# Distributed Face Recognition System for Automated Attendance System at IIITH.

Team 19

# Table of Contents

# Team Members and Contact Information


| S. No | Name | Roll Number | Email ID |
|---|---|---|---|
| 1. | Keshavan Seshadri | 20171051 | keshavan.seshadri@research.iiit.ac.in |
| 2. | Ayush Deva | 201501098 | ayush.deva@students.iiit.ac.in |
| 3. | Shreya Vanga | 2019201087 | shreya.vanga@students.iiit.ac.in |
| 4. | Neerja gangwar | 2019201020 | neerja.gangwar@students.iiit.ac.in |
| 5. | Nikhil Jakhar | 20171186 | nikhil.jakhar@research.iiit.ac.in |
| 6. | Pranay Gupta | 20161088 | pranay.gupta@research.iiit.ac.in |


# Project Link

https://github.com/K7S3/Attendance-System-Using-Image-Recognition.git

# Project Overview

The current attendance system in IIIT is very inefficient. A person has to come up to every class and has to manually mark attendance, disrupting and taking up valuable teaching time. Through this project, we are proposing a new system that reduces manual intervention and makes the entire process efficient and unnoticeable. This can be done using some camera hardware installments combined with novel face recognition architectures, in a distributed fashion. This system is highly scalable and can be extended to be used in public places for security surveillance.

# Project Context and Tenets

## User Profile and Use Cases:

1. ### Admin level

   Here the admin has the access, read, write, manipulation, detection permissions of all the students record in the database.
   The admin has sole responsibility for creation of the professor's profile.

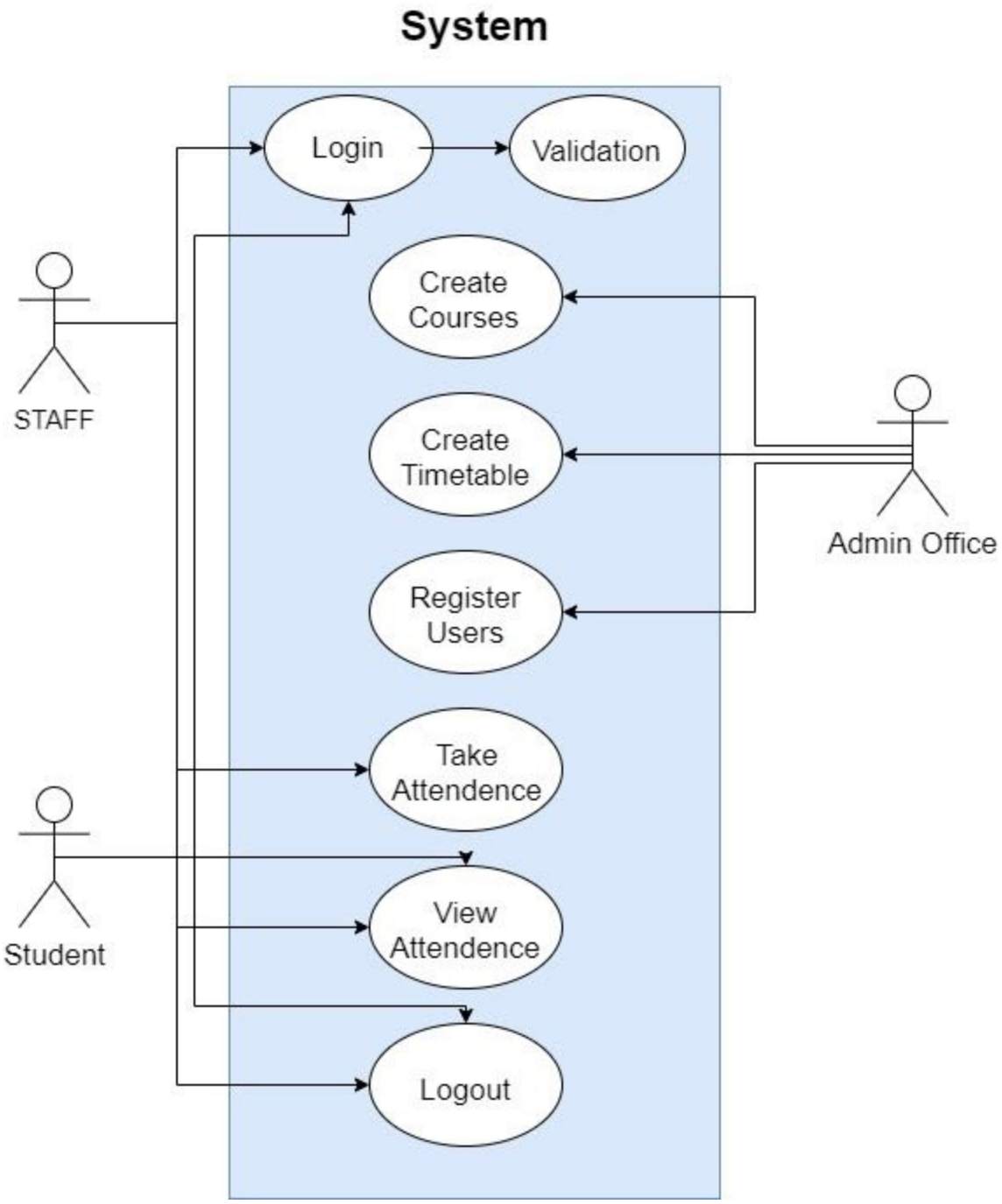
2. ### Faculty staff level

   The Teacher has the same set of permissions for students' databases.

3. ### Student level

   Student marks his/her attendance.
   The system will generate a report of the student's attendance.
   The student can view this report.

Use case Diagram



# Existing Solution and Problems

## Existing Solution

Technology used in the existing system is fingerprint sensor, In this system there were a lot of problems that were being faced in the class where strength of the students is not that small , and the number of the devices used to take the attendance are limited , so many times attendance mismatch occurred. Then it extended to wireless systems , it had its own limitations ,Then a system built which was RFID , the concept used to build this system was chip inside the Identification card provided to each student , But this system required a large number of components and space . Though this system was very accurate and chances of the error was very less , the expensiveness and maintenance of the equipment required for this system lead to failure of this system. After that System with speech recognition was taken. But these system were very complicated with a large amount of possibility of error.

## Problems with the Current System:

The problem with this approach in which attendance is taken manually is a very inconvenient task. Moreover, it is very difficult to verify one by one student in a large classroom whether the authenticated students are actually responding or not. This method could easily allow for impersonation. This approach also requires more man-power.

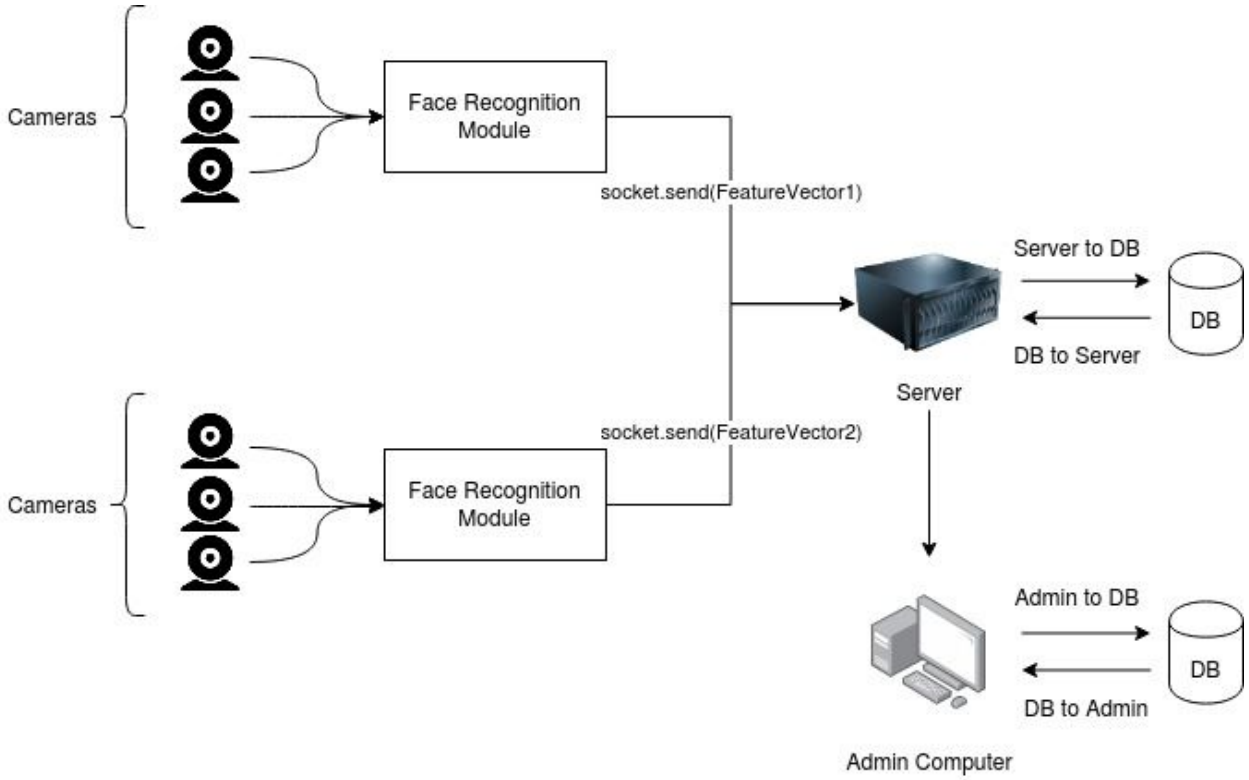
# Project Details

## Goals

The System consists of multiple cameras that capture images of the students for taking their attendance without disturbing the class. Which provides valuable attendance service for both teachers and students. System also records the entry and exit time of the student from the class

## Non-Goals

Sometimes poor lighting conditions in which the image is not clear which will result in degrading the accuracy of the system, we are not taking this into account.




# Tasks Proposed and Timeline

The task of the proposed system is to capture the face of each student and to store it in a database for their attendance. The face of the student needs to be captured in such a manner that all the features of the students' face needs to be detected. And through further processing of the images the face is recognized and we update the attendance database.
We will use Agile development methodology and ACID principles. We also plan to incorporate daily SCRUM so that all team members work everyday. The code will be documented when written.

## Task 1 : Digital Image Acquisition(1 day) (Apr 7)

- In the larger scheme of things we would require cameras to be installed on the entrances, but for the sake of the project we would using the feed incoming through the webcams
- These cameras record the feed and pass it on to the face recognition module.

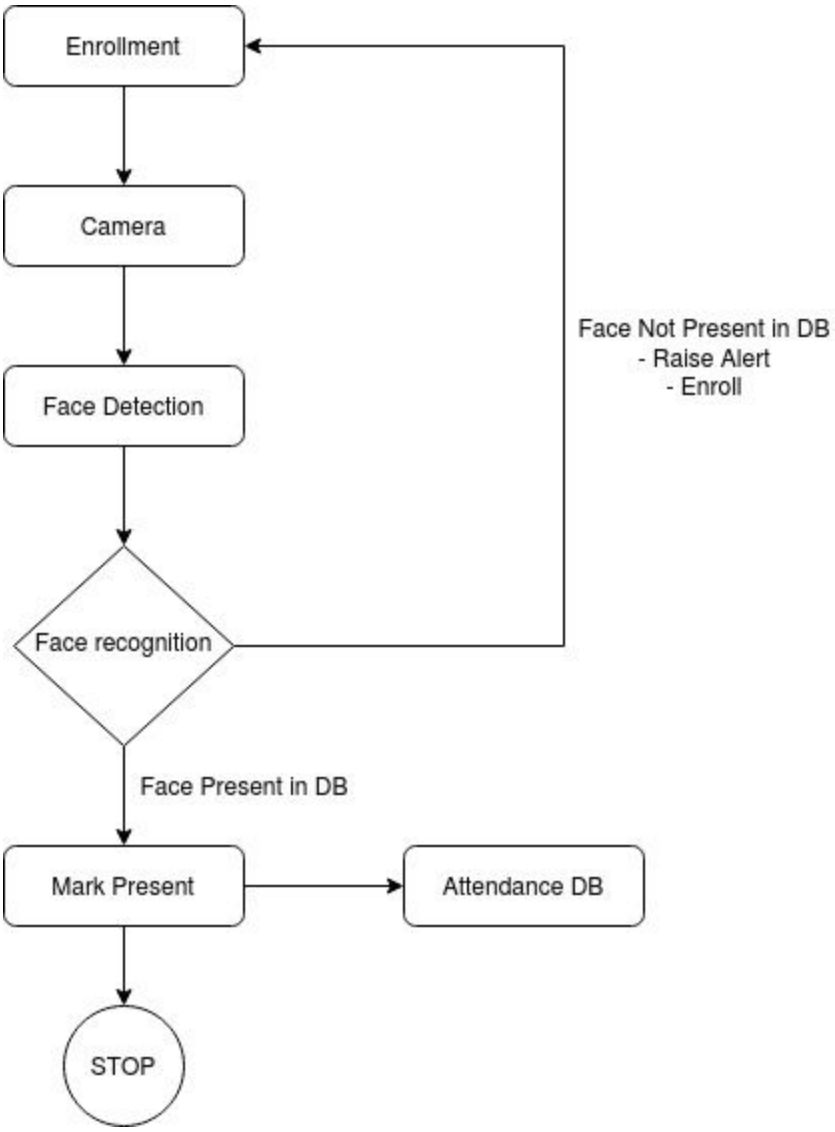
## Task 2 : Face Detection(4 days) (Apr 7 - Apr 11)

- The face detection algorithm now calculates the feature vector for every detected face, and passes it on to the central server.

- This task is done in a distributed fashion, because the multiple feeds are being captured simultaneously.
- We plan on using a pretrained facenet which is a state of the art and the most robust face detection algorithm, to get the final feature vector of the face.

## Task 3 : Enrollment and Marking(4 days) (Apr 11 - Apr 15)

- The face vector obtained in the last step is sent over to the main server, using socket programming, which contains the dictionary of the enrolled students, against the facial feature vectors.
- The face vector obtained on the central server is now compared with all the faces already present in the database, on finding a match, the candidate is marked present.
- Matching is done using the L2 distance between face vectors, and some threshold value.
- In case the face vector can not be matched with any of the vectors present in the database, the person now has to be enrolled for detection in future. The server replies accordingly with a present or a prompt for details required for enrollment.




## Task 6: Testing(4 days) (Apr 15 - Apr 19)

- We will test the model in a principled manner.
- Starting with Unit Testing
- Followed by Integration Testing
- Followed by System Testing
- Finally concluding it with acceptance testing

## Task 7: Results (2 days) (Apr 19 - Apr 20)

- We will show the results of how accurately facenet works, and also demos pertaining to both the possible scenarios.

# Challenges and Alternatives:

- Since we were hoping to deploy the system on IIIT network, but since that is not possible now, we will have to implement the socket programming section on our local machines itself.
- Sending over a large feature vector might take some time, so we will have to think of ways to reduce the dimensionality without having a loss of information(something like eigenfaces)
- Resources, the facenet is a deep model, so might end up taking some heavy computation power.
- Flask Socket IO will be used to send the feature vector to the server.
- Other Alternative is Django Channels

# FAQs

1. **Will the Project be Scalable and Maintainable?**
   The main goal of the project is to make it scalable that we can use the same code for a higher use case such as security surveillance.
2. **Is there documentation available?**
   The code is documented while writing and we will follow good naming practices for functions to make it easily comprehensible.
3. **What language is the code written?**
   The code will be written in python and will use deep learning frameworks like facenet for face recognition.
4. **What are the concepts involved?**
   Apart from all software engineering related concepts like Data Abstraction, OOP, ACID, and good design, the concepts of deep learning and distributed systems will be used.