**Task 3: - Database: Table 1(id, name, embedding) Table 2(id, checking, date)**

**- CRUD operations**

**CReate, remove, update, delete**

**- tạo dữ liệu cho database( lấy ảnh của thành viên nhóm....)**

**- documents(làm những feature nào, .....)**

# DATABASE

A screenshot of a computer

Description automatically generated

## 1. Overview

This is an overview of the database structure used in the face recognition application. The database is designed to store information about individuals, including their unique identifiers, names, and facial embeddings. Facial embeddings are numerical vectors generated by a deep learning model that uniquely represent facial features.

## 2. Database Schema

|  |  |  |
| --- | --- | --- |
| **Column** | **Data Type** | **Description** |
| ID | INT (Primary Key) | Unique identifier for each record |
| Name | VARCHAR | The name of the invidual |
| Embedding | JSON | Numerical vector reresenting facial features |

Table Details:

* **ID**: Each individual is assigned a unique integer identifier. This serves as the primary key for efficient indexing and data retrieval.
* **Name**: A string representing the name of the individual. This is stored for identification purposes.
* **Embedding**: A multi-dimensional vector (array of floating-point numbers) that encodes the unique features of an individual's face. This is generated by a pre-trained deep learning model

## 3. Database Example:

The following is an example of how the data is stored:

|  |  |  |
| --- | --- | --- |
| **ID** | **Name** | **Embedding** |
| 1 | Vu Duc Thang |  |
| 2 | Dao Minh Quang |  |
| 3 | Tran Cong Thanh |  |
| 4 | Hoang Viet Tung |  |

## 4. Technical Considerations

### 4.1. Data Storage:

* Ensure embeddings are efficiently stored to minimize disk usage while ensuring fast access.
* Consider indexing the ID column for optimized lookup performance.

### 4.2. Embedding Precision:

* Use 32-bit floating-point numbers to balance precision and storage requirements.

### 4.3. Scalability:

* Ensure the database can handle a growing number of records as the user base expands.