Documentation du Code de Gestion d'Accès Étudiant avec Reconnaissance Faciale

# Introduction

Ce programme permet de gérer un système de contrôle d'accès pour des étudiants à l'aide d'un solde associé à leur ID et de reconnaissance faciale. Il utilise une interface graphique (GUI) avec Tkinter pour interagir avec l'utilisateur, et les bibliothèques OpenCV et face\_recognition pour capturer des images et identifier les étudiants via leur visage.

Le programme est structuré autour de plusieurs fonctions et d'une interface graphique permettant à l'utilisateur d'ajouter des étudiants, de vérifier leur identité par reconnaissance faciale, et de gérer leur solde.

# 1. Fonctions Principales

## 1.1 Ajouter un étudiant

**add\_student()**

Cette fonction permet d'ajouter un étudiant avec un ID et un solde initial. Elle procède comme suit :

* Récupère l'ID et le solde de l'étudiant depuis les champs de l'interface graphique.
* Vérifie que l'ID et le solde sont valides.
* Convertit le solde en nombre flottant et sauvegarde l'étudiant dans un fichier.
* Capture une image de l'étudiant via la caméra.
* Affiche un message d'information ou d'erreur en fonction de la validité des entrées.

**save\_student\_to\_file(student\_id, balance, file\_path="students.txt")**

Cette fonction ajoute un étudiant dans un fichier texte (students.txt), en ajoutant son ID et son solde à la fin du fichier sur une nouvelle ligne

## 1.2 Vérifier un Etudiant

**check\_access()**

Cette fonction gère le contrôle d'accès de l'étudiant. Elle fonctionne comme suit :

* Capture une image de l'étudiant via la caméra.
* Vérifie si l'image correspond à un étudiant enregistré dans la base de données d'images via la reconnaissance faciale.
* Si l'étudiant est reconnu, vérifie son solde et lui accorde l'accès s'il a suffisamment de fonds.
* Met à jour le solde de l'étudiant après l'accès et affiche un message de succès.

**verify\_student(image\_path, database\_dir="app/images/")**

Cette fonction vérifie si l'image capturée par l'utilisateur correspond à un étudiant existant en la comparant avec les images stockées dans le répertoire de la base de données. Elle utilise la bibliothèque face\_recognition pour comparer les encodages faciaux.

## 1.3 Autre

**get\_student\_balance(student\_id, file\_path="students.txt")**

Cette fonction permet de récupérer le solde d'un étudiant à partir du fichier texte. Elle lit le fichier ligne par ligne et renvoie le solde associé à l'ID de l'étudiant si celui-ci est trouvé.

**update\_student\_balance(student\_id, amount, file\_path="students.txt")**

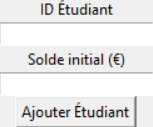
Cette fonction met à jour le solde d'un étudiant en ajoutant ou en soustrayant un montant spécifié. Elle modifie le fichier texte avec les nouvelles informations de solde.

**capture\_student\_image(student\_id, output\_dir="app/images/")**

Cette fonction capture une image de l'étudiant via la caméra et la sauvegarde dans un répertoire spécifié (app/images/) sous le nom de l'ID de l'étudiant. Elle attend que l'utilisateur appuie sur 's' pour sauvegarder l'image.

# 3. Fonctionnement du Programme

* Lorsqu'un utilisateur clique sur "Ajouter Étudiant", le programme capture l'ID et le solde de l'étudiant, enregistre l'étudiant dans un fichier et capture une image de l'étudiant via la webcam.



Ajouter un ID (Exemple : AURIACSimon)

Ajouter un solde (Exemple : 20)

Un fois le bouton cliquer, enregistrer la capture avec la touche s

* Lorsque l'utilisateur clique sur "Vérifier Accès", le programme capture une image via la webcam, compare l'image capturée avec celles enregistrées dans la base de données d'images, et vérifie si l'étudiant dispose d'un solde suffisant pour accéder.



# 4. Détails Techniques

## 4.1 Utilisation de OpenCV et face\_recognition

* **OpenCV** est utilisé pour capturer des images en temps réel à partir de la webcam.
* **face\_recognition** est utilisé pour comparer les visages des étudiants en fonction des encodages faciaux et déterminer si un étudiant est reconnu.
* **Tkinter** est utilisé pour créer les éléments de l'interface utilisateur (labels, champs de saisie, boutons).

## 4.2 Sauvegarde des Informations

* Les informations des étudiants (ID et solde) sont stockées dans un fichier texte students.txt.
* Les images des étudiants sont enregistrées dans le répertoire app/images/.

## 4.3. Gestion des Erreurs

Le programme inclut plusieurs vérifications pour éviter des erreurs courantes :

* Vérification de la validité des entrées pour l'ID et le solde.
* Gestion des erreurs liées à la caméra.
* Messages d'erreur si un étudiant n'est pas trouvé ou si son solde est insuffisant.