinh viên hoặc sinh viên được đăng ký, thông tin của họ (bao gồm URL của ảnh trên S3) sẽ được lưu trữ trong DynamoDB. Mỗi mục trong DynamoDB thường chứa các trường như user\_id, image\_url, exam\_id, attendance\_status, v.v.
  + **Cập nhật trạng thái**: Sau khi Lambda thực hiện xong việc so sánh khuôn mặt, nếu nhận diện thành công, trạng thái điểm danh của sinh viên sẽ được cập nhật trong DynamoDB. Điều này cho phép hệ thống biết được sinh viên nào đã tham gia kỳ thi và ai chưa.

### ****4. AWS Rekognition****

* **Phân tích và nhận diện khuôn mặt**:
  + Khi một sinh viên hoặc sinh viên đăng nhập hoặc tham gia kỳ thi, ảnh của họ sẽ được gửi lên AWS Rekognition thông qua Lambda.
  + **Rekognition** sẽ thực hiện hai chức năng chính:
    1. **Phát hiện khuôn mặt**: Xác định và trích xuất các khuôn mặt từ ảnh nhóm hoặc ảnh cá nhân.
    2. **So sánh khuôn mặt**: So sánh khuôn mặt trích xuất với các khuôn mặt đã được lưu trữ trong S3. Nếu Rekognition tìm thấy một khuôn mặt khớp với mức độ tương đồng (similarity) cao, nó sẽ trả về kết quả xác nhận.

### ****5. Quy trình hoạt động của hệ thống trên code đã triển khai:****

1. **Tải ảnh lên S3**:
   * Người dùng (sinh viên, sinh viên) tải ảnh của họ lên thông qua frontend (React). Ảnh này sẽ được lưu vào S3. Hệ thống sử dụng các bucket riêng biệt cho ảnh sinh viên và sinh viên.
2. **Kích hoạt Lambda bằng S3 Trigger**:
   * Mỗi khi một ảnh mới được tải lên S3, sự kiện này sẽ kích hoạt một Lambda function. Lambda sẽ nhận thông tin sự kiện và có thể xử lý ảnh, chẳng hạn như gọi Rekognition để phân tích khuôn mặt.
3. **Lưu thông tin vào DynamoDB**:
   * Lambda sau khi xử lý sẽ lưu trữ thông tin của người dùng cùng với URL của ảnh đã lưu trữ vào DynamoDB. Điều này giúp hệ thống có thể truy xuất và sử dụng lại thông tin này khi cần.
4. **Nhận diện và so sánh khuôn mặt**:
   * Khi người dùng tham gia kỳ thi, ảnh của họ sẽ được gửi đến hệ thống. Lambda sẽ lấy ảnh này, gửi đến Rekognition để so sánh với các ảnh đã lưu trữ trong S3. Nếu tìm thấy khuôn mặt khớp, Lambda sẽ cập nhật trạng thái tham gia kỳ thi của sinh viên trong DynamoDB.
5. **Trả về kết quả**:
   * Sau khi so sánh thành công, kết quả sẽ được trả về cho người dùng qua API, bao gồm thông tin về các khuôn mặt đã khớp và danh sách sinh viên đã được điểm danh.