

**CLASS ATTENDANCE FACE RECOGNITION**

**EUGENE MBOGO BSE-01-0008/2023**

**SAMUEL WANYINGE BSE-01-0009/2023**

**2024**

## Declaration

We hereby declare that this work is ours and has not been acquired from any other source. It is based on our originality and ideas. We have read and understood the university rules, and hence our work is free from any form of plagiarism.

Eugene Mbogo BSE-01-0008/2023

Signature………………………. Date………………………..

Samuel Kiganjo BSE-01-0009/2023

Sign: ………………………… Date: ……………………………

## Acknowledgment

We would like to firstly thank God for the opportunity to undertake this project in good health and life. We would also like to extend my most sincere gratitude to our Supervisor, Mr. Richard Mathenge for his consistent academic support of the project.

MR. Richard Mathenge

Sign: ………………………… Date: ……………………………

## Abstract

The implementation of a Facial Recognition System can aid in identifying or verifying a person's identity from a digital image. Accurate attendance records are vital to classroom evaluation. However, manual attendance tracking can result in errors, missed students, or duplicate entries. The adoption of the Face Recognition-based attendance system could help eliminate these shortcomings. This innovative approach involves utilizing a camera to capture input images, detecting faces using algorithms such as Haar cascade, Eigen values, support vector machines, or the Fisher face algorithm, verifying the faces against a database of student profiles, and marking attendance in a real time database(firebase). The use of OpenCV, an open-source computer vision library, ensures the efficient functioning of the system. The proposed model involves training the system with the authorized students' faces to create a database. The Face Recognition-based attendance system could help automate attendance records with high accuracy and reduce the burden of manual attendance tracking.

Table of Contents

[Declaration 2](#_Toc188632148)

[Acknowledgment 3](#_Toc188632149)

[Abstract 4](#_Toc188632150)

[CHAPTER 1 9](#_Toc188632151)

[INTRODUCTION 9](#_Toc188632152)

[1.1 Introduction 10](#_Toc188632153)

[1.3 Statement of the problem 11](#_Toc188632154)

[1.4 Implemented Solution 11](#_Toc188632155)

[1.5 Objectives 12](#_Toc188632156)

[1.6 Research Questions/Hypothesis 12](#_Toc188632157)

[1.7 Justification 13](#_Toc188632158)

[1.9 Scope of The Study 13](#_Toc188632159)

[CHAPTER 2 15](#_Toc188632160)

[LITERATURE REVIEW 15](#_Toc188632161)

[Introduction 15](#_Toc188632162)

[2.2 Reviewed Similar Systems 15](#_Toc188632163)

[2.3 Tools and Methodologies used in Reviewed Systems 19](#_Toc188632164)

[2.4. Gaps in the existing systems and the proposed solution 19](#_Toc188632165)

[2.5 The proposed solution 19](#_Toc188632166)

[Chapter 3 21](#_Toc188632167)

[CHAPTER THREE: METHODOLOGY 21](#_Toc188632168)

[3.1 Methodology and tools 21](#_Toc188632169)

[3.2 21](#_Toc188632170)

[Initial Requirements 22](#_Toc188632171)

[Design 23](#_Toc188632172)

[Prototyping 24](#_Toc188632173)

[Evaluation 24](#_Toc188632174)

[Reviewing 24](#_Toc188632175)

[Development 24](#_Toc188632176)

[Test 24](#_Toc188632177)

[Maintain 24](#_Toc188632178)

[3.2 Source of data 25](#_Toc188632179)

[3.3 Data Collection Methods 25](#_Toc188632180)

[Chapter 4 28](#_Toc188632181)

[CHAPTER FOUR: SYSTEM ANALYSIS AND DESIGN 28](#_Toc188632182)

[4.5. System analysis tools 29](#_Toc188632183)

[4.6 System investigation 30](#_Toc188632184)

[4.6.1 Introduction 30](#_Toc188632185)

[4.6.2 Data collection 31](#_Toc188632186)

[Introduction to Fact recording methods: 31](#_Toc188632187)

[4.7 System analysis 31](#_Toc188632188)

[CHAPTER 5: SYSTEM DESIGN 33](#_Toc188632189)

[5.1 Introduction 33](#_Toc188632190)

[5.2 Objectives of System Design 34](#_Toc188632191)

[5.3 System Design Tools 35](#_Toc188632192)

[5.3.1 Flow chart 35](#_Toc188632193)

[5.3.2 Data flow diagram 36](#_Toc188632194)

[5.3.3 Use case diagram 38](#_Toc188632195)

[5.4.1 Entity Life History 38](#_Toc188632196)

[5.5 Physical Design 41](#_Toc188632197)

[5.5.1 Data Dictionary 41](#_Toc188632198)

[5.5.2 File/Database Design 42](#_Toc188632199)

[5.5.3 Code Design 45](#_Toc188632200)

[5.5.6 Process / Program Design / UML 48](#_Toc188632201)

[CHAPTER SIX: SYSTEM IMPLEMENTATION 51](#_Toc188632202)

[6.1 Coding/Environment/Debugging/Techniques 51](#_Toc188632203)

[6.2 Program Listing 52](#_Toc188632204)

[6.3 Test System/Program Testing 52](#_Toc188632205)

[6.4 Test Plan 52](#_Toc188632206)

[6.5 Test Data 52](#_Toc188632207)

[CHAPTER SEVEN: USER MANUAL - DOCUMENTATION 53](#_Toc188632208)

[7.1 Installation environment 53](#_Toc188632209)

[7.2 Installation Requirements 53](#_Toc188632210)

[7.4 User instructions 53](#_Toc188632211)

[CHAPTER EIGHT: LIMITATIONS, CHALLENGES, CONCLUSIONS, AND RECOMMENDATIONS 55](#_Toc188632212)

[8.1 Limitation 55](#_Toc188632213)

[8.2 Challenges 55](#_Toc188632214)

[8.3 Degree of Success 55](#_Toc188632215)

[8.4 Learning experience 55](#_Toc188632216)

LIST OF ABRIVIATIONS.

AI- Artificial Intelligence

CV- Computer Vision

RFID- Radio Frequency Identification

HOG- Histogram of Oriented Gradients

ML - Machine Learning

## CHAPTER 1

### INTRODUCTION

#### Background of study

Computer Vision (CV) is a field of computer science focused on replicating aspects of the human visual system's complexity, enabling machines to process and interpret objects in images and videos in a manner similar to humans. Since its inception in the 1950s, where it was used to distinguish between typed and handwritten text, CV has evolved significantly. Its growth is largely driven by the increasing availability of vast amounts of visual data and advancements in computing power. Modern algorithms and hardware have enabled CV systems to achieve remarkable accuracy, with some systems reaching 99% in object identification (2024, February 9). Training a computer to visualize data typically involves feeding it large volumes of images and allowing it to identify patterns among relatable elements.

CV applications span numerous industries, including self-driving cars, augmented and mixed reality, facial recognition, and healthcare. For instance, in healthcare, CV is used to identify symptoms in MRI scans, demonstrating its potential to improve diagnostics and patient outcomes ("7 applications of computer vision," 2022).

In educational institutions in Kenya, maintaining a minimum attendance rate of 75% is often mandatory for students to qualify for examinations. However, students have devised various methods to bypass this requirement. One of the most prevalent practices is the use of proxy attendance, where students have classmates sign the attendance sheet on their behalf. Additionally, some students take advantage of opportunities to pre-sign for classes they plan to miss. These practices undermine the integrity of attendance tracking and contribute to a gradual decline in academic performance over time. Proxy attendance not only reduces the effectiveness of classroom engagement but also compromises the ability of educators to accurately assess students' commitment and participation.

The issue of proxy attendance is not unique to Kenya; it is a global problem affecting educational systems in various countries. Studies have shown that poor class attendance is closely linked to lower academic performance and higher dropout rates. Institutions relying on manual attendance systems are particularly vulnerable to this malpractice, highlighting the need for innovative solutions.

By leveraging computer vision technologies, it is possible to create automated systems capable of accurately tracking student attendance through facial recognition. This approach eliminates the possibility of proxies and ensures the reliability of attendance data. Implementing such technologies could significantly enhance the quality of education, promote accountability among students, and improve overall academic outcomes.

### Introduction

Face recognition is whereby a system can detect a face in a video or image and run an algorithm to recognize who's face it is. This biometric technology relies primarily on databases whereby the image is compared to the already existing images on the database. This technology has been widely used in the tech industry since most phones can be unlocked using the user's face (Smartphones with the face unlock feature). The security sector also uses it to reduce crime and prevent violence. Although the technology is not yet at its peak, advancements in AI will enable this technology to be more accurate in the coming future.

Most organizations use biometrics such as fingerprints, RFID card systems, and voice recognition. Other biometrics have their flaws; they are not 100% accurate. The conventional system of calling students' names is accurate but time-consuming. In the RFID card system, each student assigns a card with their corresponding identity, but there is a chance that the student may misplace the card or an unauthorized person can use the card for fake attendance ("Disadvantages of RFID. Mostly minor or you can minimize them," 2021). The use of face recognition for student attendance marking is intelligent for attendance management since it is more accurate, faster, and reduces the chances of proxy attendance.

Our research seeks to use facial recognition to identify and record students' academic year attendance. This project will record the time a student arrives in class and how often the student attends the lecture during the semester. The student's images will be stored within a database. The school will need to install CCTV cameras to get the live video when students get into the lecture room. Images from the live recording will be compared to the images on the database to get real-time marking of the class attendance.

### 1.3 Statement of the problem

The manual attendance system currently employed in many educational institutions is time-consuming, prone to errors, and inefficient. This process involves teachers physically calling out student names and recording their attendance manually, leading to potential inaccuracies and administrative overhead. Moreover, traditional methods can be susceptible to fraud and abuse, such as students marking attendance on behalf of others or falsifying records.

These challenges necessitate a more efficient, accurate, and secure method for managing class attendance. The proposed facial recognition-based system aims to address these shortcomings by automating the attendance process, reducing the likelihood of errors, and enhancing the overall efficiency of educational institutions.

### 1.4 Implemented Solution

This project addresses the issue of absenteeism and the use of proxies in universities by eliminating traditional attendance marking methods, such as calling out names or signing attendance sheets, which are cumbersome and prone to distractions. The proposed system provides an automated solution by capturing students' faces as they enter the classroom and comparing them with an existing database for automatic attendance logging.

The system ensures reliable facial recognition even under varying conditions, such as changes in lighting, angles, or minor facial obstructions, by leveraging advanced algorithms and robust image preprocessing techniques. During examinations, the system compares live images of students with stored records, ensuring that only authorized individuals sit for exams. This approach eliminates proxy attendance, enhances accuracy, and provides a faster, more reliable alternative to manual attendance methods.

### 1.5 Objectives

#### 1.5.1 General Objectives

The main objective of the proposed system is to create a system that records class attendance using facial recognition.

#### 1.5.2 Specific Objectives

1. Design a database to securely store student facial data and attendance records, enabling seamless comparison and retrieval

2. Implement advanced facial recognition algorithms to ensure reliable

identification under varying conditions, such as changes in lighting, angles, or minor facial obstructions.

3. To create a seamless integration between a mobile application/web application and a cloud database(firebase) for efficient data collection, training, and real-time attendance management.

### 1.6 Research Questions/Hypothesis

1. How can high-quality facial image data be collected from students in a variety of lighting conditions, poses, and expressions?
2. How can data privacy and ethical considerations be addressed during data collection and storage?
3. How can the app ensure user-friendliness, accessibility, and seamless integration with the facial recognition system?

### 1.7 Justification

The school will be able to get the attendance of students autonomously, therefore, reducing cases of truancy. The system will be reliable and save time for signing attendance. Students must attend most of their classes in person to be signed in by the system. The system will also reduce the chances of other students doing exams on each other's behalf.

We believe in the saying you get what you pay for; in this instance, we will reflect on the parent who has sent his/her child to school to get a better education. The parent sacrifices the little resource or finance they have to give their child a better education, but the child comes to campus, gets involved in the wrong crowd, and goes downhill. The result would be missing classes, cheating in exams, getting multiple supplementary examinations, etc. The parent did everything to give the child a good education but did not get the desired result. In theory, our system will be able to make that a thing of the past. With the system being able to provide a detailed graph of students' attendance during the semester, parents will know what caused the poor performance of their children.

### 1.9 Scope of The Study

#### 1.9.1 In Scope

1. Students will upload their images through a user-friendly mobile or web-based application.
2. Student images will be securely stored in the Firebase storage library, enabling easy access and processing by Python-based algorithms.
3. Class attendance will be recorded in real-time as students enter the classroom, ensuring an accurate and up-to-date attendance log.

#### 1.9.Out Scope

1. Functions such as course management, scheduling, or general academic performance tracking are outside the scope of this project.
2. The system requires an active internet connection to interact with Firebase and will not support offline attendance tracking or storage.
3. Dataset will only be limited to a few students during the project.

## CHAPTER 2

## LITERATURE REVIEW

### Introduction

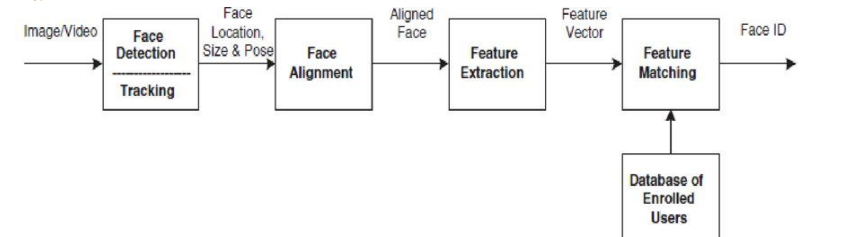
In the face detection and recognition system, the process flow is initiated by being able to detect the facial features from camera or a picture store in a memory. The algorithm processes the image captured and identifies the number of faces in the image by analyzing from the learned pattern and compare them to filter out the rest. This image processing uses multiple algorithm that takes facial features and compare them with known database.

The motivation behind this project is to simplify the means by which attendance is taken during lectures and how much time it takes. The use of ID cards or manually calling out attendance and writing it down on sheets is not productive and efficient. This system will detect the number of faces on the class and will also identify them from the store database. With the face detection and recognition system in place, it will be easy to tell if a student is actually present in the classroom.

### 2.2 Reviewed Similar Systems

#### 2.2.1 Haar Cascades facial recognition

Haar Cascade classifier is a real-time face detector library that identifies objects in an image or video (Mittal, 2020). It is mainly used in facial recognition systems since it can recognize human faces and train a program to recognize them. Below is a diagram of a system created by the use of Haar cascades.



***Figure 1***

Face detection: This process detects and separates the facial area from the background image

Face alignment: This process focus on finding the best localization and normalization of the face; where the detection step roughly estimates the position of the face, this step outlines the facial components, such as face outline, eyes, nose, ears, and mouth.

Feature extraction: Feature extraction results in adequate valuable information for distinguishing between the faces of different persons and stability concerning the geometrical and photometrical variations.

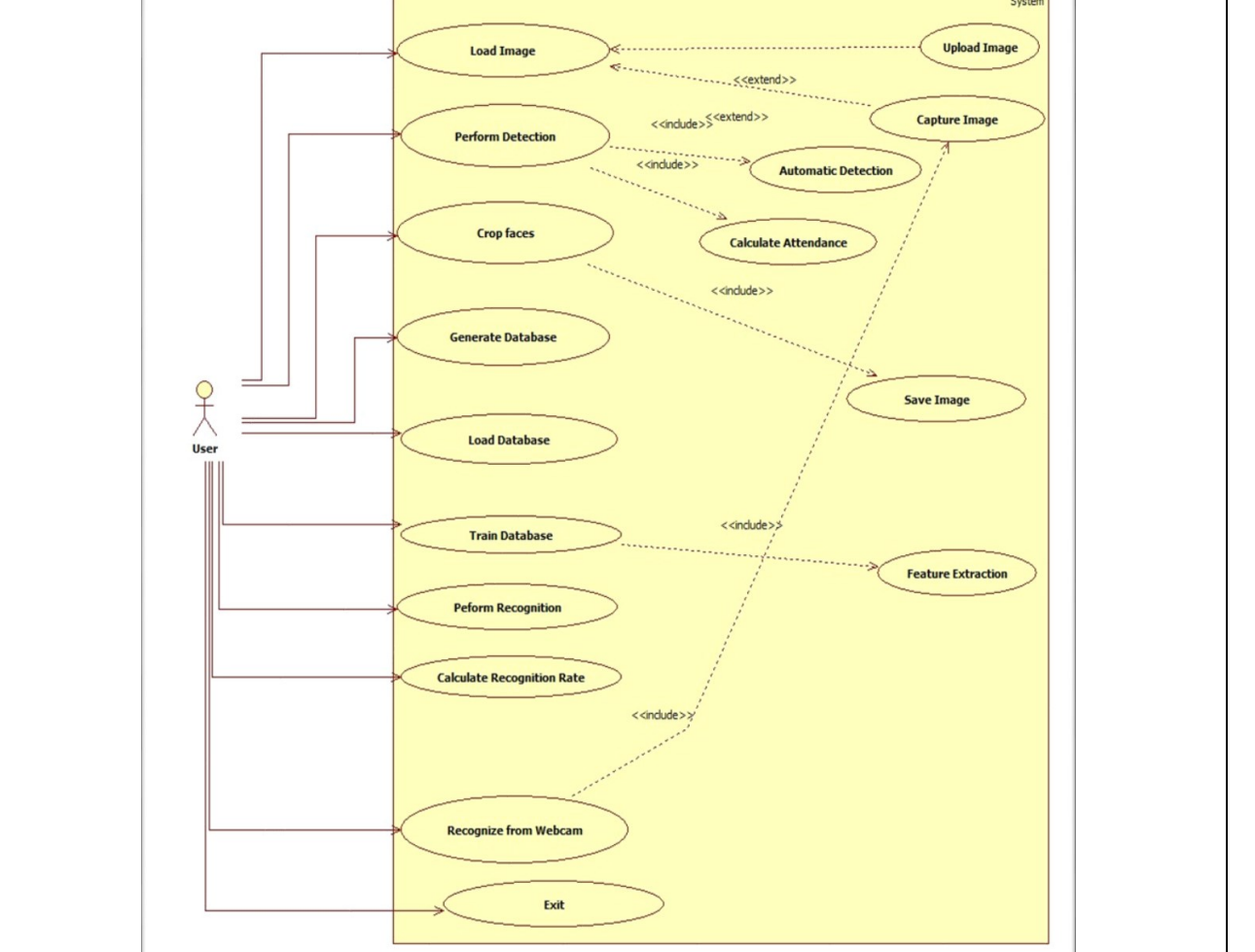
Face matching: The extracted features are compared to those stored in the database, and decisions are made according to sufficient confidence in the match score.

Applications include;

* Facial Recognition- It has been used in mobile phone security primarily to determine the validity of users secure login.
* Autonomous vehicles- Autonomous vehicles require knowledge about their surroundings, and Haar cascades can help identify objects, such as pedestrians, traffic lights, and sidewalks, to produce more informed decisions and increase safety.
* Image Search and Object Recognition: Expanding off facial recognition, any variety of objects can be searched for by using a computer vision algorithm, such as Haar cascades

#### 2.2.2 previous class attendance based on face recognition

This is a project done by students as a final year project at Kingston University London in 2018. The system will be presented an image either via camera or from memory and it must detect the number of faces on it automatically. After identifying faces, the system should crop the faces from the image and store them in memory for image recognition which will be done in the second step. The system should be able to automatically count the number of faces detected on the image. The second step will be the recognition part where the system will be able to match faces from the stored dataset and compare it to the input data from the first step. A software will be used for this system which automatically sorts out the faces. The software will be inter-active so to facilitate interaction between multiple tasks as required. Because the system has two steps, the second phase of the system will involve the training of images on a dataset that are to be used for recognition.The system behavior has been explained in the following flowchart



Technology Used The key algorithms are Viola-Jones for face detection and Hidden Markov Model with SVD. • The implementation of The Viola-Jones algorithm is available on softwares like MATLAB, OpenCV and Web Browsers (using adobe flash). • The existing implementation of the Hidden Markov Model with SVD for face recognition are available on MATLAB, C++ and OpenCV libraries.

#### Biometric Systems

A Biometric system is reliable because it cannot be stolen, borrowed, bought, or forgotten like a password or ID. Biometric systems include fingerprint, face, voice, and iris. The fingerprint is a physical biometric aspect. It is used to identify a person’s identity due to its uniqueness, where no two persons can share the same fingerprint. Although fingerprint recognition has been deployed for a decade, it has become one of the most common biometrics nowadays. The fingerprint identification system is a cheap but solid mechanism simultaneously.

Moreover, it is a simple way to identify humans speedily and accurately. Fingerprint systems have been used widely to sign class attendance (INCUBSENCE.com, 2020). This system is accurate, effective, and fast, but some of the drawbacks include the following:

* Delay- Biometric attendance devices take more time to mark attendance and cause a long queue of workers in the morning and the evening.
* Infection Carrier- Coronavirus spreads from human touch. If an employee touches the biometric attendance system with coronavirus and touches her surface, she is susceptible to getting infected with the coronavirus.
* Physical Challenges- Many employees have physical challenges due to lost or damaged physical parts. Thus it is challenging to get them involved in the enrollment process.
* Environmental Challenges-Under extreme cold or extreme heat, the error rate is also very high. Thus this poses a challenge for using a biometric attendance system.

#### RFID Systems

The RFID system was primarily used in educational institutions and organizations like the medical department. The RFID cards can sometimes face interference when another card is within the same field. Students will require to undergo training on how to use the cards.



***Figure 2***

#### Signing Attendance

Signing an attendance sheet is a traditional method whereby students sign the sheets with their signatures. This system is unreliable since it is time-consuming and highly inaccurate since students can use proxies to sign on their behalf.

### 2.3 Tools and Methodologies used in Reviewed Systems

The RFID system uses cards and the system to scan the cards. In signing the sheet, students are only required to put their signatures to affirm that they have attended the class. Biometric scanners will also be used to scan for student biometrics.

### 2.4. Gaps in the existing systems and the proposed solution

The main flaw with the Haar cascade system is that it cannot detect faces from a different orientation. The system only recognizes the face from a frontal view. The library will use the frontal face default XML file to train the system; thus, orientation will only be limited to the frontal view.

The traditional way of attendance marking consumes much time, either through name-calling or passing the sign sheet, distracting students' attention.

RFID cards are more accurate than other attendance methods, but students can use the card on behalf of other students. RFID systems are costlier and require more costly equipment to be maintained regularly. This system also requires the card users to undergo training, and in case the card is misplaced and found in the wrong hand, users can face severe problems if the card is used for malicious activities.

### 2.5 The proposed solution

The system will have multiple images of the same student, and images will be used to train the system for high accuracy. The system will also save time as the attendance will be automated and would not affect the student lecture in any way possible. The system is more secure compared to other systems that were used previously. Face detection is also more accurate, fast, and efficient than other systems such as the RFID, voice recognition, and signing attendance sheets.

## Chapter 3

## CHAPTER THREE: METHODOLOGY

This project requires a lot of data to have a high degree of accuracy in detecting students' faces. The methodology we will use is prototyping since the system will require the constant addition of data and feedback from the students. This methodology will follow specific criteria: Image acquisition, dataset creation, dataset storage, image recognition, and lastly, signing of an cloud database.

### Methodology and tools

Prototyping Methodology

Initial

Requirement

Design

Eva

luation

Development

Test

Maintain

Prototyping

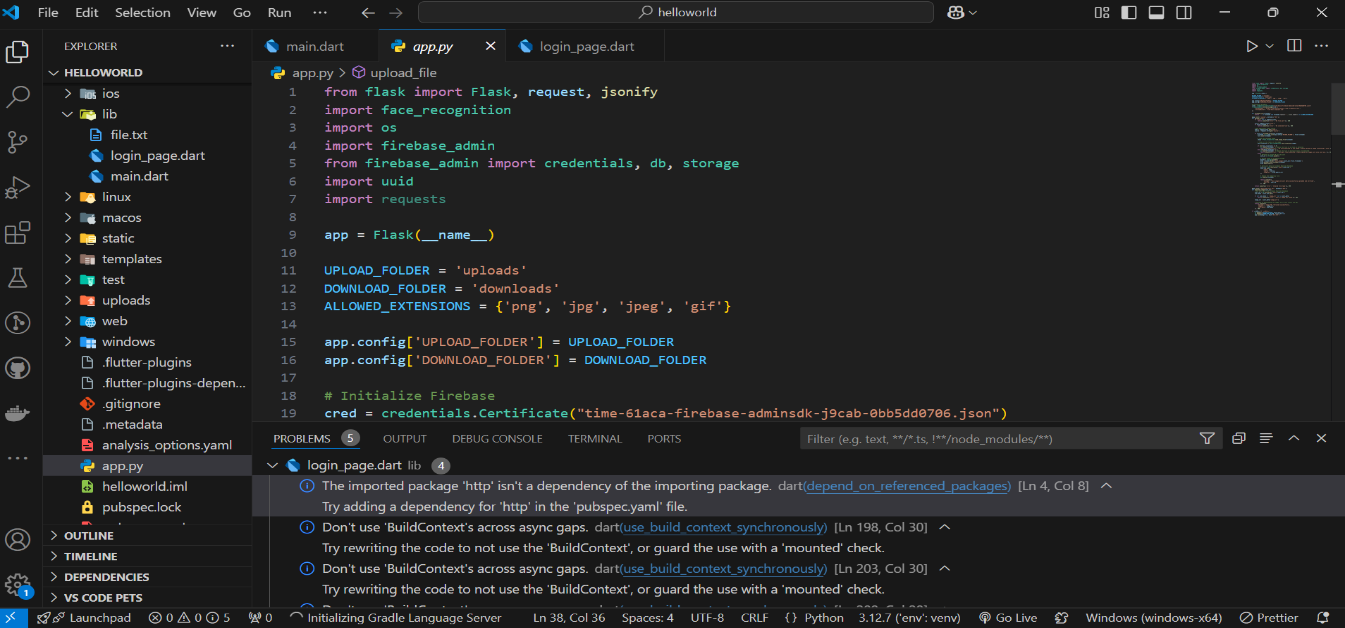
Reviewing

### 

*Figure 3*

### Initial Requirements

Students will first and foremost be required to upload their images. For this reason, we have created a mobile application/web application to allow students to upload several facial images to the Firebase storage. This application will record the student's images and course. After the images are uploaded to the Firebase, we will use python to sort the images and categorize them. Then we will use the images to train the system to identify images and achieve accuracy in identifying the student images. The system will require a high-quality webcam to record video as live input to the system.

****

*Figure 4*

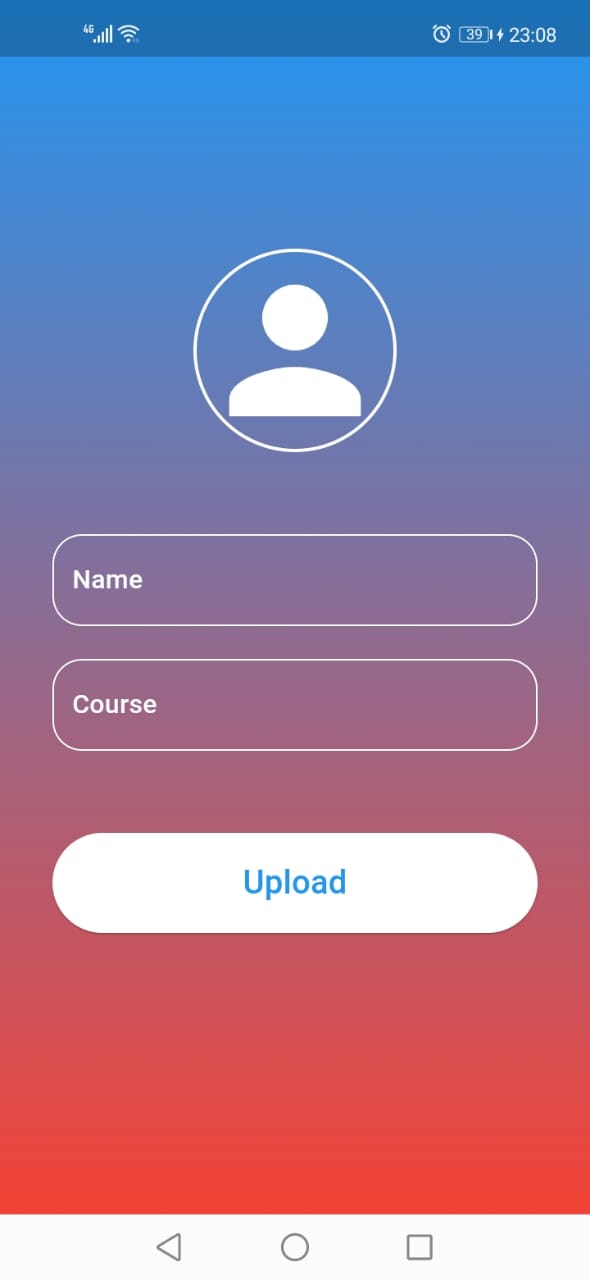
******

Figure 5

### Design

Since we will require feedback and testing, we will require the help of some of our classmates. They will need our mobile application to upload their images to the Firebase. These images will be stored in a dataset and be used to train the facial recognition system. we will use the face recognition library from python feed in the student images and see what percentage of accuracy the system has in identifying the students.

### Prototyping

Recognition accuracy can be explained as to how the system will get the correct detail of the student once the student has attended the class. The accuracy level of the system will depend on the images of students uploaded. we will use evolutionary prototyping in order to get the highest percentage of recognition accuracy.

### Evaluation

Evaluation will entail focusing on the strength and weaknesses of the system. we will focus mainly on recognition accuracy. The higher the level of accuracy, the more the system is likely to identify a student without any errors.

### Reviewing

It will need to be reviewed if the system has an accuracy level lower than 80%. This will entail adding more dataset images or slightly adjusting the python code to accommodate changes such as face orientation, wearing things like spectacles, or lighting within a classroom.

### Development

If the system attains an accuracy of above 80% in recognizing student images, we can move to the development face.

### Test

The testing phase will include adding more students to the database and seeing how the system will progress.

### Maintain

This will include regular updates to the system in terms of updating images and tweaking the code from time to time to maintain a high level of accuracy.

### 3.2 Source of data

The primary data source will include the details obtained from the mobile application/web application. If the student cannot upload the image, we can upload it to the Firebase. The input of the live video capturing will be the webcam.

### 3.3 Data Collection Methods

#### 3.3.1 Observation

We have observed that students who use the traditional form of signing class attendance and how easily it is for them to sign as proxies and how distracted they are when they are signing the attendance

#### 3.3.2 Questionnaire

This data collection technique involves developing a list of question related to the research topic and giving respondents to fill them according to their opinion. To accomplish this, the questionnaire will develop a google form and send it to our target group since some guidance is needed. It will also add some instruction on how to go about it.

#### 3.3.3 Literature Review

The literature review will be our project's primary gathering of background information. Here data will be gathered by going through existing systems to derive data that has already been collected to develop and implement the facial recognition system.

#### 3.4 Resources required / materials Hardware and Software specification

**Operating System:**

1. Windows: Windows 10 or later, preferably with a 64-bit processor.
2. Linux: Ubuntu 18.04 or later, or other popular Linux distributions.

**Programming Languages and Libraries:**

1. Python: A popular choice for machine learning and computer vision tasks, with libraries like OpenCV, TensorFlow, Flask,Numpy, Dlib and Keras.
2. C++: Often used for performance-critical applications, with libraries like cmake.
3. Java: Can be used for developing cross-platform applications, with libraries like DeepLearning4J.

**Development Environment:**

1. Integrated Development Environment (IDE): Visual Studio Code, PyCharm, IntelliJ IDEA, or Eclipse.
2. Version Control: Git for managing code changes and collaboration.

**Machine learning frameworks**

1. TensorFlow: A popular open-source platform for machine learning, including deep learning.
2. Keras: A high-level API that runs on top of TensorFlow or Theano, simplifying model building.
3. OpenCV: A comprehensive computer vision library with modules for image processing, object detection, and facial recognition.
4. dlib is a C++ toolkit that provides a variety of algorithms for machine learning and computer vision tasks
5. CMake is a cross-platform build system generator that automates the process of building executable programs and libraries from source code. It is designed to be independent of any particular compiler or operating system, making it a versatile tool for managing complex projects.

**Database:**

1. Real-time Database(firebase): A traditional cloud-hosted NoSQL database that provides real-time updates to connected clients.
2. Cloud Storage: A secure and scalable cloud storage solution for storing user-generated content.
3. Cloud Firestore: A flexible, scalable NoSQL cloud database that allows you to store and retrieve data in real-time.

**Cloud Services (optional):**

1. Firebase: A Google platform that provides backend services like cloud storage, authentication, and real-time database, which can be useful for storing and managing user data.

MySQL or PostgreSQL: Relational databases for larger-scale applications

## Chapter 4

## CHAPTER FOUR: SYSTEM ANALYSIS AND DESIGN

4.1 Introduction

In this chapter, the developers will analyze the survey before the system was made and after the development of the system and what impact it has brought to the school. The designers will further explain the cross-cutting issues patterning to the design and the strategies used for implementation.

**4.2SystemDevelopmentMethodology**  
The system adopts an agile development methodology, emphasizing iterative development, continuous feedback, and collaboration with students. Agile is chosen for its flexibility in accommodating changes and delivering incremental improvements throughout the development lifecycle.

4.3 Feasibility Study

**4.3.1 Technical Feasibility**

The system's technical requirement is economical and does not use any other additional Hardware and software. The technical evaluation must also assess whether the existing systems can be upgraded to use the new technology and whether the organization has the expertise to use it

4.3.2 **Economic Feasibility**:

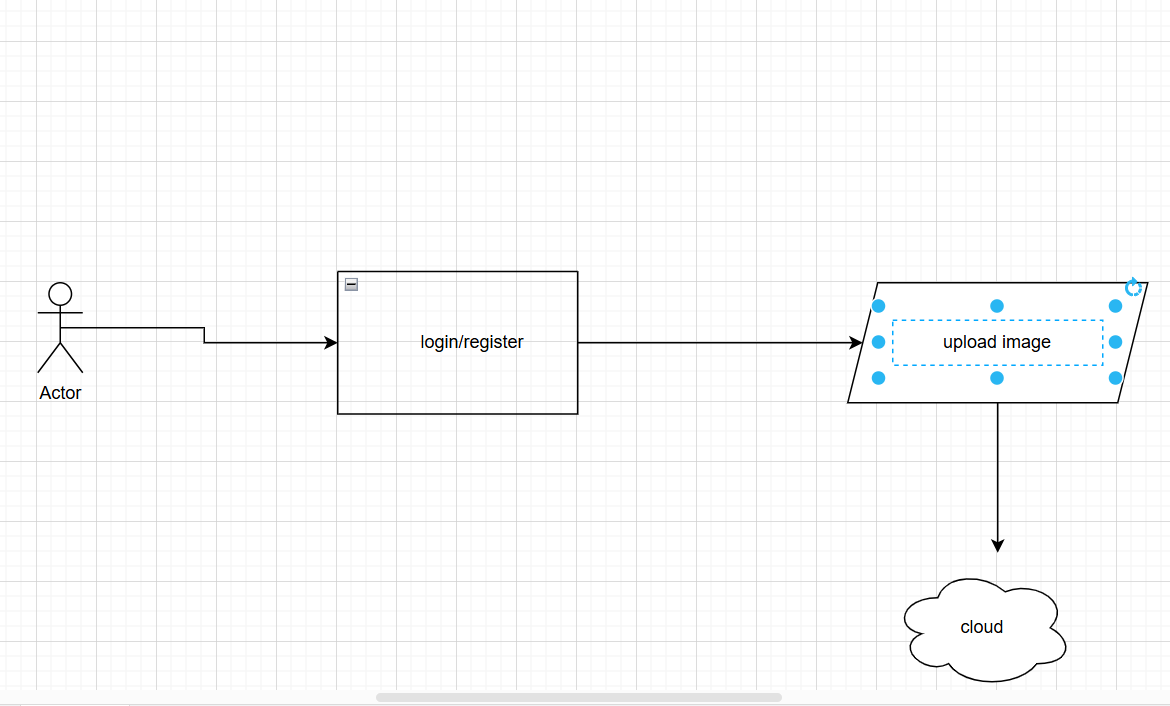
The system leverages open-source tools and cloud services with affordable pricing plans, reducing development and deployment costs.

4.3.3 **Operational Feasibility**:

The system is relatively easy to use and learn due to its simple but attractive interface. The user requires no special training to operate the system. Acceptance revolves around the current system and its personnel. Technical performance includes issues such as determining whether the system can provide the correct information for the Department personnel student details and whether the system can be organized so that it always delivers this information at the right place and on time using intranet services.

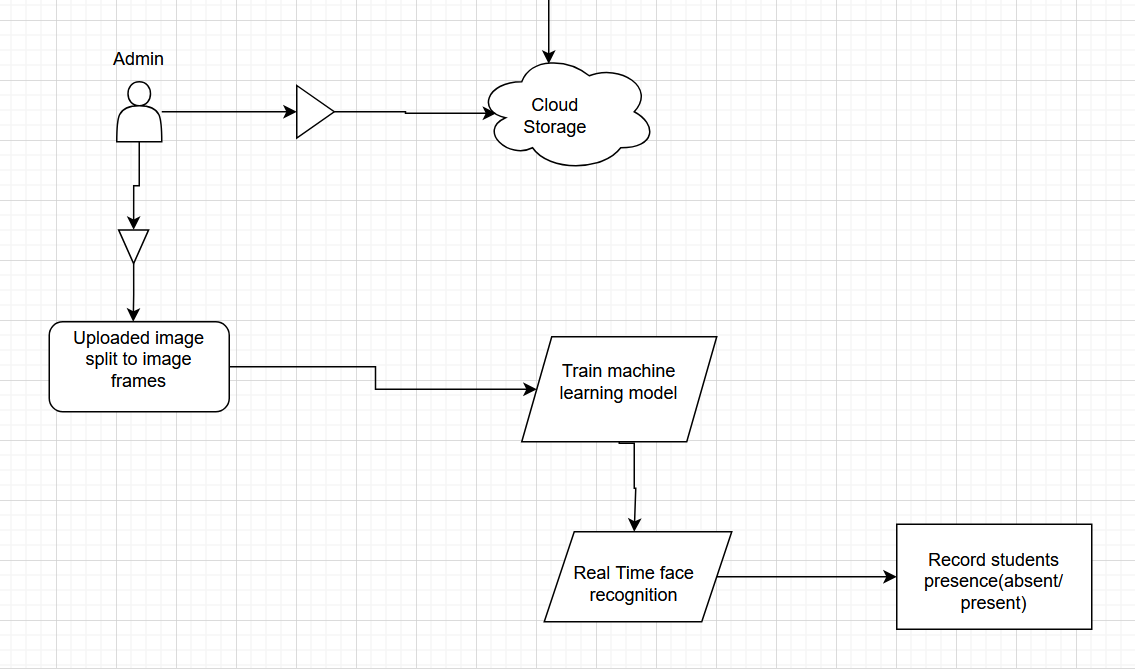
# 4.5. System analysis tools

**4.5.1 Use case diagrams**

This diagram will show how users can interact with the system. The interaction between the user and the system will represent how the system can accomplish its set goals. 

**4.5.2 Flow chart**

This diagram will show a breakdown of the system from how it initiates, takes in data, performs logical operation on it according to a set of predefined instructions and produce a desired output.



*Figure 8 Flow chart*

**4.5.3 Data flow diagram**

1. This diagram shows the road map of data as it circulates in the system. It will show how data will enter into the system, the different processes it will undergo, where it will be stored, and finally, the output information. The design of this diagram is illustrated below.

*Figure 9 Flow of the system*

# 4.6 System investigation

## 4.6.1 Introduction

A preliminary investigation of the system involved data collection through various fact-finding techniques. Firstly, the data were identified and distinguished between the existing and proposed system. Using fact-finding techniques, we could fully comprehend the current system to determine whether the proposed system could be implemented successfully and whether it was feasible.

## 4.6.2 Data collection

The primary data collection technique was by use of Literature review. The primary source was Google Scholar, where many relevant and significant articles were analyzed. Studies showed that most students find face recognition the best way to sign for class attendance since it has more benefits than traditional methods, including signing attendance sheets. A General web search also concluded that institutions have opted for facial recognition as it is cheaper in the long run and has more accuracy.

**4.6.3 Fact Recording**

# Introduction to Fact recording methods:

Input requirements- The proposed system will require students to upload their current image. The image should only contain one person at a time, proper lighting, and image quality.

Output requirements- The proposed system will be required to give detailed information containing the date, time, and the number of times a student has been to class.

Process requirements- The proposed system will require training the images using machine learning. The trained model will be used to compare the video Livestream to the model.

# 4.7 System analysis

**4.7.1 Functional requirements**

This is what the system will be able to do.

**4.7.1.1 Process-oriented functional requirements**

The system will be able to record the time a student gets into class- The live video stream will be used to show what time a student gets into class. This information will be used by the machine learning model to recognize a student and record the time a student is in class.

The system will send an email to every student who is registered for the class. The email will entail a confirmation that the student was present during the lecture or if he was absent.

The machine should have a high degree of accuracy in recognizing students (70% and above).

**4.7.1.2 Information-oriented functional requirement**

The system should contain a list of all students registered for the lecture (names and admission numbers).

**4.7.2 Non-functional requirements**

These are the various characteristics the system should possess.

**4.7.2.1 Operational requirements**

Responsiveness to all devices- The mobile application used to register student details should be operational on all devices (android, ios)

**4.7.2.2 Performance requirements**

1. Working schedule – the system should operate 24/7 after installation.

**4.7.2.3 Security requirements**

Only registered students should be able to register on the application.

**4.7.3 User requirements**

It contains a list of what the users need for the system to be functional. They include:

The mobile application uploads the images to the cloud and internet connection.

# CHAPTER 5: SYSTEM DESIGN

## 5.1 Introduction

The Class Attendance System using Facial Recognition and Mobile/web App is an innovative solution that aims to streamline and enhance the process of recording and managing student attendance within educational institutions. By leveraging the power of facial recognition technology and the convenience of a mobile application, this system offers an efficient and reliable method of tracking attendance, reducing manual efforts, and improving accuracy.

Traditional methods of taking attendance, such as paper-based sign-in sheets or manual roll-calls, can be time-consuming, prone to errors, and susceptible to misuse. With the advent of facial recognition technology and the widespread use of mobile devices, this system presents a modern and automated approach to address these challenges.

The system consists of two primary components: a mobile application and a backend infrastructure. The mobile application serves as the user interface, allowing students to upload videos of themselves for attendance verification, while teachers and administrators can manage attendance records and generate reports. The backend infrastructure comprises a facial recognition engine, which processes the uploaded videos, detects and recognizes faces, and matches them with the student profiles stored in the system's database.

The system design encompasses various aspects, including user management, video processing, facial recognition algorithms, database management, and secure communication protocols. It embraces principles of modularity, encapsulation, and separation of concerns to ensure a well-organized and maintainable codebase. Additionally, the system prioritizes security and privacy by implementing encryption, secure authentication mechanisms, and compliance with privacy regulations.

Through the integration of these components and the utilization of cutting-edge technologies, the Class Attendance System using Facial Recognition and Mobile App offers several advantages. It reduces the time and effort required for attendance management, eliminates the possibility of proxy attendance or errors in manual recording, and provides real-time attendance tracking. Moreover, the system enhances data accuracy, facilitates data analysis, and enables seamless integration with existing educational systems or learning management platforms.

This system design document will delve into the details of the system's architecture, functionality, data flow, and logical design. It aims to provide a comprehensive understanding of how the system operates, enabling efficient implementation and successful deployment.

By implementing the Class Attendance System using Facial Recognition and Mobile App, educational institutions can embrace technological advancements to revolutionize the attendance tracking process, improve operational efficiency, and enhance overall student experience.

## 5.2 Objectives of System Design

Objectives of this chapter will include:

* Streamline Class Attendance: The primary objective is to create a system that automates and simplifies the process of taking class attendance. The system should eliminate manual attendance tracking methods, reduce administrative burden, and provide accurate and efficient attendance management.
* Improve Accuracy and Reliability: The system should leverage facial recognition technology to enhance the accuracy and reliability of attendance records. By using biometric data, the system aims to minimize errors, eliminate the possibility of proxy attendance, and ensure that attendance records reflect actual student presence.
* Enhance User Experience: The mobile app should provide a user-friendly interface for both teachers and students. The system should offer seamless and intuitive interactions, allowing teachers to easily take attendance and students to conveniently view their attendance records.
* Optimize Time Efficiency: The system should minimize the time required for attendance management tasks. By automating the process through facial recognition and video-based attendance, teachers can focus more on instructional activities rather than manual administrative tasks.
* Enable Real-Time Attendance Tracking: The system should enable real-time attendance tracking, allowing teachers to access attendance data instantly. This feature provides up-to-date information and enables immediate intervention in case of attendance discrepancies or emergencies.
* Ensure Data Security and Privacy: The system should prioritize the security and privacy of student data. Appropriate measures, such as encryption, access control, and compliance with privacy regulations, should be implemented to protect sensitive information.
* Accommodate Scalability: The system should be designed to accommodate potential future growth, such as an increasing number of students or additional features. Scalability considerations ensure that the system can handle expanding user demands without significant performance degradation.

## 5.3 System Design Tools

This section illustrate different Unified Modeling languages diagrams that represent the entire flow of the system. The diagrams to be used in this project are flow chart, data flow diagram and a use case diagram.

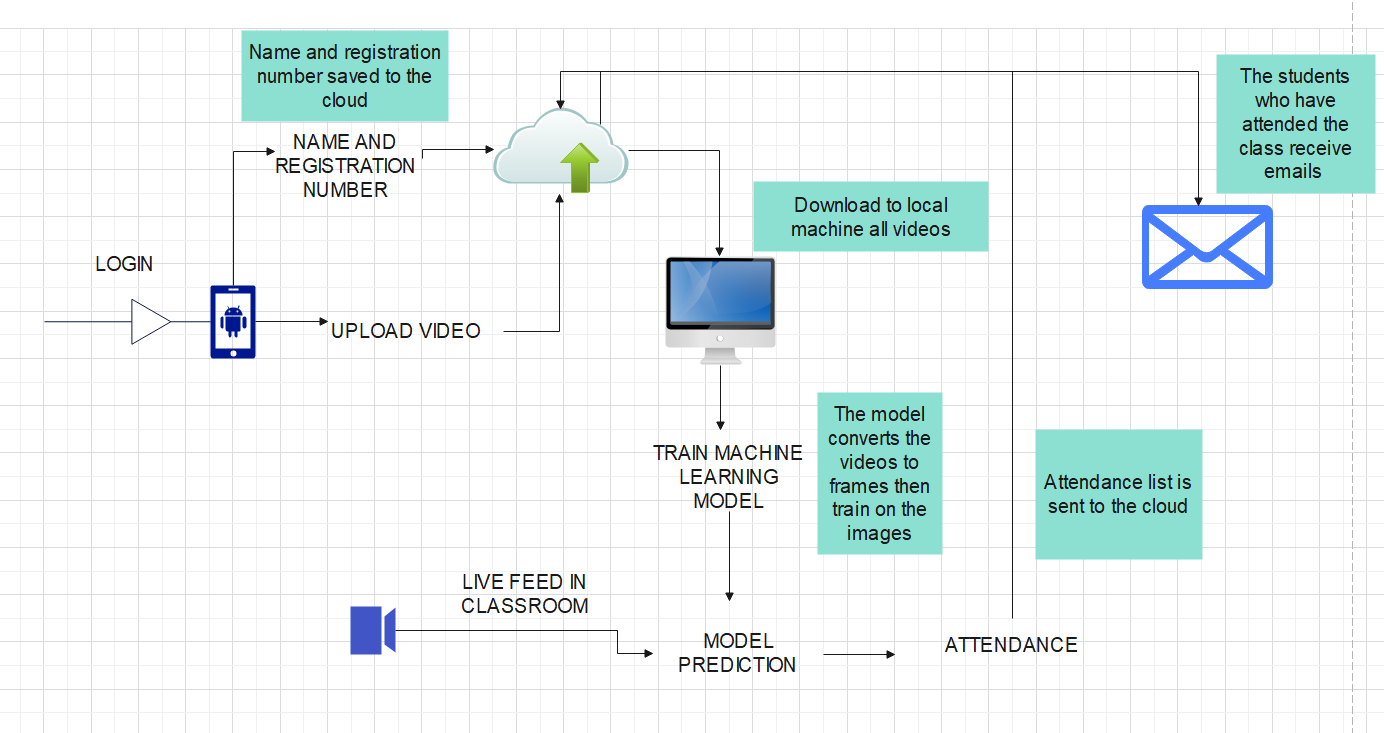
### 5.3.1 Flow chart

The system will consist of several interconnected modules that work together to perform specific functions. The input module (Video upload) will capture facial images of students in the classroom, which will be passed on to the pre-processing module. The pre-processing module will ensure that the captured images are of sufficient quality, and any necessary adjustments or corrections will be made to improve the accuracy of the recognition process. During the training process, the system will capture multiple images of each student's face from various angles and lighting conditions. These images will be used to create a database of facial features unique to each student.

The next module is the recognition module, which is responsible for identifying the faces in the captured images and matching them to a database of known students. The recognition module will use advanced algorithms to analyze the captured images and perform facial recognition using biometric data such as facial landmarks, texture, and shape.

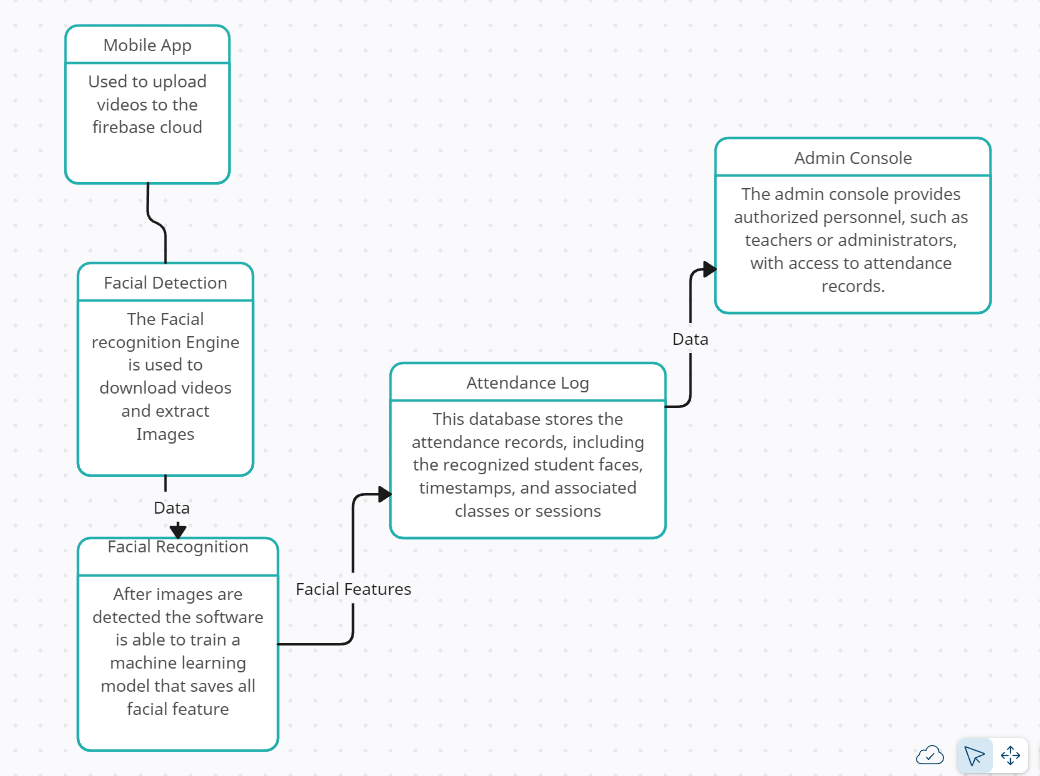
Once the recognition module has identified the faces in the captured images, the output module will generate a list of students present in the classroom, which will be used to update the attendance records. The attendance records will be stored in a database and can be accessed by authorized personnel to track students' attendance and performance.

In addition to the modules mentioned earlier, the system will also include a notification module. The notification module is responsible for sending email notifications to students who have attended the class.Once the attendance records have been updated, the notification module will access the attendance database and generate a list of students who have attended the class. The notification module will then use the email addresses stored in the database to send personalized emails to each student, notifying them that they have been marked as present in the class.



### 5.3.2 Data flow diagram

* + 1. The data flow in this system begins with the mobile app, where users initiate attendance requests. The facial recognition engine processes the uploaded videos, detects and recognizes faces, and matches them with existing student records. The attendance records are then stored in the attendance log database. Finally, authorized personnel can access and manage the attendance data through the admin console.
  + The Mobile App: The mobile app serves as the user interface for both students and teachers. It allows students to upload videos for facial recognition and teachers to request attendance.
  + User Interface: The user interface of the mobile app enables users to interact with the system, such as submitting attendance requests or viewing attendance records.
  + Facial Recognition Engine: The facial recognition engine is responsible for processing the uploaded videos and performing face detection and recognition algorithms.
  + Face Detection and Recognition: This component receives the video input and extracts facial features from it. It identifies and recognizes individual faces to determine student identities.
  + Attendance Log Database: This database stores the attendance records, including the recognized student faces, timestamps, and associated classes or sessions.
  + Admin Console: The admin console provides authorized personnel, such as teachers or administrators, with access to attendance records. They can view, manage, and generate reports based on the stored attendance data.

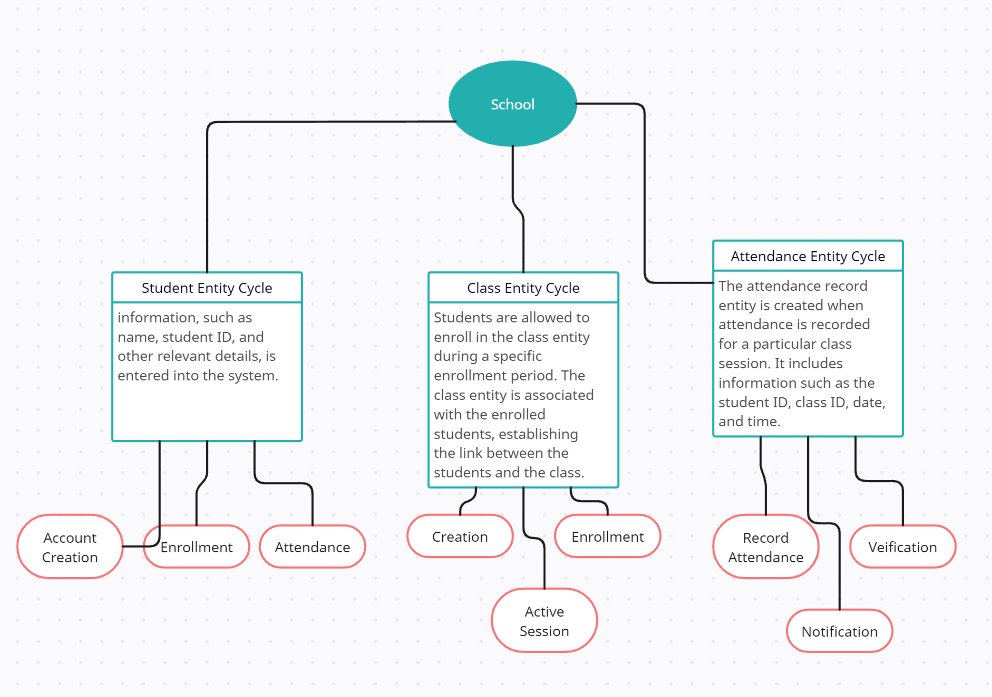


### 5.3.3 Use case diagram

This is a diagram that will show how users can interact with the system. The interaction between the user and system will represent how the different action a user is required to accomplish for the system to perform an action.

### 

### 5.4.1 Entity Life History

* + 1. The entity life cycle refers to the different stages or states that an entity goes through during its existence within a system. In the context of the class attendance system using facial recognition and the mobile app, the primary entities involved are students, classes, and attendance records. Here's a typical entity life cycle for each of these entities:
    2. Student Entity Life Cycle:
    3. a. Creation: A student entity is created when a new student joins the educational institution. Their information, such as name, student ID, and other relevant details, is entered into the system.
    4. b. Enrollment: The student entity is enrolled in specific classes or courses within the system, indicating the subjects they are registered for in a given semester or academic period.
    5. c. Regular Attendance: The student entity's attendance is recorded for each class they attend. This information is stored in the attendance records, including the date, time, and duration of the class attended.
    6. d. Graduation or Departure: When a student completes their studies or leaves the institution, their student entity may be marked as graduated or inactive, respectively.
    7. Class Entity Life Cycle:
    8. a. Creation: A class entity is created when a new class or course is offered within the educational institution. It includes details such as the course code, course name, instructor, and class schedule.
    9. b. Enrollment Period: Students are allowed to enroll in the class entity during a specific enrollment period. The class entity is associated with the enrolled students, establishing the link between the students and the class.
    10. c. Active Session: The class entity is marked as an active session when the class is in progress according to the scheduled dates and times. Attendance can be recorded for each active session.
    11. d. Completion: When the scheduled classes for a specific session are completed, the class entity is marked as completed or concluded.
    12. Attendance Record Entity Life Cycle:
    13. a. Recording Attendance: The attendance record entity is created when attendance is recorded for a particular class session. It includes information such as the student ID, class ID, date, and time.
    14. b. Verification: The attendance record entity goes through a verification process where the facial recognition engine matches the detected face with the student's profile in the system.
    15. c. Marking Status: The attendance record entity is marked as either present or absent based on the outcome of the facial recognition verification.
    16. d. Storage and Retrieval: The attendance record entity is stored in the attendance log database for future reference and can be accessed by authorized users for analysis or reporting purposes.
    17. 

## 5.5 Physical Design

### 5.5.1 Data Dictionary

* + - * 1. Student Entity

|  |  |  |
| --- | --- | --- |
| **Attribute Name** | **Data Type** | **Required** |
| Student Name | String | Yes |
| Registration No. | Var Char | Yes |
| Course | String | Yes |

Attendance Record Entity

|  |  |  |
| --- | --- | --- |
| **Attribute Name** | **Data Type** | **Required** |
| Student Id | VarChar | yes |
| Class ID | VarChar | yes |
| Time | DateTime | yes |
| Date | DateTime | yes |
| Status | String | yes |

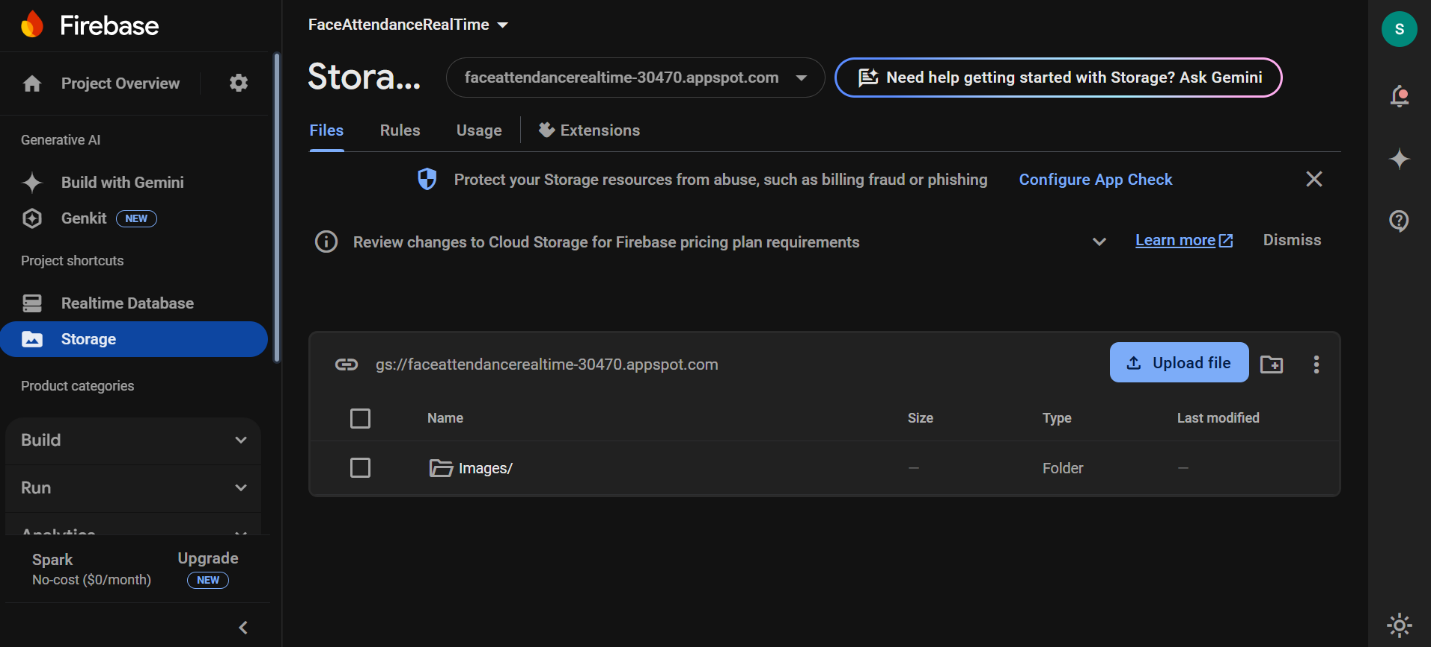
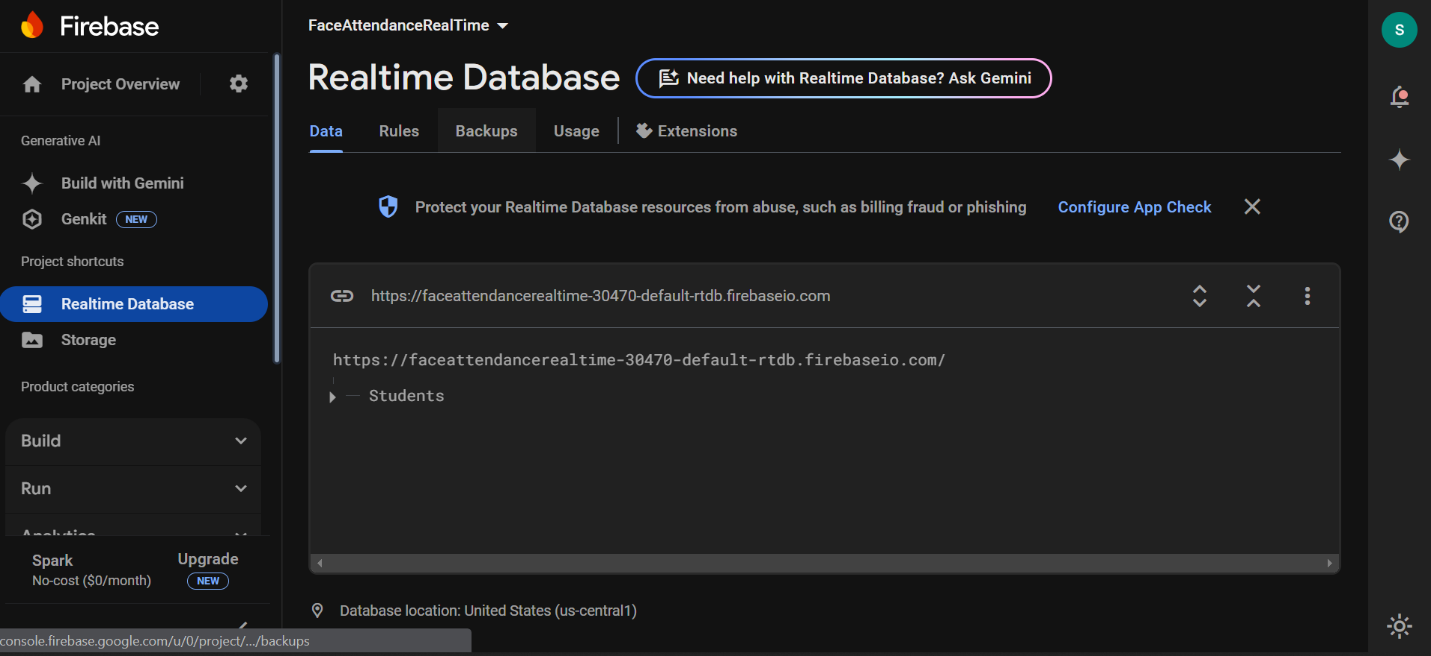
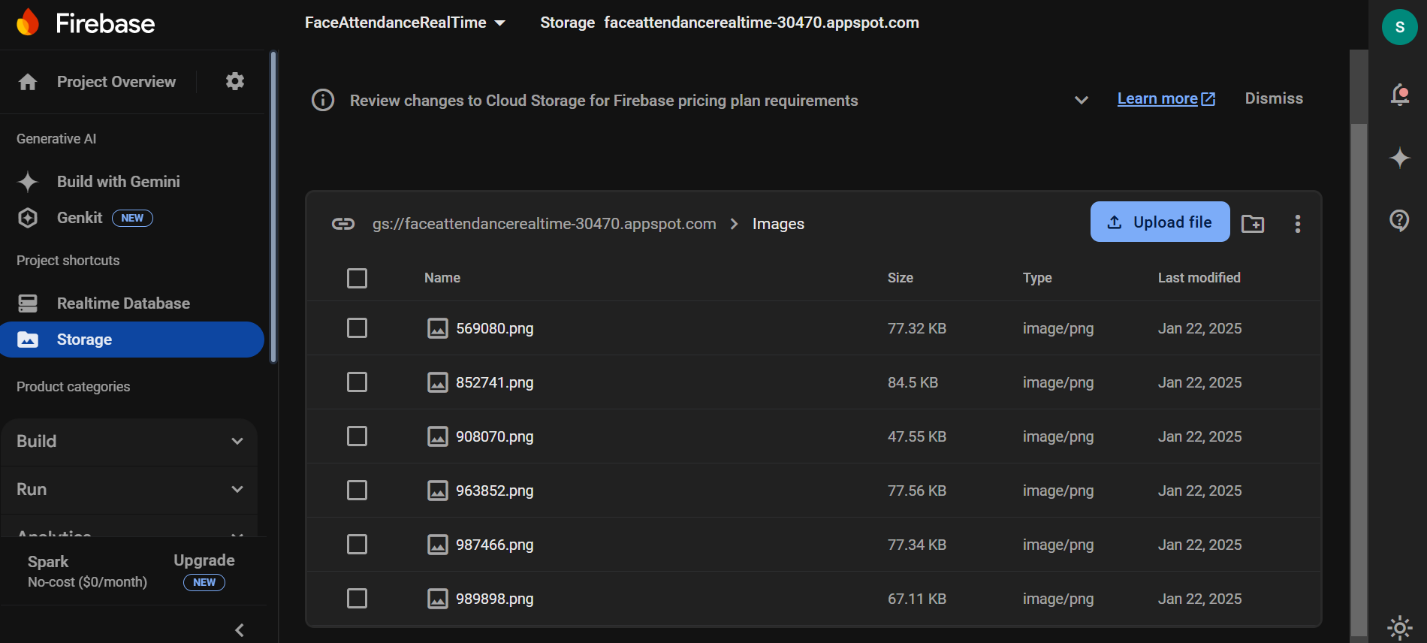
Facial Recognition Model

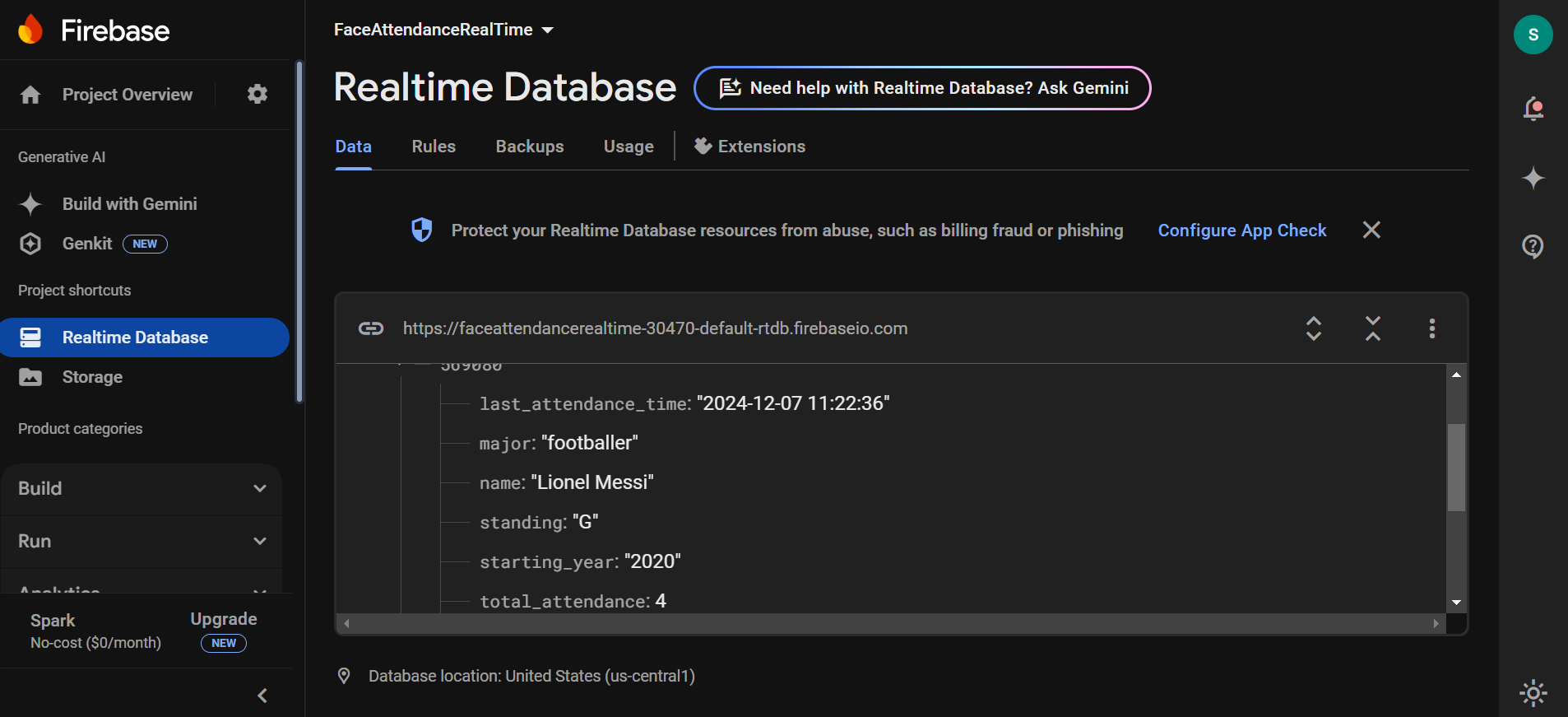
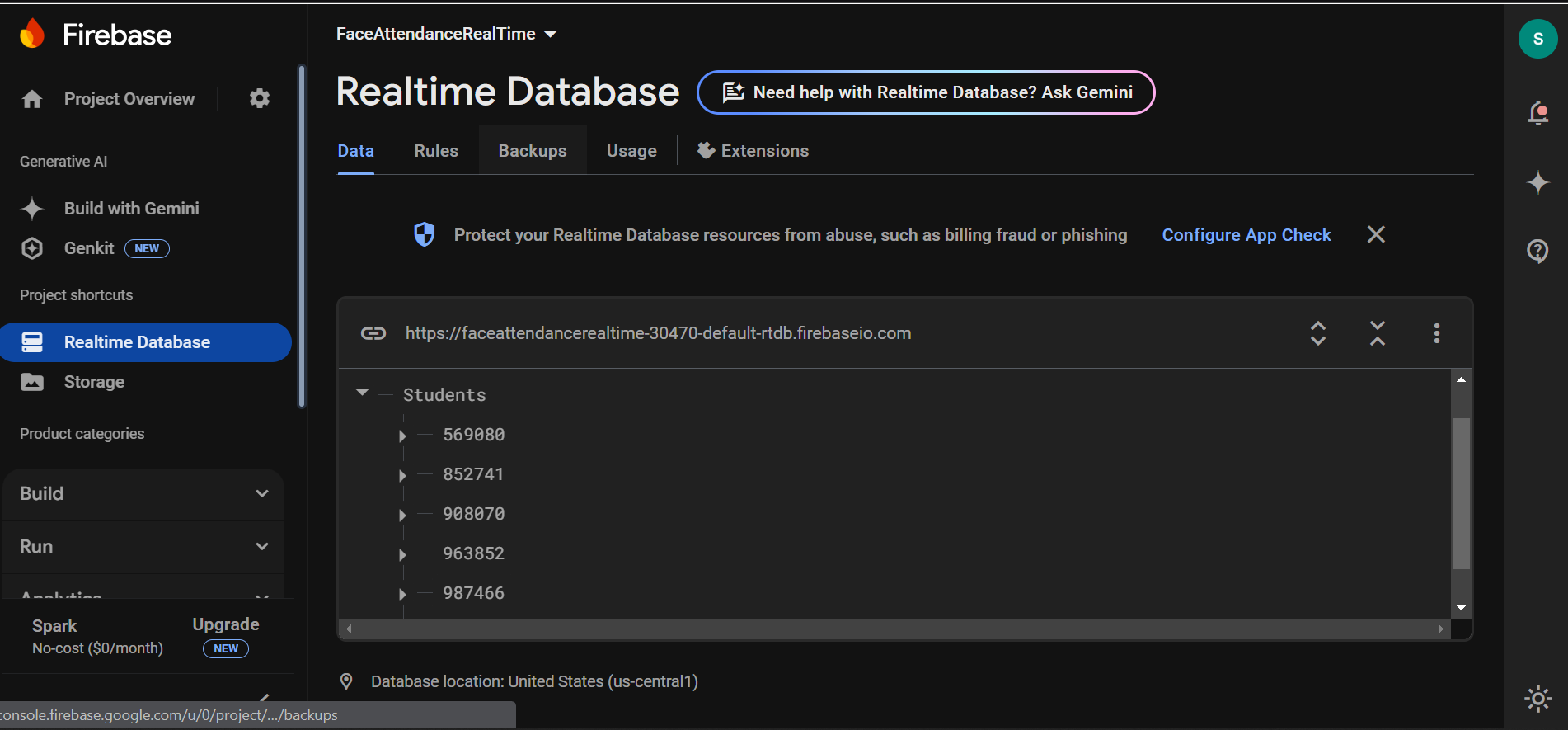
|  |  |  |
| --- | --- | --- |
| **Attribute Name** | **Data Type** | **Required** |
| Model ID | H5 | Yes |
| Training Data | images (png, jpeg) | Yes |
| Confidence Threshhold | Float | Yes |

Attendance Log Database

|  |  |  |
| --- | --- | --- |
| **Attribute Name** | **Data Type** | **Required** |
| Attendance ID | VarChar | Yes |
| Time Stamp | DateTime | Yes |

### 5.5.2 File/Database Design

* + 1. The file/database design plays a crucial role in the efficient storage, retrieval, and management of data within the Class Attendance System. It involves determining the structure of the database, defining tables, establishing relationships between entities, and optimizing data access.
    2. 
  1. 



### 

### 

# 

# 

# 

# 

### 5.5.3 Code Design

* + 1. Code design refers to the organization and structure of the software code that implements the class attendance system using facial recognition and the mobile app. It involves breaking down the system's functionality into modular components, defining relationships between these components, and designing the code structure for efficient development, maintenance, and extensibility. Here are some aspects of code design for the system:
    2. **Modular Design:**
    3. Divide the system's functionality into smaller, manageable modules or classes based on their responsibilities. For example, separate modules can handle user authentication, video processing, facial recognition, database operations, and user interface interactions.
    4. **Code Reusability and Maintainability:**
    5. Write modular and reusable code components to minimize redundancy and promote code maintainability.
    6. Encapsulate frequently used functionalities as reusable functions or methods to avoid duplicating code.
    7. **Error Handling and Exception Management:**
    8. Implement proper error handling mechanisms to gracefully handle exceptions and errors that may occur during runtime.Use try-catch blocks or exception handling techniques to catch and handle errors, ensuring the system's stability and preventing crashes.



# 



# 

# 

### 5.5.6 Process / Program Design / UML

#### 5.5.6.1 System Flowchart

User Authentication:

The system starts with user authentication, where users (students, teachers, administrators) provide their credentials to access the system.If the authentication is successful, the user proceeds to the main system functionality. Otherwise, they are prompted to re-enter valid credentials.

Mobile Application Functionality:

Once authenticated, users interact with the mobile application to perform various actions. Students can upload videos of themselves for attendance verification, view their attendance history, or access other relevant features.

Video Upload and Processing:

When a student uploads a video for attendance verification, the system initiates the video processing stage. The system extracts frames or keyframes from the video to obtain images for facial recognition. These extracted images are passed to the facial recognition engine for further processing.

Facial Recognition:

The facial recognition engine receives the extracted images and applies facial detection and recognition algorithms.It compares the detected faces with the stored student profiles to verify attendance.The engine generates a match result for each detected face and associates it with the corresponding student.

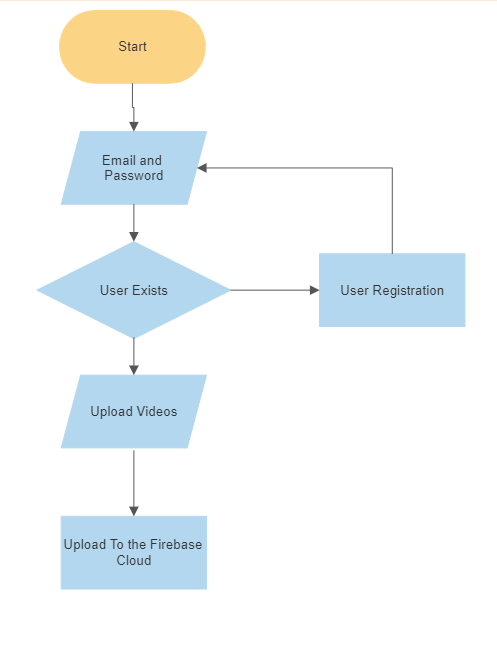
Attendance Recording:

Based on the match result from the facial recognition engine, the system records the attendance status for each student. If a match is found, the student's attendance is marked as present; otherwise, it is marked as absent.

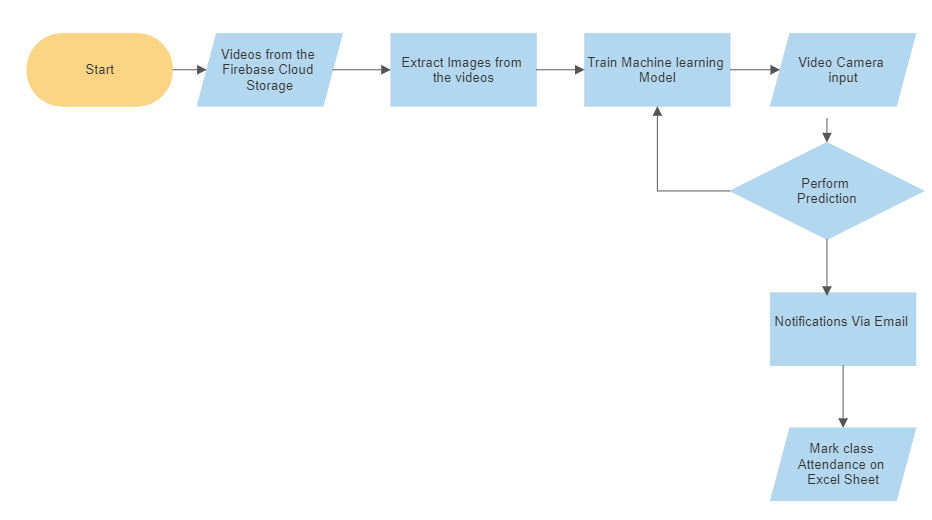
Reporting and Notifications:

The system provides reporting functionalities, allowing teachers and administrators to generate attendance reports based on specified parameters.Notifications can be sent to students, teachers, or administrators regarding attendance status updates, reminders, or other relevant information.

Student Flowchart



System Flowchart



# CHAPTER SIX: SYSTEM IMPLEMENTATION

**Tools used for coding and testing**

The following tools were used to build and test the mobile application.

1. Android studio- This IDE was used to develop and test the application. It has tools that enable code editing, testing, and debugging. It also has the android virtual device (AVD) manager, which acts as an emulator for running applications on the computer.
2. Firebase- It provides a real-time database that stores data as JSON tree objects, and all users connected to the database receive live updates as changes are made.
3. Visual Studio code- This IDE is used to develop and test the python code. It can also be used for debugging and testing the code.
4. Anaconda is an IDE mainly used with VS code to create an environment variables environment.

## 6.1 Coding/Environment/Debugging/Techniques

**6.1.1 Software coding/Environment/Debugging/Technique**

* **Front-end:** The mobile application was built using Android Studio/vscode for face detection.
* **Back-end:**  The back-end was built by the services of Firebase. It was used to store the videos and relay the emails to students. Tensorflow, Numpy, Open Cv, and Mediapipe - all modules within python were used to train the machine learning model and offer accurate predictions to the system.
* **Environment:** To successfully install all the modules needed to work, an environment had to be created with the anaconda platform. Without these environment modules, like Open CV lacked some functions that would be needed for the system to be functional.

## 6.2 Program Listing

This includes the additional comments and print statement outputs used as the testing was one. The information outputted by the compiler, as well as the respective error messages, if any, will be included in the Appendix part.

## 6.3 Test System/Program Testing

This was done to ensure the system objectives are met and its purpose is achieved in a transparent manner. Functionality test, compatibility test, its performance on different devices, and a test on different modules/units were done.

## 6.4 Test Plan

* 1. **6.4.1 System Test**

This was conducted on the whole system. All the android mobile application components and modules were integrated and then tested to determine if the standards were being met. It also involved identifying the accuracy level of the prediction model. The percentage accuracy needs to be above 70% for it to give a prediction.

* 1. **6.4.2 Functionality Test**

The system's different functions were tested to meet the objective set for the system. This ensured that the system met the developer’s goals and user-specific requirements. After the test, everything seemed to work as expected.

* 1. **6.4.3 Compatibility Test**

This was done to ensure that the mobile application could run on different mobile phones. However, during the test, it was discovered that the python code requires python 3.10 and above.

## 6.5 Test Data

This clearly describes how data is stored in different Firebase services, which are stored as a collection.

# CHAPTER SEVEN: USER MANUAL - DOCUMENTATION

## 7.1 Installation environment

A student will be able to upload the application from my GitHub page https://github.com/Samuelwanyinge05/face-recognition.The admin will, on the other will, need to clone my GitHub repository to their local machine. The admin will require VS code installed, Anaconda IDE, python 3.12, a Firebase account, an active internet connection

## 7.2 Installation Requirements

1. A laptop or desktop, preferably running a 64-bit windows 10 operating system
2. Visual studio code
3. Internet connection
4. Webcam
5. Minimum of 8GB Ram
6. Anaconda IDE
7. Pycharm

**7.3 Installation Procedures**

1. Download the latest version of Visual Studio code, python 3.10, and Anaconda IDE.
2. Create a conda environment. To set up the environment, the user needs to go to cmd and type conda create –n ‘environment name’ python=3.12
3. After creating the environment, the user needs to activate the environment using the command conda activate ‘environment name.’ Once the environment is active, the user must install the python modules. The modules will be installed using the command pip install mediapipe opencv-contrib-python NumPy pickle TensorFlow smtplib SSL. The installation of all the modules will take some time.

## 7.4 User instructions

**7.4.1 Student Instructions**

1. The student will be required to download the app from https://github.com/Samuelwanyinge05/face-recognition
2. After installing the application, the student must upload a video of themselves. The video requires the student to be alone. No other person in the video has proper lighting. If the student requires spectacles, he/she is supposed to put them on during the video recording.
3. After selecting the video that meets all the specifications, the student is supposed to fill in their name and registration number and then click on upload
4. **7.4.2 Admin Instructions**
   1. Visit <https://console.firebase.google.com/project/python-facial-recognition/storage>, then download the folder named videos
   2. To convert the videos to still images, the user must run picture.py. This python script will convert the video to images.
   3. The user will then be required to run trainer.py. This code will start to train a machine learning model and give a model named keras\_model.h5.
   4. Finally, the user will be required to run model.py. After training the model on student images, the code can run independently. It will be able to record the time a student gets into the classroom and send emails to them once they are recognized.

# CHAPTER EIGHT: LIMITATIONS, CHALLENGES, CONCLUSIONS, AND RECOMMENDATIONS

## 8.1 Limitation

* Time constraints. The application was built alongside a regular school semester which interfered with the project ‘s completion schedule
* Financial constraints- The system requires high-end cameras, which tend to be expensive.
* The application was developed using an android platform, which is not available in other operating systems. The application needs to be launched on an android smartphone; hence, it will be limited to users who do not have android operating system-based smartphones.

## 8.2 Challenges

The most difficult challenge I encountered was training a machine learning model that requires a laptop with high computational power. This limited me in the dataset that I collected since my machine barely meets the standards of training a model.

Earlier I created a mobile application to only upload images, but that would require students to upload more images which would be tedious; in addition, the images would have several facial angles, which would make the system fail in detecting students from different angles.

## 8.3 Degree of Success

The mobile application was completed successfully based on the objectives specified in this project. I created a model with a high accuracy of about 70% on every prediction.

## 8.4 Learning experience

In the process of developing this system, I was able to learn several things. I could interact with Java for mobile development, which developed my skills in android. Firebase enabled me to work with datasets in real time. I was able to continue developing my skills in python, in particular, working with computer vision, I got an idea of how the system can be used in different areas.

**8.5 Recommendation**

* Increasing the number of webcams to get a high viewing angle so no student will be recorded missing when they are in the classroom.
* Better lighting within classrooms
* Training and Support: Provide comprehensive training materials and resources to users to ensure they are proficient in using the system effectively. Additionally, establish a dedicated support channel to address user queries, troubleshoot issues, and provide timely assistance
* Enhanced Mobile App Features: Continuously improve the mobile application by adding features that further enhance the user experience. This could include features such as push notifications for attendance updates, in-app messaging for communication between students and teachers, and a personalized dashboard providing relevant attendance statistics and insights to students and teachers.

REFERENCES

Manjula, Dr. A. and Kalpana, D. and Guguloth, Sanjay, Facial Recognition Attendance Monitoring System using Deep Learning Techniques (March 23, 2023). International Journal for Innovative Engineering & Management Research, Vol. 12, No. 3, 2023

Smartphones with the face unlock feature - Times of India. (2020, February 28). The Times of India. [https://timesofindia.indiatimes.com/most-searched-products/mobilephones/mobiles/smartphones-with-the-face-unlockfeature/articleshow/74381235.cms?from=mdr](https://timesofindia.indiatimes.com/most-searched-products/mobile-phones/mobiles/smartphones-with-the-face-unlock-feature/articleshow/74381235.cms?from=mdr)

Disadvantages of RFID. Mostly minor, or you can minimize them. (2019, March 7). Wireless Technology Advisor. [https://www.wireless-technologyadvisor.com/disadvantages-of-rfid.html](https://www.wireless-technology-advisor.com/disadvantages-of-rfid.html)

What do you think about student skipping classes in university? (2020, May 15). Quora. [https://www.quora.com/What-do-you-think-about-student-skipping-classes-inuniversity](https://www.quora.com/What-do-you-think-about-student-skipping-classes-in-university)

Mihajlovic, I. (2020, February 9). Everything you ever wanted to know about computer vision. Here's a look at why it is so awesome.

Medium. [https://towardsdatascience.com/everything-you-ever-wanted-to-know-aboutcomputer-vision-heres-a-look-why-it-s-so-awesome-e8a58dfb641e](https://towardsdatascience.com/everything-you-ever-wanted-to-know-about-computer-vision-heres-a-look-why-it-s-so-awesome-e8a58dfb641e)

7 applications of computer vision. (2019, October 18). ATRIA

Innovation.<https://www.atriainnovation.com/en/7-applications-of-computer-vision/>

Joseph, Jomon, K. "Automatic attendance management system using face recognition." International Journal of Science and Research (IJSR) 2.11 (2020).

Kanti, Jyotshana, and Shubha Sharm. "Neural network with face recognition". International journal of science and research IJSR (2019).

Romero, M., and Lee, Y. (2020). A National Portrait of Chronic Absenteeism in the Early Grades. New York, NY: The National Center for Children in Poverty.

Mittal, A. (2020, December 21). Haar cascades, explained. Medium. <https://medium.com/analytics-vidhya/haar-cascades-explained-38210e57970d>

Hickman, G.P., Bartholomew, M., and Mathwig, J. (2020). The Differential Development

Trajectories of Rural High School Dropouts and Graduates: Executive Summary. Phoenix, AZ:

The College of Teacher Education and Leadership at the Arizona State University at the West

Campus

Python OpenCV: Converting an image to grayscale. (2019, April 13).

techtutorialsx. [https://techtutorialsx.com/2018/06/02/python-opencv-converting-an-image-togray-scale/](https://techtutorialsx.com/2018/06/02/python-opencv-converting-an-image-to-gray-scale/)

Face detection vs. Face recognition: What's the difference? (2019, May 9). FaceFirst Face

Recognition Software.<https://www.facefirst.com/blog/face-detection-vs-face-recognition/>

Understanding the challenges of radio frequency identification (RFID) technology. (2019,

February 9). AssetWorks.<https://www.assetworks.com/rfid-technology/>