

Automated Attendance System



Group Members

Muhammad Aisam Irshad	19I-0847
Tayyaba Azam	19I-0851
Huda Shoaib	19I-0828

Project Supervisor

Dr. Mukhtar Ullah

Department of Electrical Engineering

National University of Computer and Emerging Sciences, Islamabad 2019

Developer's Submission

“This report is being submitted to the Department of Electrical Engineering of the National University of Computer and Emerging Sciences in partial fulfillment of the requirements for the degree of BS in Electrical Engineering”

Developer's Declaration

“We take full responsibility for the project work conducted during the Final Year Project (FYP) titled “**Automated Attendance System**”. We solemnly declare that the project work presented in the FYP report is done solely by us with no significant help from any other person; however, small help wherever taken is duly acknowledged. We have also written the complete FYP report by ourselves. Moreover, we have not presented this FYP (or substantially similar project work) or any part of the thesis previously to any other degree awarding institution within Pakistan or abroad.

We understand that the management of the Department of Electrical Engineering of National University of Computer and Emerging Sciences has a zero-tolerance policy towards plagiarism. Therefore, we as an author of the above-mentioned FYP report solemnly declare that no portion of our report has been plagiarized and any material used in the report from other sources is properly referenced. Moreover, the report does not contain any literal citing of more than 70 words (total) even by giving a reference unless we have obtained the written permission of the publisher to do so. Furthermore, the work presented in the report is our own work and we have positively cited the related work of the other projects by clearly differentiating our work from their relevant work.

We further understand that if we are found guilty of any form of plagiarism in our FYP report even after our graduation, the University reserves the right to withdraw our BS degree. Moreover, the University will also have the right to publish our names on its website that keeps a record of the students who committed plagiarism in their FYP reports.”

Muhammad Aisam Irshad

BS(EE) 2019-0847

Tayyaba Azam

BS(EE) 2019-0851

Huda Shoaib

BS(EE) 2019-0828

Certified by Supervisor

Verified by Plagiarism Cell Officer

Abstract

Maintaining the attendance of the students in an institution is a hefty task. There is always difficulty in handling attendance manually. This project aims at designing an attendance system that automatically monitors and manages attendance in an institution efficiently. The whole system is developed with ESP32 Cameras mounted at different places and angles in the room , databases, a web portal and different Machine Learning algorithms for face detection and recognition which will monitor attendance throughout the lecture.ESP32 microcontrollers are used to send live image data on Amazon Cloud that manages the database of students, teachers and class schedule. This system will reduce a lot of manual work of teachers and administrators of any institution as we will be providing the attendance to the academic office and will provide an option of adding, deleting and updating the records directly. The proposed work comprises four most popular trends in technology research; IoT , Machine Learning ,Cloud Computing and Web Portal Development.

Acknowledgements

We would like to express our sincere gratitude to our supervisor, Dr.Mukhtar Ullah for providing their invaluable guidance, comments and suggestions throughout the course of this project.

Table of Contents

Developer’s Submission 2

Developer’s Declaration 3

Abstract 4

Acknowledgements 5

Table of Contents 6

Chapter 1 Background and Introduction 8

 1.1 Motivation 8

 1.2 Problem statement 8

 1.3 Literature review 9

 1.4 Report Outline 9

Chapter 2 Solution Design & Implementation 10

 2.1 Block Diagram 10

 2.2 Flow Chart 11

 2.3 Software Implementation 12

 2.4 Hardware Implementation 25

Chapter 3 Implementation Plan 27

 3.1 Project Deliverables 27

 3.2 Gantt Chart and Project Schedule 27

 3.3 Major Work Distribution 28

 3.4 Budget 28

 3.5 Meetings 28

Chapter 4 Result and Recommendations 29

 4.1 Results of Trained Model 29

 4.2 Results on Web Portal 31

 4.3 Conclusions 32

 4.4 Recommendations / Future Work 32

Societal and Environmental Impact 32

UN Sustainable Development Goals 32

Lifelong Learning 33

 Appendix-A: Project Codes 34

 A1 Project Codes (ESP32 camera to AWS) 34

Table of Contents

A2 Project Codes (Creation of Database) 39

A3 Project Codes (Machine Learning) 44

Appendix B: Budget 51

Appendix C: Minutes of Meetings..... 52

References..... 53

Chapter 1 Background and Introduction

“Automated Attendance System” is based on cameras for real-time monitoring and machine learning models for data analysis. To achieve our target, we are using Amazon Web Services (AWS) and some other local services. This process involves different steps among which Machine-Learning is the main concept. Through Machine learning we can train our system for prediction and decision-making. We will be training a certain dataset containing the student images so the model can detect faces accurately. A Web Portal is also designed so that a user-friendly interface can be provided for more efficient use of this system.

After using all the above-mentioned methods, we will result in a highly economical, effective, and state of the art solution because it will automate the attendance and will overcome all of the problems mentioned below.

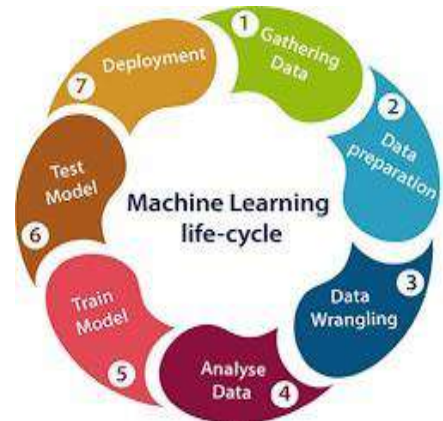


Figure 1 Machine Learning Lifecycle

1.1 Motivation

The motivation behind this proposed project was that the manual attendance system doesn't motivate students for better attendance percentages. A good attendance percentage improves student outcomes as it keeps them in loop with the curriculum. Chronic absenteeism or missing 10 percent or more of school days for any reason is a proven early warning sign of academic risk and school dropout [1]. Moreover, manual entry of attendance is always a risk of human error. Staff may record students' attendance incorrectly in an attendance sheet or the information may be incorrectly entered into web portals. It is a hassle for teachers to make analysis of attendance data manually and ensure no error prone in that process [2]. The concerned department can spend a lot of time correcting these errors which leads to an increase in their work. To solve these problems there was a need for an efficient system which could help with automatic marking of attendance.

1.2 Problem statement

Manual Attendance is an inefficient method as it is time-consuming and laborious. The administration has to update attendance manually resulting in proxies and errors. If this problem remains unresolved the workload will keep on increasing and maintaining attendance manually will be a tedious job. To resolve this problem, we propose an Automated Attendance System which will monitor students' presence and automate attendance, resulting in an efficient and optimized system.

1.3 Literature review

Precise Integrated Contactless Attendance Tracking, Recording and Analyzing System This paper is on a project in which attendance of employees is marked after facial recognition and temperature monitoring on the employee. The resultant data is sent to the cloud (AWS) and stored. As we are also using AWS as our cloud service so this paper will help us a lot in achieving the goal.[3]

Face Recognition System using Facenet Algorithm for Employee Presence

It highlights the features and accuracy of a Machine learning model Google Facenet. It is elaborating all the process in detail which will be helpful in our project. [4]

Face Recognition based smart attendance system using IOT

This paper discusses a project that consists of modules of OpenCV algorithms on python platform. To make use of IOT, email feature is used which is inbuilt in raspberry pi. The SMT Protocol helps us to do so. This basically will help us to integrate the hardware with the software. [5]

Face Recognition in Video Streams and its Application in Freedom Fighters Discovery - A Machine Learning Approach

This paper is elaborating the advantages of face recognition through machine learning and highlights the advantages of doing so. Moreover, it overall tells how machine learning models work. We will use these concepts in our project to have a better understanding.[6]

An approach towards development of automated attendance system using face detection and recognition.

This paper compares different techniques that can be used to automate attendance and emphasizes facial recognition techniques. We can have a good insight of those and apply in our projects.[7]

1.4 Report Outline

This report is further divided into multiple chapters as listed below.

In chapter 2, the proposed solution is discussed in detail. It includes details of block diagram, Flow chart, creation and deployment of database, implementation of machine learning algorithms and interfacing on ESP32 cameras locally and on cloud.

Chapter 3 discusses the overall work distribution, Gantt Charts, the overall expenses of the project and meetings with the supervisor.

Chapter 4 discusses the results acquired and output on the Web Portal. It further discusses the conclusions drawn from the obtained results and the recommendations/future work that is proposed for further enhancements.

The rest of the report includes important impact on the society, lifelong learning's and US Sustainable Development Goals

Chapter 2 Solution Design & Implementation

This chapter discusses the complete design and implementation of the proposed system. Section 2.1 discusses the block diagram and module specifications. The details of flow chart are presented in section 2.2 whereas, Section 2.3 discusses in detail the software implementation such as databases, web portal development, cloud computing and machine learning. The hardware implementation is presented in Section 2.5.

2.1 Block Diagram

Figure 2.1 shows the complete block diagram of the project. The details of each block with are discussed below.

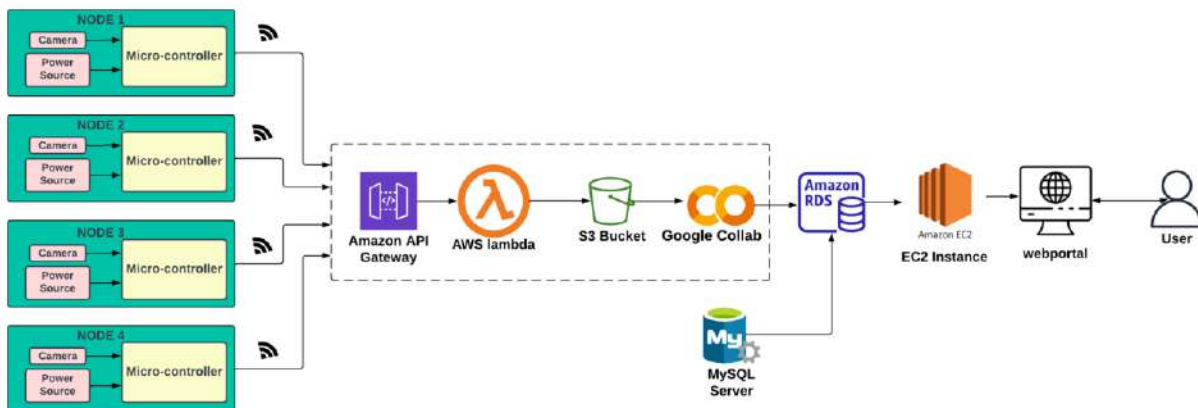


Figure 2. 1 Block diagram of the project.

- **Node:**

The nodes represented in the block diagram are ESP32 cameras which are sending images in bits through Wi-Fi.

- **Amazon API Gateway:**

It is an AWS service which is used to connect two softwares components to communicate with each other through certain protocols.

- **AWS Lambda:**

AWS Lambda function is used to encode the images in raw format. It will get triggered whenever the API gateway sends the request to it.

- **S3 Bucket**

S3 Bucket is an Amazon Web Service which is used for object storage i.e., in this project we are uploading our real time images to the S3 bucket.

- **Google Collaborator**

Google Collaborator is used to implement the machine learning model. We uploaded our own model also which will be the best suite for face recognition. It is a cloud platform with virtual resources i-e RAM, GPU, CPU etc.

- **MySQL Server**

MySQL server is used to make the student relational database.

- **Amazon RDS**

Amazon RDS is to store the database on cloud which is made on the local database server.

- **EC2 Instance**

EC2 Instance is used to connect databases with the backend of web portal.

- **Web portal**

Web portal is used to display all the student data and provide an interface to add /drop the data and perform operations directly on the database. It is also used to see the attendance of the students.

2.2 Flow Chart

Figure 2.2 shows the complete flow chart of the project. The details are discussed below.

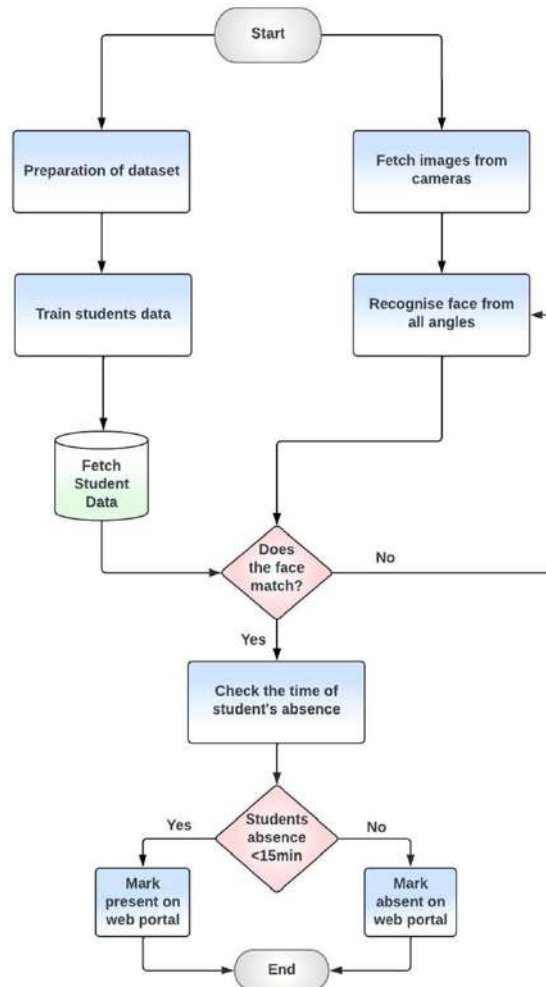


Figure 2. 2 Flow chart of project

In this flowchart there are two parts. The left side of the flowchart is related to the dataset and the right part is related to ESP32 camera execution. So, in this flowchart we are preparing a dataset of the students using an ESP32 camera. When the dataset is prepared, we will train it with the model we made. On the other hand, we are capturing pictures in real time using ESP32 camera and sending them to S3 bucket using API gateway. The faces will be recognized using the trained model. We are checking the time of student absence as well. If the absence is less than

fifteen minutes, then the student will be marked present on the portal else he will be marked absent.

2.3 Software Implementation

In software implementation we had four major areas:

- 1) Interfacing of ESP32 Cam with AWS
- 2) Design and Development of Database and its deployment on AWS.
- 3) Implementation of Machine Learning Algorithm
- 4) Development and Integration of Web Portal

2.3.1 Interfacing of ESP32 camera with AWS:

To send pictures from the ESP32 camera to the cloud we used AWS API Gateway and lambda function. API Gateway has two purposes:

- I. It generates a URL which makes a secure connection to S3 bucket.
- II. It accepts the request and sends it to the lambda function in our case.

The images are sent in bits so that the delay is minimized. When the lambda function receives the request from API Gateway, it will receive the bits, encode them in raw image and send it to S3 bucket. The code is in appendix A-1. Screenshots are attached below:

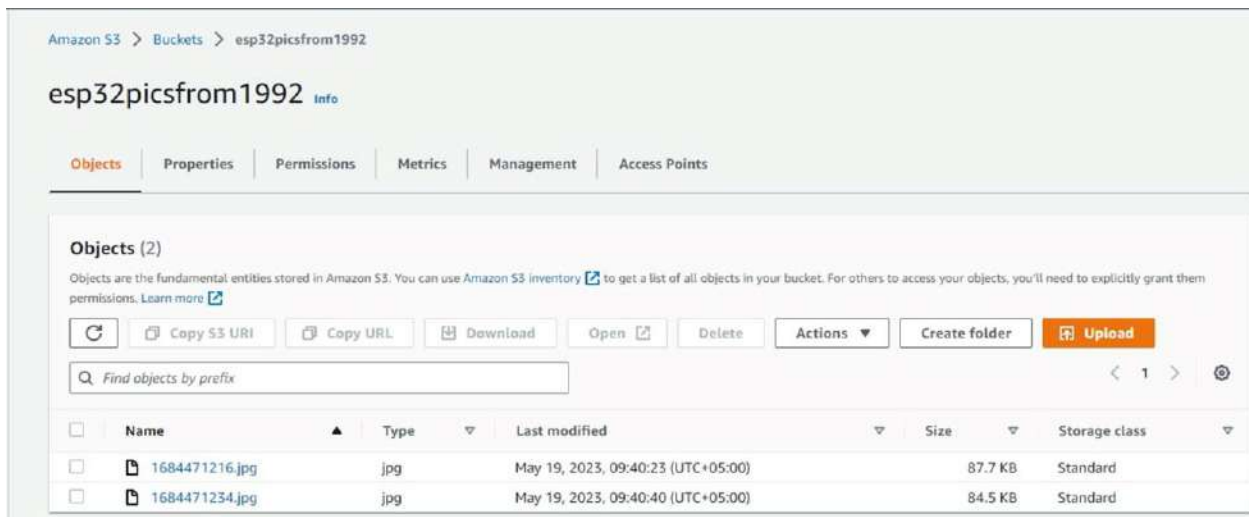


Figure 2.3. 1 S3 Bucket

Chapter 2: Solution Design & Implementation

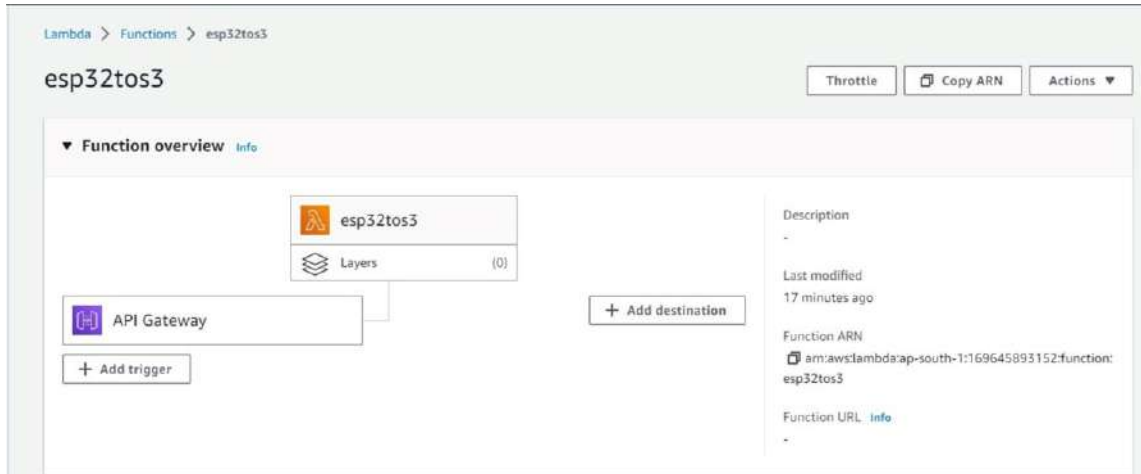


Figure 2.3. 2 Lambda Function

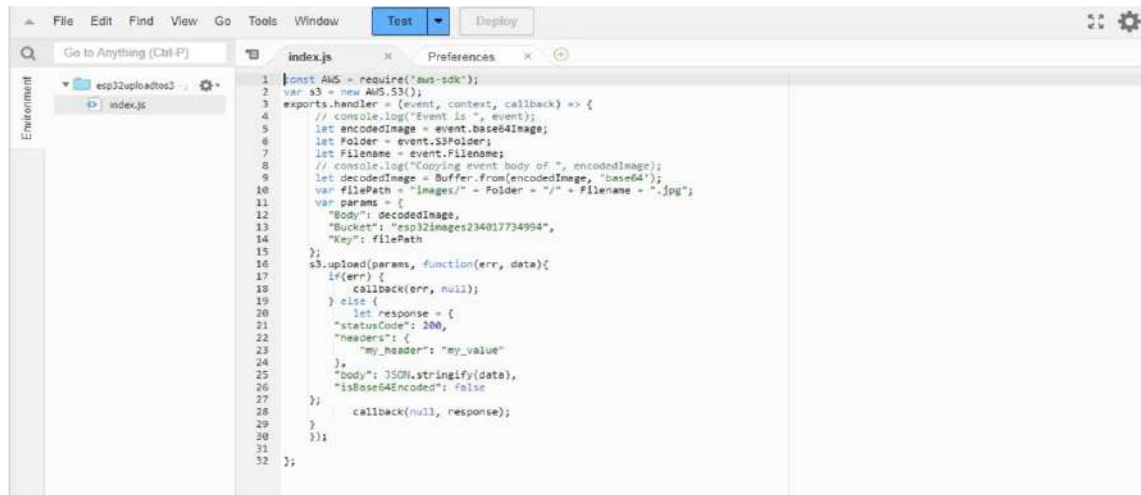


Figure 2.3. 3 Image Encoding in Lambda Function

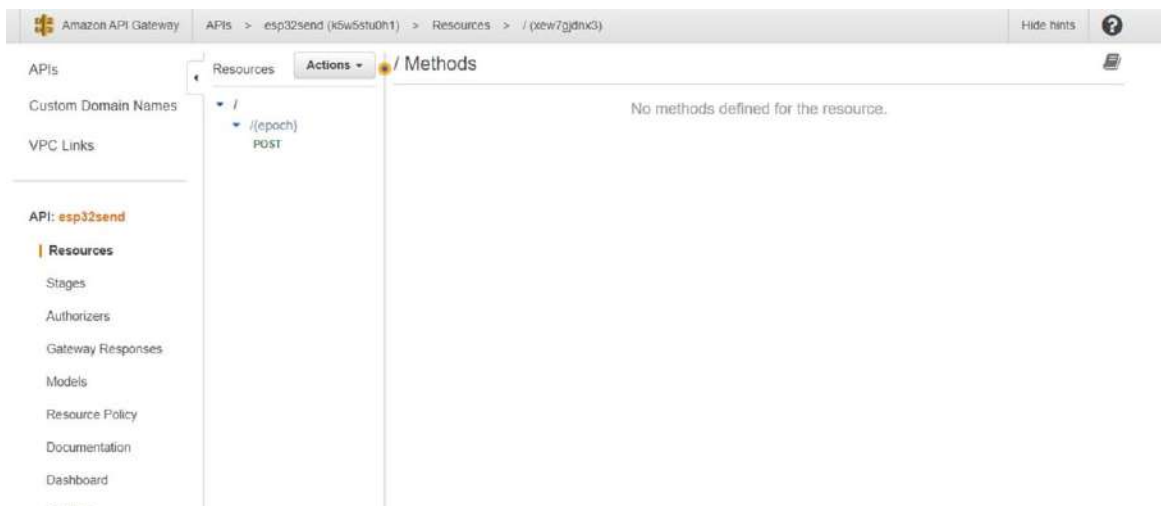


Figure 2.3. 4 API Gateway

2.3.2 Design and Development of Database and its deployment on AWS

Creation of Database

A database named as Attendance was created locally on Wamp Server as shown in Appendix A2. Figure shows the schema of the database that includes eight relations:

- Students with attributes Roll_No, Student_Name, Father_Name, Section, Dept_ID, Email, Phone, and Gender. Roll_No is considered as the primary key as every student will be having their own unique Roll number.
- Department containing attributes Dept_ID, Dept_Name, Dept_Block, Total_Floors where Dept_ID is the primary key and Foreign key that references to Dept_ID of relation Student.
- Teachers having attributes Teacher_Name, Teacher_ID, Phone_No, Email, Gender, Dept_ID and Course_No. Each teacher will have their specific and unique Teacher Identification so the attribute Teacher_ID is considered as primary key whereas Dept_ID is foreign key that creates a relationship between teachers and Department.
- Courses with attributes Course_No i-e primary key, Course_Name, Teacher_ID and Dept_ID. Attributes Teacher_ID and Dept_ID are foreign keys as they create relationship of relation Courses with Teachers and Department respectively.
- Room_schedule with attributes Course_No (Foreign Key), Section, Teacher_ID, Date, start_time and end_time. Here the attribute Course_No references the attribute Course_No of relation Courses.
- Enrollment containing attributes Roll_No and Courses_No, Both the attributes are considered to be the foreign keys as relationship is build with students and courses relation respectively.
- Students_attendance relation contains attributes Roll_No, Students_Name, Course_No and attendance where Roll_No and Course_No are foreign keys that build relation with Students and Courses respectively.
- loginform having attributes ID, User and pass. This relation is created to store login information of the Webportal.

Chapter 2: Solution Design & Implementation

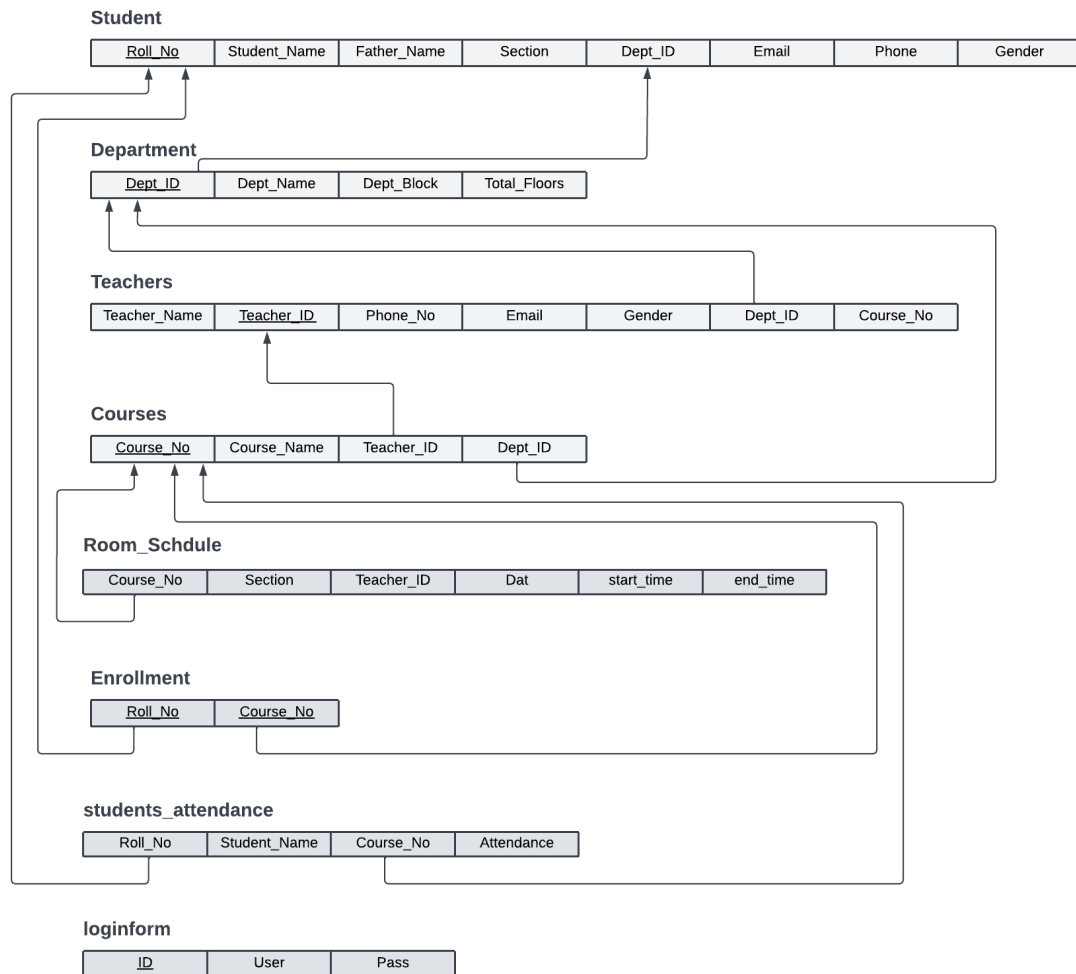


Figure 2.3. 5 Schema of Attendance Database

Creating RDS Instance on AWS

Relational Database (RDS) was used to develop and scale attendance database with high availability, following steps were taken to create RDS instance

- i. The “Create database” option was selected on the navigation pane of RDS service.
- ii. Database engine or engine type i-e MYSQL was selected
- iii. Database was configured i-e name of the database instance identifier (database-1), A master username (Admin), and a password for security was set. Storage type (db.t3. micro) was selected.
- iv. “Deletion protection” option was selected to prevent the database instance from being accidentally deleted.
- v. After the RDS instance was created an endpoint was generated.

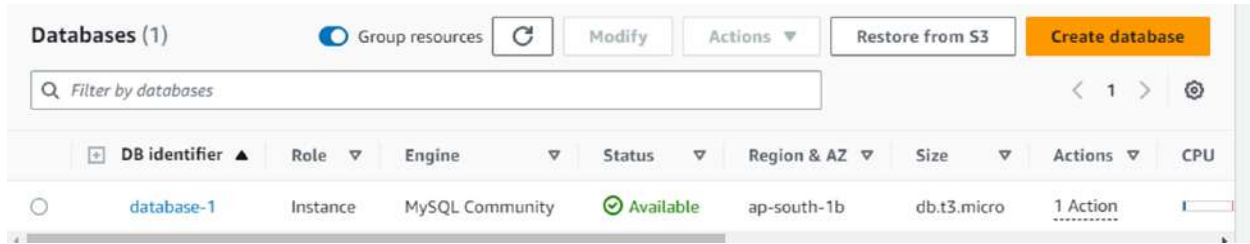


Figure 2.3. 6 RDS Instance

Importing Database to AWS

To import locally made database to AWS RDS, MYSQL workbench was used. For this purpose first a connection was made using the endpoint of database-1. To test the connection master credentials were used. Figure below shows the connection made named as fyp-database. Then a locally made attendance database was imported to this connection successfully as shown in figure.

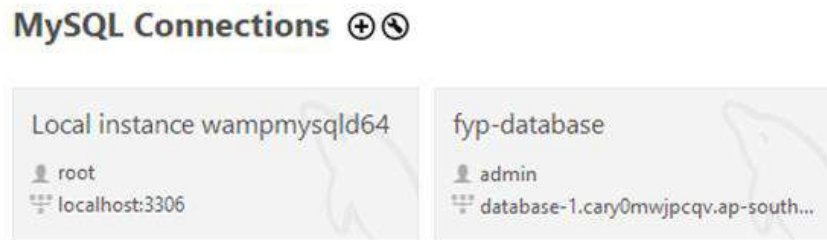


Figure 2.3. 7 Connection with Database



Figure 2.3. 8 Importing database on RDS.

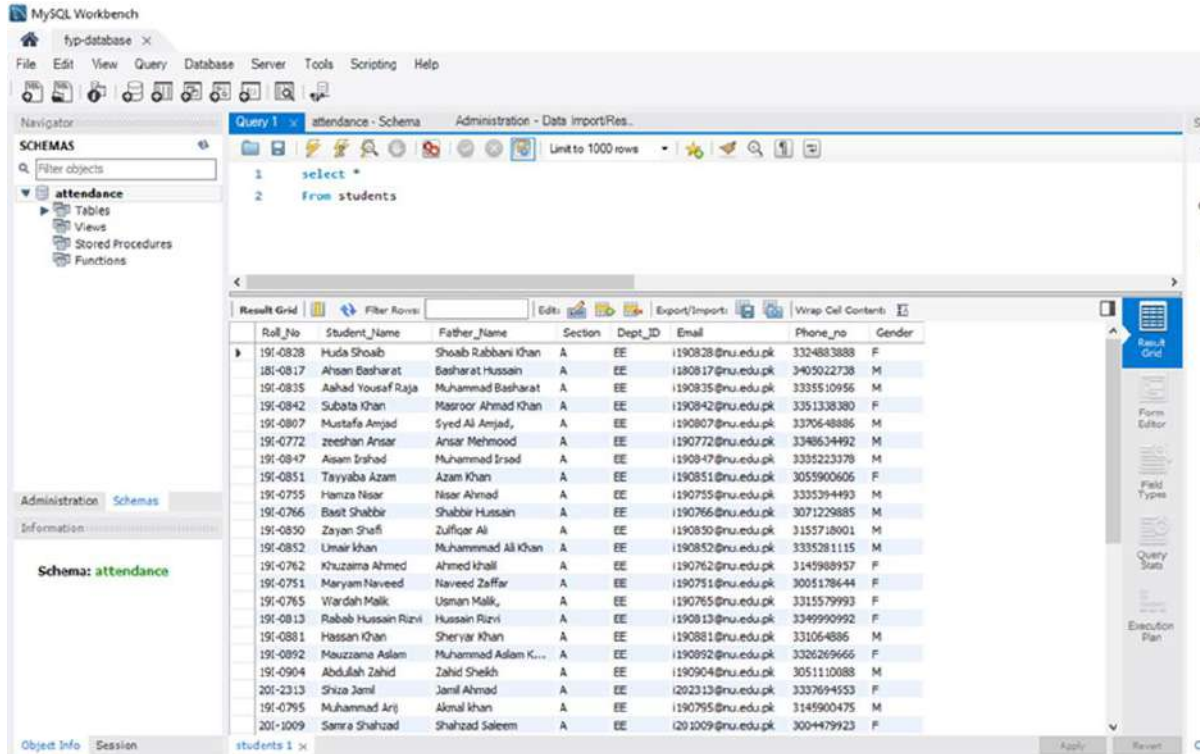


Figure 2.3. 9 Database imported on RDS.

2.3.3 Implementation of Machine Learning Algorithm

For Machine Learning three major phases are addressed.

1. Data collection and preprocessing
2. Feature Extraction
3. Model Training

Data Collection:

After thorough research, the appropriate datasets were made using ESP 32 Camera in which a good number of pictures were taken of each student from different angles. This data will help us figure out and test the accuracy of our model in a classroom environment. We prepared a dataset of 40 people with pictures in different poses. It includes pictures with and without spectacles. These pictures were taken in normal light so that they can be trained with ease.

Chapter 2: Solution Design & Implementation



Figure 2.3. 10 Captured Pictures for Dataset

However, the obtained datasets needed to be processed before they could be used.

Data preprocessing:

Effective processing of data lays a significant foundation for producing valuable results in machine learning work or computer vision tasks.

- Our preparatory stage for these elaborate procedures involves detecting primary visual cues that help identify human features accurately; hence the need for an advanced tool like MTCNN was designed explicitly with three critical steps-levels .
 - To start the process of identifying faces, a cascade of convolutional networks is employed in the face detection stage. The potential face regions are located and refined by using non-maximum suppression to eliminate overlapping bounding boxes.
 - In the subsequent stage - facial landmark localization- MTCNN detects key features like eyes, nose, and mouth within the detected face region. These features help align and adjust extracted faces.
 - Finally, to ensure consistency in alignment, the face alignment stage uses affine transformations based on detected landmarks and normalizes face orientation, size, and pose.

The code is attached to Appendix A3.1

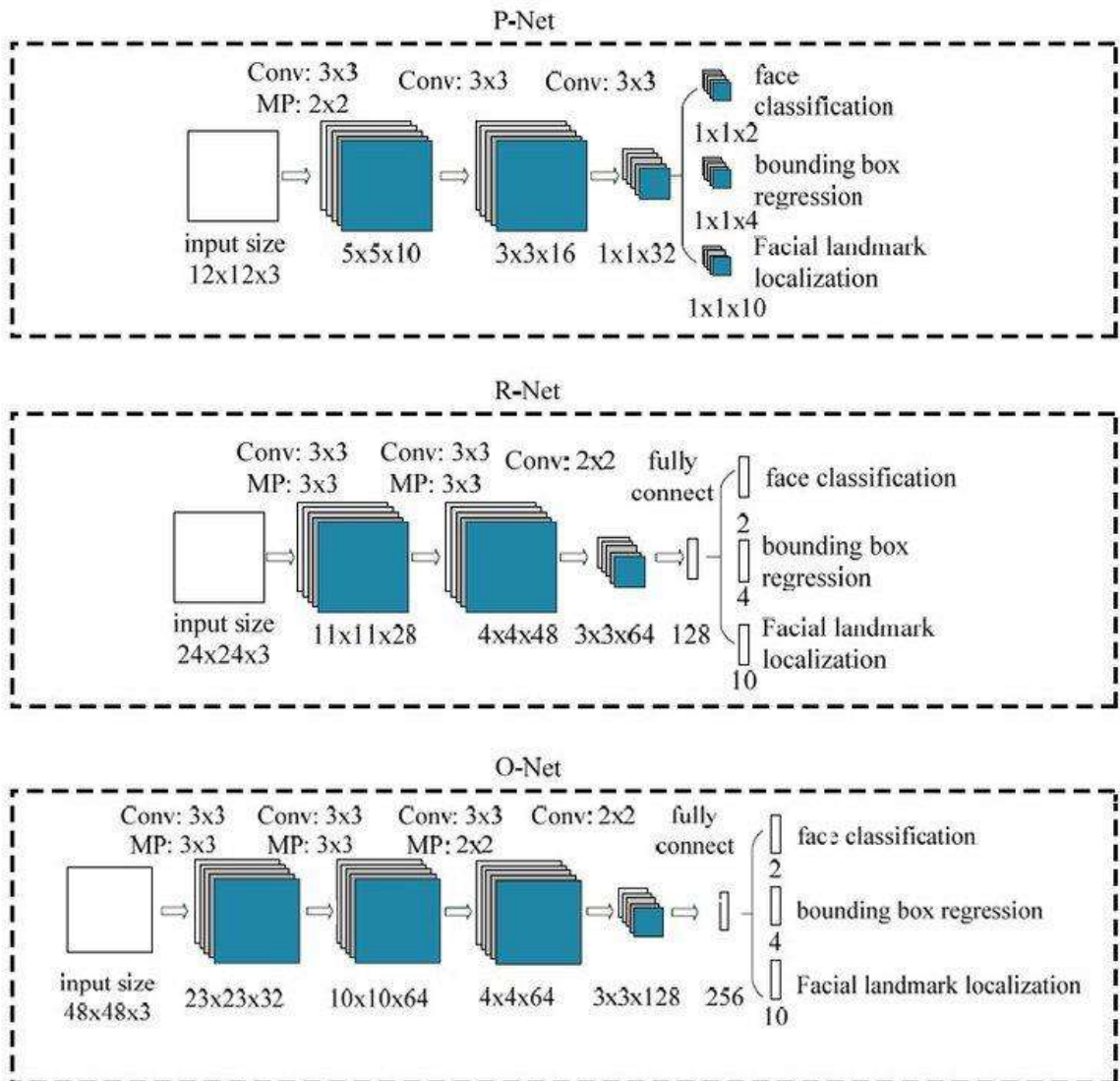


Figure 2.3. 11 MTCNN Architecture

Feature Extraction

Once MTCNN has extracted faces, we may need to make further modifications to fit the systems Embedded Extraction Using FaceNet Library. FaceNet uses a deep convolutional neural network (CNN) architecture to learn large datasets of face images. The mesh is designed to map face images into a high-dimensional space where people's faces are close together and different people's faces are far apart. This device allows facial recognition and recognition. To extract face embeddings using FaceNet, a pre-trained model is used for normalized and aligned face images. The mesh processes each image through its layers, resulting in a corresponding number or vector. These embeddings can be viewed as a condensed representation of facial features that can be used in subsequent face recognition. Combination of exposure and size normalization, along with the removal of FaceNet embeddings, provides several benefits for face recognition:

Chapter 2: Solution Design & Implementation

- **Reduced Differentiation:** Normalizing exposure and size helps reduce the impact of changes. from different faces and angles. This increases the power and accuracy of facial recognition.
- **Performance and Disable Notes:** FaceNet embeds provide a numerical comparison that captures unique faces. These embeds help to save, retrieve and compare face data efficiently.
- **Interoperability:** Standardized pose and size and FaceNet embeds facilitate interoperability between different face recognition systems. Multiple applications and models can be easily shared using a standard representation.
- **Improved Matching Performance:** By optimizing and optimizing face matching, the similarity measure for face matching becomes more efficient.
- **Normalized exposure and size makes matching increasingly reliable,** improving the overall performance of the face recognition system.

The code is attached to Appendix A3.2.

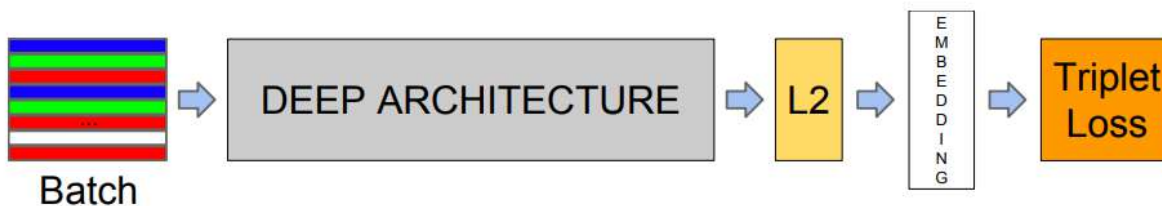


Figure 2.3. 12 Google Facenet Architecture

Model Training

We used the SVM (Support Vector Machine) classifier to train our model. We focused on important tasks such as splitting the data for testing and training, selecting the SVM classifier and selecting the appropriate kernel for the classifier.

- **Split Data into Testing and Training**

Before training the model, it is important to divide the available data into two groups: training and simulation. The training method is used to train the SVM classifier while the test method is used to evaluate its performance. The data classification process aims to create two mutually exclusive groups that represent the entire data set. Generally, a random sampling strategy is used to ensure an unbiased distribution. It is important to control the class distribution of the original data in training sessions and tests so as not to bias each class. One of the best ways is to split data into training and testing using a predefined ratio such as 80% for training and 20% for testing. However, the specific rate can be adjusted according to the size and characteristics of the data.

- **Select SVM Classifier**

SVM is a powerful and widely used classification system in machine learning and is known for its ability to handle high data and nonlinear decision boundaries. SVM classifiers are particularly useful for facial recognition due to their ability to manage complex processing environments. When choosing a SVM classifier, it is important to consider things like

dataset size, dimensionality, and complexity of the classification problem. SVM classifiers are well suited for small to medium datasets with medium to high precision.

- **Choosing the Right Kernel**

In the SVM, the kernel function is responsible for transforming the input data into a higher-level domain where classes can be separated from the plane. Core selection can affect the performance of the SVM classifier and its ability to handle different types of data. There are many cores used in SVM:

- **Linear Core:** Linear core is considered a decision boundary in the original space. It is suitable for data where the data is linearly separated, or the number of features is too large.
- **Polynomial Kernels:** Polynomial kernels use polynomial functions to map data to high-dimensional space. It can capture the relationship and be useful when the decision boundary is non-linear.
- **Radial Based Function (RBF) Kernels:** RBF kernels are widely used because of their ability to handle nonlinear data. Maps data to an infinite dimensional space suitable for situations where there is no prior knowledge of the underlying structure of the data.

The choice of kernel depends on the characteristics of the data and the complexity of the classification problem. We chose RBF as our kernel because it was the best fit for our model and gave the least absolute error and maximum accuracy. Refer to Appendix A3.3 for the model training code.

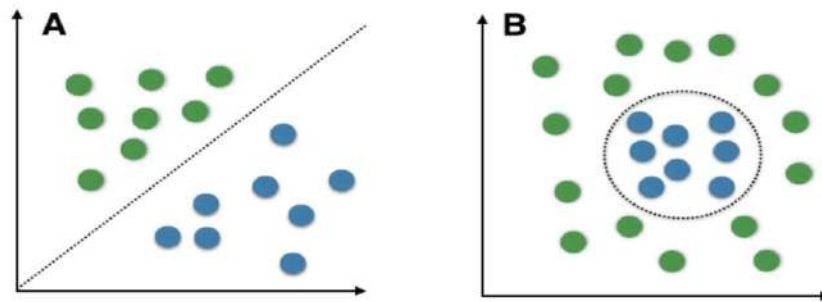


Figure 2.3.13 A: Linearly Separable Data B: Non-Linearly Separable Data

2.3.4 Development and Integration of Web Portal

Creation of Web portal

A web portal was created to show the attendance of students. The front-end development phase includes the use of appropriate technologies and patterns, such as HTML, CSS, JavaScript and back-end programming languages such as PHP to link with RDS and create search, delete and insert students and courses options on the portal.



Figure 2.3.14 Login page of Web portal



Figure 2.3.15 Home Page of Web Portal

Creating EC2 Instance on AWS

An EC2 instance named website was created to provide maximum scalability and availability to the website. Following steps were followed for creating EC2 instance.

- I. On the EC2 console the “Launch Instance” button was selected to start the creation process.
- II. Name (website) and OS Image (Ubuntu 16.04 LTS) was selected for the EC2 instance.
- III. An Instant type of t2. micro was selected that met our requirements.
- IV. Next security groups were configured to allow inbound traffic on port 80 (HTTP) , port 443 (HTTPS) and port 22 (SSH).
- V. Before finalizing the launch, a key pair for SSH access was generated to securely connect with Amazon EC2.

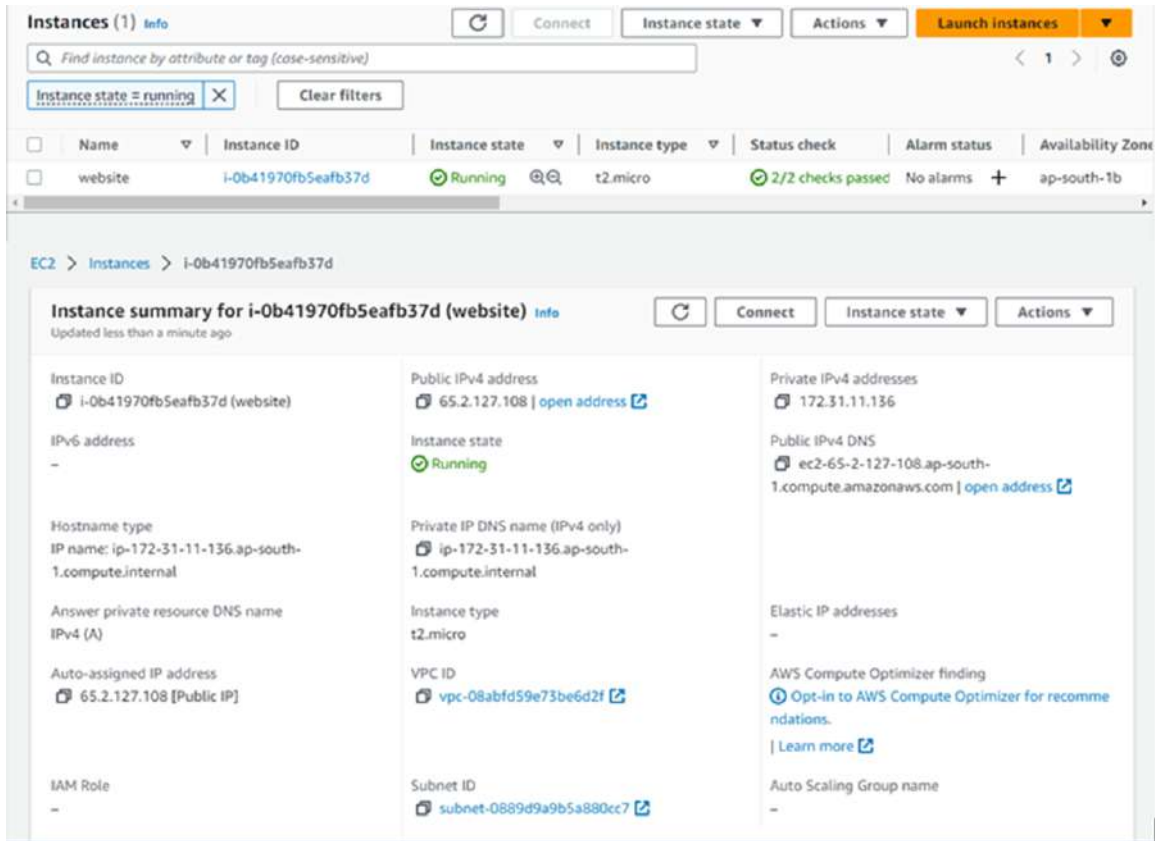


Figure 2.3.16 Creating and Configuring EC2 Instance

Deploying Website on AWS

To deploy our website on EC2 the following steps were taken.

- I. Connection was made to the EC2 instance using SSH. Tool called PuTTY was used to establish an SSH connection as shown in figure below.

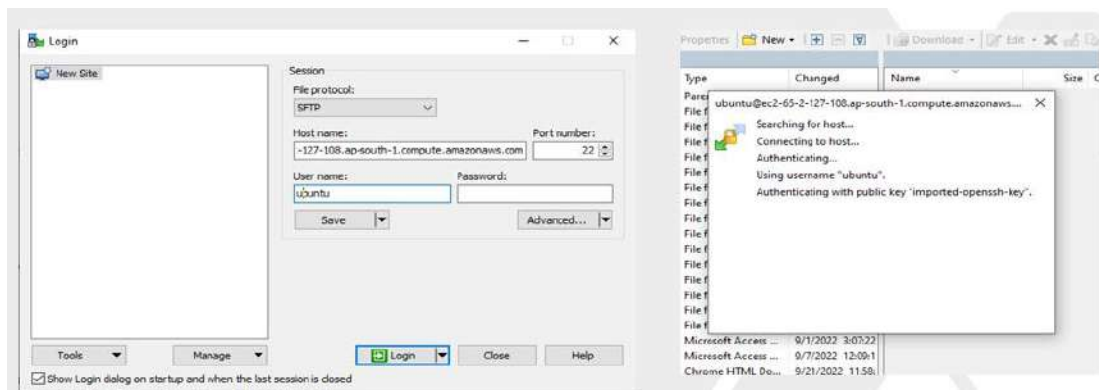


Figure 2.3.17 Connecting to EC2 Instance

- II. Necessary software stack on the EC2 instance was installed and configured. This includes a web server Apache, Php and MYSQL.

Chapter 2: Solution Design & Implementation

- III. Website files were then uploaded on the EC2 instance. For this Purpose, secure file transfer protocols like WinSCP (Windows Secure Copy) were used to transfer files from my local machine to the EC2 instance.
- IV. Our website files were placed in the /var/www/html directory. Figure 2.3.18 shows the transferred files to EC2.

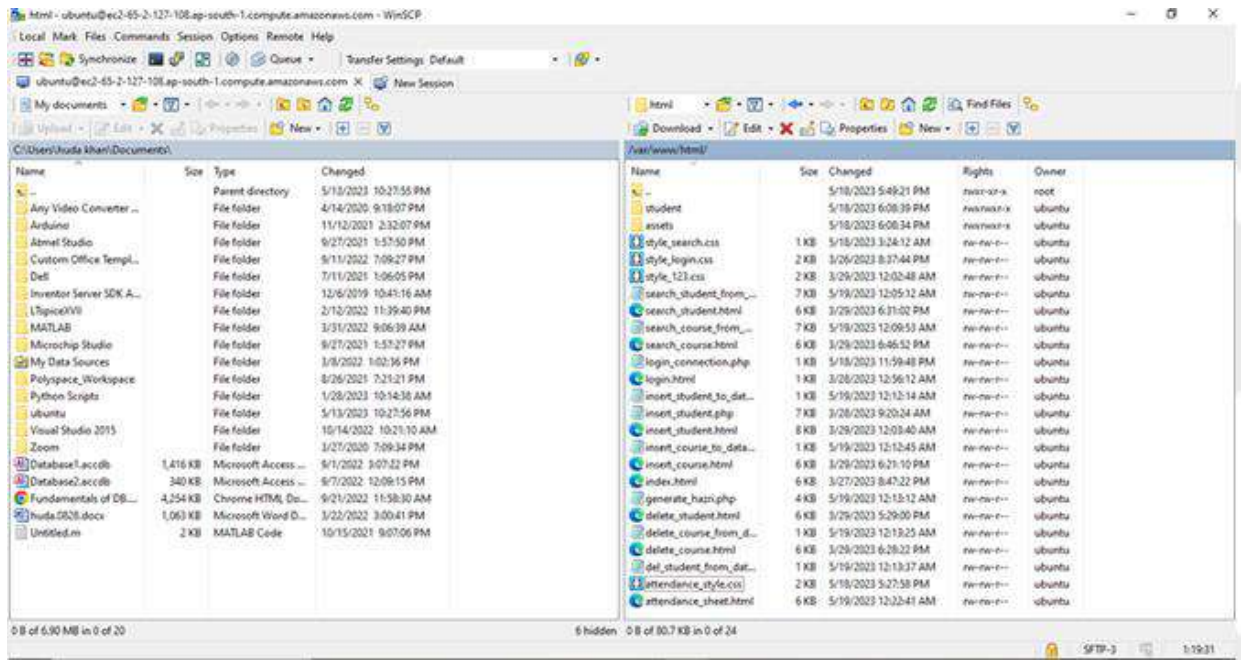


Figure 2.3.18 Deploying website on EC2 instance

Integration of web portal with RDS

To connect EC2 and RDS always ensure that the group security policies of the RDS instance and the EC2 instance allow proper network connections between them. This includes configuring inbound rules for the security team to allow appropriate ports and protocols. EC2's security groups were configured to allow access from the RDS instance i.e inbound traffic was allowed on port for the database engine (port 3306 for MySQL) as shown in figure. Website was tested by accessing the EC2 instance's public IP address (65.2.127.108) in the browser

Chapter 2: Solution Design & Implementation

Security groups

- sg-0753aa6c6f647beff (ec2-rds-1)
- sg-05fd33505a680313d (launch-wizard-1)

▼ Inbound rules

Security group rule ID	Port range	Protocol	Source	Security groups
sgr-06623cd9c7fdd9cac	80	TCP	0.0.0.0/0	launch-wizard-1
sgr-059cd2929c55647db	3306	TCP	0.0.0.0/0	launch-wizard-1
sgr-035398b128092a9db	443	TCP	0.0.0.0/0	launch-wizard-1
sgr-026f3ad3093daa877	22	TCP	0.0.0.0/0	launch-wizard-1

Figure 2.3.19 Integrating Web portal with RDS

2.4 Hardware Implementation

The hardware used in our project is an ESP32 camera and FTDI Programmer.

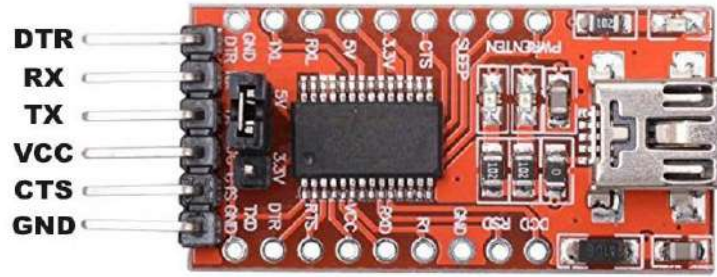
2.4.1. ESP32 camera

ESP32 camera is a small size and low power consuming camera module based on ESP32. It comes with an OV2640 camera sensor which is 2 MP. It requires an input of 5V to operate.



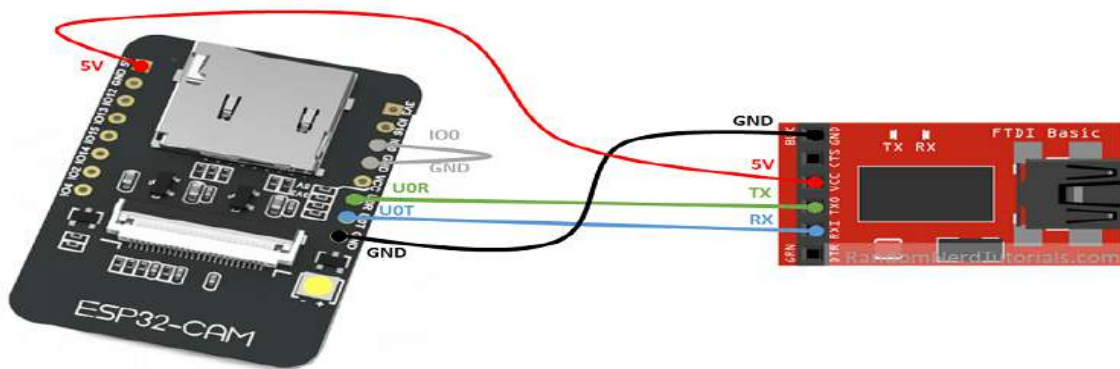
2.4.2. FTDI Programmer

FTDI programmer is used to interface ESP32 camera through USB cable. It converts USB signal to UART and also supplies power to ESP32 camera.



2.4.3. ESP32 camera with FTDI Programmer

FTDI Programmer and ESP32 camera serially communicate with each other using UART. The TX pin of FTDI connects with the UOR of the ESP32 camera and vice versa. When we need to burn code, the IO0 pin is connected to ground and when the code is burnt it is removed so that ESP32 camera can operate the burnt code. Schematic diagram is below:



Chapter 3 Implementation Plan

3.1 Project Deliverables

FYP 1	FYP 2
<p><u>Deliverables:</u></p> <p>D1.1 Development of Database on AWS (Achieved) D1.2 Preparation of Dataset (Achieved) D1.3 Deployment of Camera Modules (Achieved)</p> <p><u>Milestones:</u></p> <ul style="list-style-type: none"> • <i>Creating and Managing Database using RDS service on Amazon [D1.1]</i> • <i>Capturing pictures of different students and preparing the dataset [D1.2]</i> • <i>Interfacing of Camera Modules with Arduino IDE [D1.3]</i> • <i>Interfacing of Camera Modules with AWS [D1.4]</i> 	<p><u>Deliverables:</u></p> <p>D2.1 Machine Learning Model (Achieved). D2.2 Frontend Development (Achieved). D2.3 Backend Development.(Achieved). D2.4 Deployment of Model.(Achieved). D2.5 Marking Attendance. (40%)</p> <p><u>Milestones:</u></p> <ul style="list-style-type: none"> • <i>Choosing the right model for face recognition. [D2.1]</i> • <i>Training and testing of ML Model. [D2.1]</i> • <i>Design of UI/UX. [D2.2]</i> • <i>Integrating database with frontend and AWS. [D2.3]</i> • <i>Choosing the right ML service for model and learning how it works. [D2.4]</i>

3.2 Gantt Chart and Project Schedule

3.2.1: FYP 1

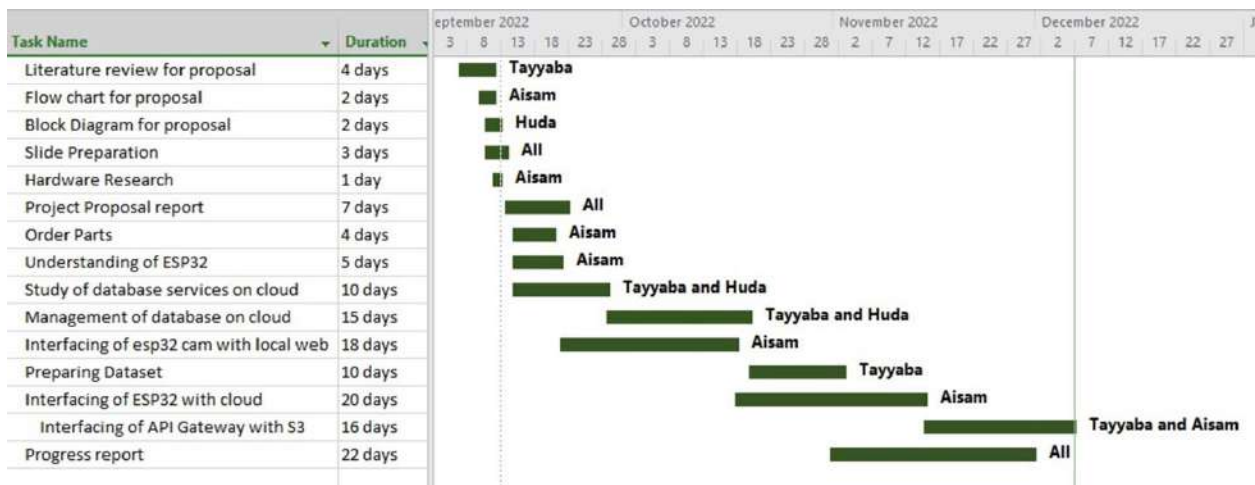


Figure 3.2.1 Gantt Chart FYP1

Chapter 3: Implementation Plan

3.2.2: FYP 2

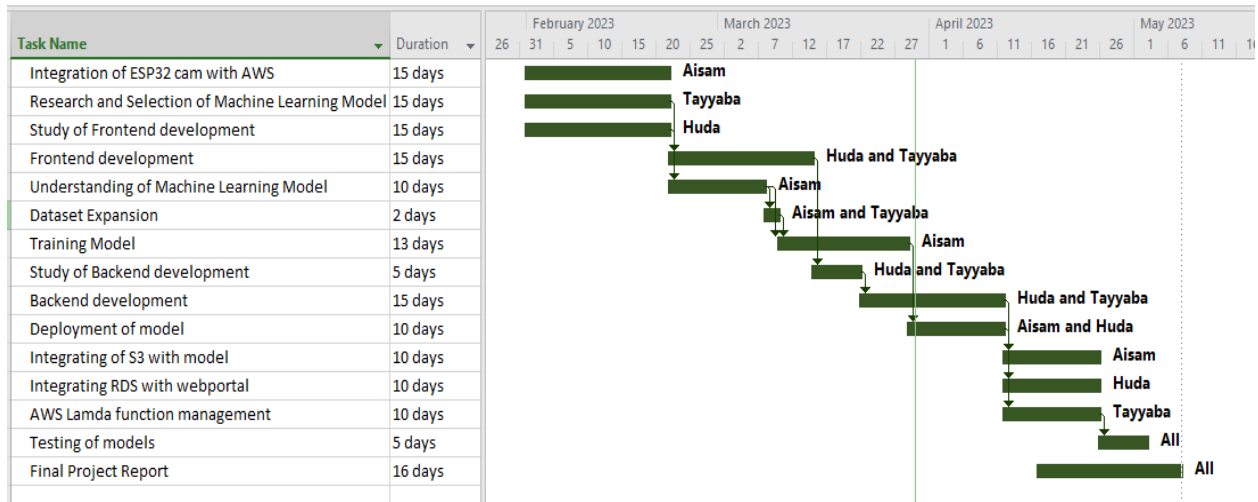
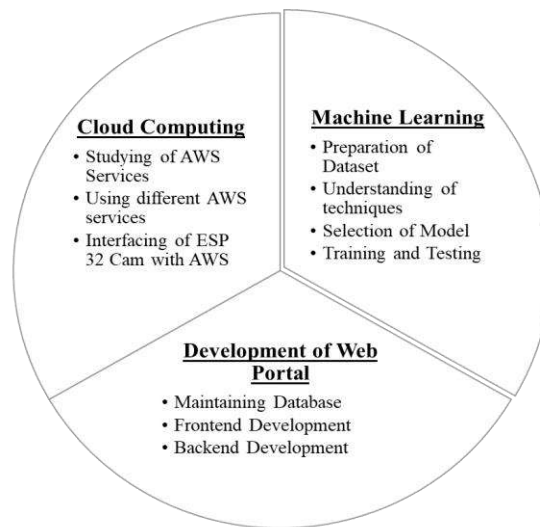


Figure 3.2.2 Timeline FYP2

3.3 Major Work Distribution



3.4 Budget

A detailed budget plan is attached in Appendix-B which covers the cost for import of hardware, price of each individual components and services used. The budget may vary in future according to the government taxation policy and inflation.

3.5 Meetings

To ensure efficient workflow and to report timely progress we had several meetings with our supervisor on-campus. The minutes of the meetings are attached in Appendix-C as a reference.

Chapter 4 Result and Recommendations

To implement our prototype, we had:

- a. Four ESP32 Cameras deployed in our classrooms.
- b. Images of all the students sitting collectively in class.

For testing in university, pictures were taken in the classroom environment. Those pictures were then passed to the trained model and various model evaluation parameters were measured and attendance of students was marked on the web portal accordingly. Details are mentioned below:

4.1 Results of Trained Model

At first when we calculated the Model Accuracy it was found that the model was overfitting as the accuracy was too high for unseen data. To overcome this issue, we implemented various algorithms listed below:

- In machine learning, overfitting occurs when a model performs well on training data but fails on unseen data. K-fold cross-validation is a widely used technique to evaluate and reduce the over-performance of face recognition models. This report discusses the concept of K-fold cross-validation and its application to measure model overfitting-fold cross-validation is a resampling operation that divides data into K subsets or multiples of equal proportions. The model is then trained and evaluated K times, each fold is used once as a validation set, and the rest is used for training. This method allows comprehensive evaluation of the performance of the model on different datasets.

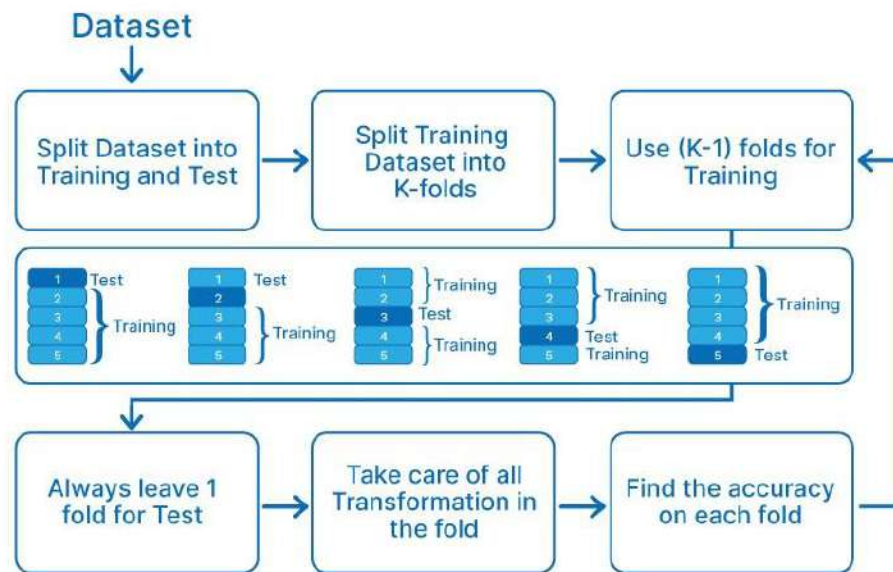


Figure 4.1 K Fold Cross Validation

- Regularization was also performed as it is a powerful technique to combat overfitting in machine learning models. It helps prevent the model from overfitting the training data by introducing unsatisfied or unsuspected penalty conditions. Regularization aims to find a balance between fitting the training data well and controlling unobserved data. There are

two types of regularization functions used to solve overfitting problems: L1 regularization (Lasso) and L2 regularization (Ridge) by which the parameter (λ) is tuned.

- This parameter (λ) determines the balance between the function of fitting and editing training data. Smaller lambda values allow the model to better fit the training data, potentially reducing the risk of overfitting. On the other hand, a larger value of λ increases the performance constant, making the model simpler and less distracting, but may compromise the quality of fitting data training.

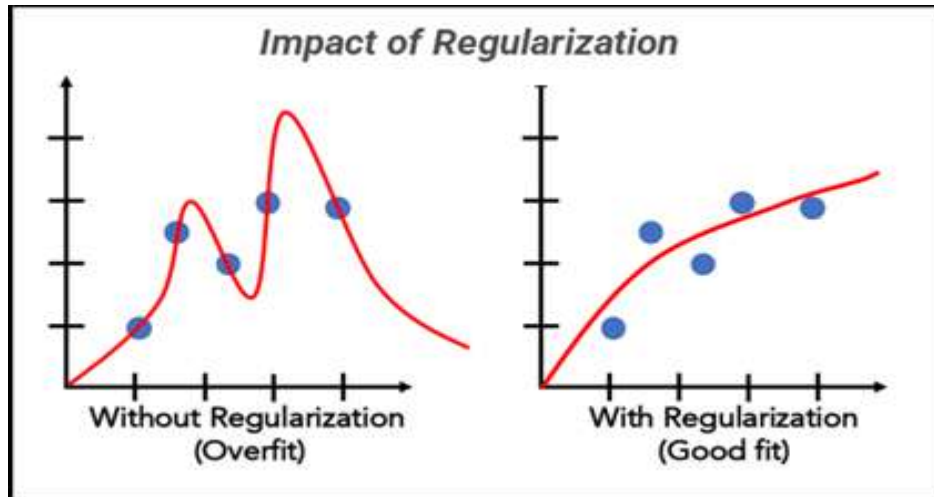


Figure 4.2 Impact of Regularization

- These graphs were also made to check the model performance. As seen, training error also oscillates at the negative axis while test error is only on the positive side which tells its less comparatively.

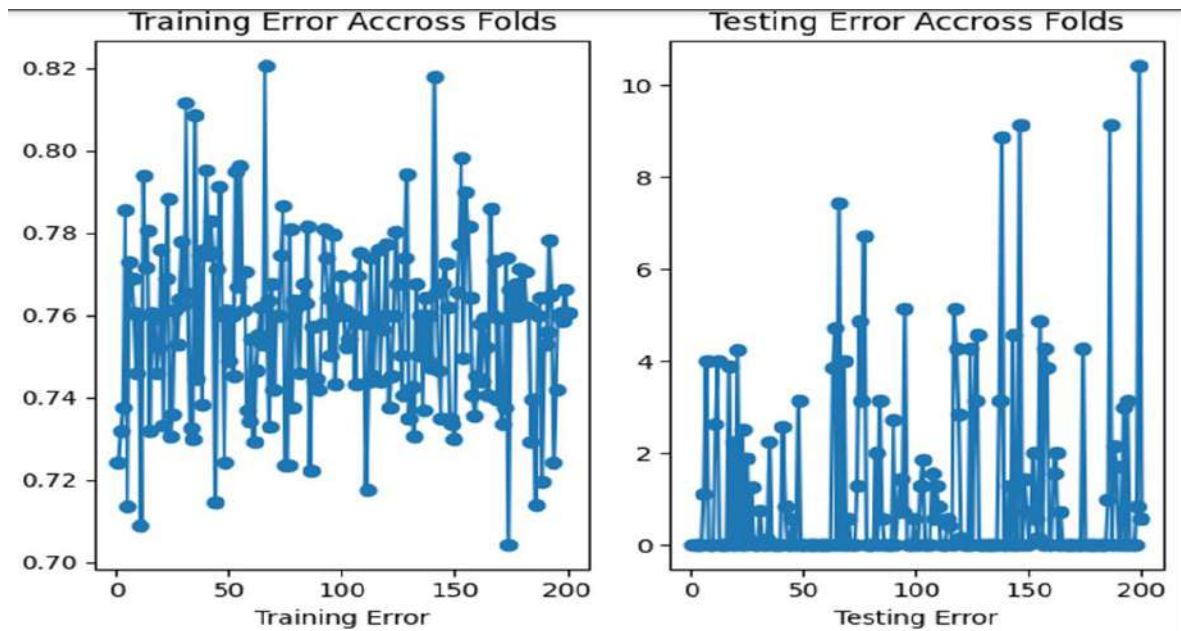


Figure 4.3 Mean Absolute Errors of train and test data.

Results after eliminating overfitting:

```
[INFO] Performing SVM Training .....  
Average Precision: 0.8869890873015872  
Average Recall: 0.8886607142857144  
Average F1-score: 0.8868869047619047  
Average Accuracy: 0.9301785714285714  
Classification Report (Average):  
Mean MAE: 1.0535714285714286
```

As it is clear from this classification report that overfitting is eliminated, and all the other parameters also fit according to the model resulting in it to work efficiently.

4.2 Results on Web Portal

To display the result on the Web Portal, the RDS instance named database-1 was connected to the web portal using the instance's endpoint and master credentials. Refer to Appendix A4 for the connection code. Machine Learning Model extracted the real-time images from s3 bucket and gave the prediction i-e which students are present in class. Further back-end query was generated to send these predictions of the model to the web portal. Figure 4.4 shows the marked attendance of students that were enrolled in the Programming fundamentals course. Other than this the web portal contains the options of inserting, searching and deleting courses and students.

Roll No	Student Name	Course Number	Attendance
19I-0817	Ahsan Basharat	PF0019	P
19I-0751	Maryam Naveed	PF0019	P
19I-0762	Khuzama Ahmed	PF0019	A
19I-0765	Wardah Malik	PF0019	A
19I-0766	Bast Shabbir	PF0019	P
19I-0772	zeeshan Ansar	PF0019	P
19I-0796	Muhammad Anj	PF0019	A
19I-0807	Muataza Amjad	PF0019	P
19I-0813	Rabab Hussain Rizvi	PF0019	A

Figure 4.4 Marking Attendance on Web Portal

4.3 Conclusions

Automated attendance system is for schools, colleges, and universities where an institution can use ESP32 cam modules to take pictures of the class and mark attendance automatically. The images are sent to the S3 bucket where we have our machine learning models already trained on the dataset that will take the pictures sent by the module, apply the Machine Learning Algorithms, and send the list of students present on Web Portal. This report comprehensively tells us about the methods we will use and the processes we will follow to achieve our product.

4.4 Recommendations / Future Work

Despite that, the prototype is quite promising, and its accuracy is commendable, however it still lacks in various aspects. Firstly, the system needs to have a large amount of dataset to get more accurate results. Secondly, features such as low light detection and face detection in mask can also be added to improve the efficiency of the product. Moreover, all of this process should be automated in a sense that instead of training all the data manually and applying all the above steps the user can give input image from the portal which could automatically train the ML model and implement it.

Societal and Environmental Impact

It is essential for organizations to automate their attendance recording to ensure the accuracy and efficacy of records. It brings about the following essential benefits for the organization.

- a) Automating attendance eliminates the risk of incorrect data entry and maintains accurate and real-time data. This will help smooth operation and monitoring of the performance and performance of each student. This ultimately creates competition within the organization.
- b) The automated attendance system improves productivity. It removes the need for manual labor and eases down a variety of tasks at the management's end.
- c) Automatic systems use electronic storage for information which reduces paper consumption that leads to less forestation which reduces global warming.
- d) It is cost and power efficient as ESP32 is used which consumes very low energy.

UN Sustainable Development Goals

The Sustainable Development Goals (SDGs), also known as the Global Goals, were adopted by the United Nations in 2015 as a global call to end poverty, protect the planet, and ensure peace and prosperity for all by 2030. They recognize that actions in one area will affect the outcomes of others and that development must balance social, economic and environmental sustainability. Among these 17 SDGs following are the goals that our project satisfies.

- **Quality Education:**

Regular attendance of students in the current education system plays an important role in performance evaluation and quality control. Attendance plays an important role in determining the academic performance of students in educational institutions. The regularity of attendance shows that the students are less likely to engage in destructive activities. Automatic attendance system is less prone to proxies which reduces the risk of school failure and early dropout ensuring the quality education.

- **Responsible consumption and Production method:**

Our project works as a sustainable technology as it does not require use of magnetic cards, instead a facial recognition technique will be used. This technique allows the use of Esp-32 camera for capturing the pictures of class, which is power efficient and provides a higher resolution. Therefore, methods alternative to magnetic cards works as a substitute technology to save energy leading to responsible consumption and production. Also, the components used are recyclable and don't lead to pollution.

- **Climate Action:**

People use about 360,000 pages a year for attendance; this number is alarming as a lot of paper is wasted for short term usage. The project supports electronic data storage; thus reducing paper consumption. Once attendance control is achieved, all information is digitized and stored in a database instead of copying the information. Technology promotes environmental awareness because it can save some forests from encroachment so they can protect against climate change.

- **Life on Hand:**

Automatic attendance involves the reduction of paper usage that leads to less cut down of trees. Greenpeace explains that forests are the most diverse system in the world, containing more than 80 percent of all land animals, plants and insects. By reducing deforestation, this project manages the forest and provides shelter and security to forest-dwelling communities.

Lifelong Learning

Following are the techniques we will be learning and implementing in our project.

- a) **Amazon Web Services (AWS):** Amazon web services will be used to perform different tasks in our project. Some of the services include AWS sage maker, RDS, lambda function etc.
- b) **Development of web portal:** Frontend and Backend of the web portal will be designed to display the attendance of students along with their personal record.
- c) **Java Script & Python:** Different programming languages will be used for web portal development and machine learning.
- d) **MySQL Workbench:** Purpose of using MySQL Workbench is to transmit locally created database on cloud.
- e) **Postman API:** It is used to check the connection between API Gateway and Lambda Function.
- f) **PuTTY:** PuTTY is an open-source terminal emulator and SSH client for Windows used to connect to remote servers using various protocols. PuTTYGen is a useful utility in conjunction with PuTTY to generate and manage encryption keys for SSH and exchange various types of keys.
- g) **WinSCP:** WinSCP is a free and open file transfer program for Windows that allows users to transfer files between local and remote computers using protocols such as FTP, SFTP, SCP and WebDAV. It provides a user-friendly graphical interface with features such as drag-and-drop file transfers, remote file updates, synchronization, scripting, and automation. It is a versatile tool for managing data on remote servers.

Appendix-A: Project Codes**A1 Project Codes (ESP32 camera to AWS)**

```
#include "esp_http_client.h"
#include "esp_camera.h"
#include <WiFi.h>
#include "Arduino.h"
#include "Base64.h"
#include "mbedtls/base64.h"
#include <NTPClient.h>
#include <WiFiUdp.h>

const char* ssid ="Area 51";
const char* password = "14081999";
int capture_interval = 5000; // Microseconds between captures

bool internet_connected = false;
long current_millis;
long last_capture_millis = 0;

// CAMERA_MODEL_AI_THINKER
#define PWDN_GPIO_NUM 32
#define RESET_GPIO_NUM -1
#define XCLK_GPIO_NUM 0
#define SIOD_GPIO_NUM 26
#define SIOC_GPIO_NUM 27
#define Y9_GPIO_NUM 35
#define Y8_GPIO_NUM 34
#define Y7_GPIO_NUM 39
#define Y6_GPIO_NUM 36
#define Y5_GPIO_NUM 21
#define Y4_GPIO_NUM 19
#define Y3_GPIO_NUM 18
#define Y2_GPIO_NUM 5
#define VSYNC_GPIO_NUM 25
#define HREF_GPIO_NUM 23
#define PCLK_GPIO_NUM 22

void setup()
{
  Serial.begin(115200);
  WiFi.mode(WIFI_STA);
  Serial.println();
```

```
Serial.print("Connecting to ");
Serial.println(ssid);
WiFi.begin(ssid, password);
while (WiFi.status() != WL_CONNECTED)
{
  Serial.print(".");
  delay(500);
}
Serial.println("Internet connected");
camera_config_t config;
config.ledc_channel = LEDC_CHANNEL_0;
config.ledc_timer = LEDC_TIMER_0;
config.pin_d0 = Y2_GPIO_NUM;
config.pin_d1 = Y3_GPIO_NUM;
config.pin_d2 = Y4_GPIO_NUM;
config.pin_d3 = Y5_GPIO_NUM;
config.pin_d4 = Y6_GPIO_NUM;
config.pin_d5 = Y7_GPIO_NUM;
config.pin_d6 = Y8_GPIO_NUM;
config.pin_d7 = Y9_GPIO_NUM;
config.pin_xclk = XCLK_GPIO_NUM;
config.pin_pclk = PCLK_GPIO_NUM;
config.pin_vsync = VSYNC_GPIO_NUM;
config.pin_href = HREF_GPIO_NUM;
config.pin_sscb_sda = SIOD_GPIO_NUM;
config.pin_sscb_scl = SIOC_GPIO_NUM;
config.pin_pwdn = PWDN_GPIO_NUM;
config.pin_reset = RESET_GPIO_NUM;
config.xclk_freq_hz = 20000000;
config.pixel_format = PIXFORMAT_JPEG;
//init with high specs to pre-allocate larger buffers
config.frame_size = FRAMESIZE_UXGA;
config.jpeg_quality = 10;
config.fb_count = 4;
config.grab_mode = CAMERA_GRAB_LATEST;
// camera init
esp_err_t err = esp_camera_init(&config);
if (err != ESP_OK)
{
  Serial.printf("Camera init failed with error 0x%x", err);
  return;
}
}

esp_err_t _http_event_handler(esp_http_client_event_t *evt)
{
```

```

switch (evt->event_id)
{
case HTTP_EVENT_ERROR:
    Serial.println("HTTP_EVENT_ERROR");
    break;
case HTTP_EVENT_ON_CONNECTED:
    Serial.println("HTTP_EVENT_ON_CONNECTED");
    break;
case HTTP_EVENT_HEADER_SENT:
    Serial.println("HTTP_EVENT_HEADER_SENT");
    break;
case HTTP_EVENT_ON_HEADER:
    Serial.println();
    Serial.printf("HTTP_EVENT_ON_HEADER, key=%s, value=%s", evt->header_key, evt-
>header_value);
    break;
case HTTP_EVENT_ON_DATA:
    Serial.println();
    Serial.printf("HTTP_EVENT_ON_DATA, len=%d", evt->data_len);
case HTTP_EVENT_ON_FINISH:
    Serial.println("");
    Serial.println("HTTP_EVENT_ON_FINISH");
    break;
case HTTP_EVENT_DISCONNECTED:
    Serial.println("HTTP_EVENT_DISCONNECTED");
    break;
}
return ESP_OK;
}

static esp_err_t take_send_photo()
{
    sensor_t * s = esp_camera_sensor_get();
    s->set_framesize(s,FRAMESIZE_UXGA);
    s->set_brightness(s, 0);    // -2 to 2
    s->set_contrast(s, 0);     // -2 to 2
    s->set_saturation(s, 0);   // -2 to 2
    s->set_special_effect(s, 0); // 0 to 6 (0 - No Effect, 1 - Negative, 2 - Grayscale, 3 - Red Tint, 4
- Green Tint, 5 - Blue Tint, 6 - Sepia)
    s->set_whitebal(s, 1);     // 0 = disable , 1 = enable
    s->set_awb_gain(s, 1);     // 0 = disable , 1 = enable
    s->set_wb_mode(s, 0);      // 0 to 4 - if awb_gain enabled (0 - Auto, 1 - Sunny, 2 - Cloudy, 3
- Office, 4 - Home)
    s->set_exposure_ctrl(s, 1); // 0 = disable , 1 = enable
    s->set_aec2(s, 1);        // 0 = disable , 1 = enable
}

```

Appendix

```
s->set_ae_level(s, 0); // -2 to 2
s->set_aec_value(s, 300); // 0 to 1200
s->set_gain_ctrl(s, 1); // 0 = disable , 1 = enable
s->set_agc_gain(s, 0); // 0 to 30
s->set_gainceiling(s, (gainceiling_t)0); // 0 to 6
s->set_bpc(s, 0); // 0 = disable , 1 = enable
s->set_wpc(s, 1); // 0 = disable , 1 = enable
s->set_raw_gma(s, 1); // 0 = disable , 1 = enable
s->set_lenc(s, 1); // 0 = disable , 1 = enable
s->set_hmirror(s, 0); // 0 = disable , 1 = enable
s->set_vflip(s, 0); // 0 = disable , 1 = enable
s->set_dcw(s, 1); // 0 = disable , 1 = enable
s->set_colorbar(s, 0); // 0 = disable , 1 = enable
delay(5000);
```

```
Serial.println("Taking picture...");
camera_fb_t * fb = NULL;
esp_err_t res = ESP_OK;
```

```
fb = esp_camera_fb_get();
if (!fb)
{
    Serial.println("Camera capture failed");
    return ESP_FAIL;
}
size_t length=fb->len;
Serial.print("length is");
Serial.println(length);
esp_http_client_handle_t http_client;
esp_http_client_config_t config_client = {0};
WiFiUDP ntpUDP;
NTPClient timeClient(ntpUDP);
timeClient.begin();
timeClient.update();
String Time = String(timeClient.getEpochTime());
String MAC = String(WiFi.macAddress());
Serial.print("Time: "); Serial.print(Time);
Serial.print("MAC: "); Serial.print(MAC);
static const char cert[] PROGMEM= R"EOF(
-----BEGIN CERTIFICATE-----
MIIDQTCCAimgAwIBAgITBmyfz5m/jAo54vB4ikPmljZbyjANBgkqhkiG9w0BAQsF
ADA5MQswCQYDVQQGEwJVUzEPMA0GA1UEChMGQW1hem9uMRkwFwYDVQQDE
xBBbWF6
```

```

b24gUm9vdCBDQSAxMB4XDTE1MDUyNjAwMDAwMFOxDTM4MDExNzAwMDAwM
FowOTEL
MAkGA1UEBhMCVVMxZDZANBgNVBAoTBkFtYXpvcjEzMjcGA1UEAxMQQW1hem9
uIFJv
b3QgQ0EgMTCCASlwDQYJKoZIhvcNAQEBBQADggEPADCCAQoCggEBALJ4gHHKeN
Xj
ca9HgFB0fW7Y14h29Jl091ghYPI0hAEvrAlthtOgQ3pOsqTQNroBvo3bSMgHFzZM
9O6II8c+6zf1tRn4SWiw3te5djdYZ6k/oI2peVKVuRF4fn9tBb6dNqcmzU5L/qw
IFAGbHrQgLKm+a/sRxmPUDgH3KKHOVj4utWp+UhnMJbulHheb4mjUcAwhmahRWa6
VOujw5H5SNz/0egwLX0tdHA114gk957EWW67c4cX8jJGKLhD+redqsq08p8kDi1L
93FcXmn/6pUCyziKrlA4b9v7LWIbxcceVOF34GfID5yHI9Y/QCB/IIDegEw+OyQm
jgSubJrIqg0CAwEAANCMCAwDwYDVR0TAQH/BAUwAwEB/zAOBgNVHQ8BAf8EB
AMC
AYYwHQYDVR0OBBYEFIQYziU07LwMIJQuCFmcx7IQTgoIMA0GCSqGSib3DQEBCw
UA
A4IBAQCY8jdaQZChGsV2USggNiMOruYou6r4IK5IpDB/G/wkjUu0yKGX9rbxenDI
U5PMCCjImCXPI6T53iHTfIUJrU6adTrCC2qJeHZERxhI1Bjtt/mv0tadQ1wUs
N+gDS63pYaACbvXy8MWy7Vu33PqUXHeeE6V/Uq2V8viTO96LXFvKWJbYK8U90vv
o/ufQJVtMVT8QtPHRh8jrdkPSHCa2XV4cdFyQzR1bldZwgJcJmApzyMZFO6IQ6XU
5MsI+yMRQ+hDKXJioaldXgjUkK642M4UwtBV8ob2xJNDd2ZhwLnoQdeXeGADbkpy
rqXRfboQnoZsG4q5WTP468SQvvG5
-----END CERTIFICATE-----
)EOF";

```

```

String post_url2 = "https://k5w5stu0h1.execute-api.ap-south-1.amazonaws.com/send/" +
Time; // Location where images are POSTED
char post_url3[post_url2.length() + 1];
post_url2.toCharArray(post_url3, sizeof(post_url3));
config_client.url = post_url3;
config_client.cert_pem= cert;
config_client.event_handler = _http_event_handler;
config_client.method = HTTP_METHOD_POST;
http_client = esp_http_client_init(&config_client);
esp_http_client_set_post_field(http_client, (const char *)fb->buf, fb->len);
esp_http_client_set_header(http_client, "Content-Type", "image/jpg");
esp_err_t err = esp_http_client_perform(http_client);
if (err == ESP_OK)
{
    Serial.print("esp_http_client_get_status_code: ");
    Serial.println(esp_http_client_get_status_code(http_client));
}
esp_http_client_cleanup(http_client);
esp_camera_fb_return(fb);
}

```

```

void loop()
{
  current_millis = millis();
  if (current_millis - last_capture_millis > capture_interval)
  { // Take another picture
    last_capture_millis = millis();
    take_send_photo();
  }
}

```

A2 Project Codes (Creation of Database)

```

CREATE DATABASE attendance;
CREATE TABLE Students (
  Roll_No VARCHAR(8) NOT NULL,
  Student_Name VARCHAR(20) NOT NULL,
  Father_Name VARCHAR(20) NOT NULL,
  Section CHAR(1) NOT NULL,
  Dept_ID CHAR(5) NOT NULL,
  Email VARCHAR(17) NOT NULL,
  Phone_no INT(11) NOT NULL,
  Gender CHAR(1) NOT NULL,
  PRIMARY KEY(Roll_No)
);
ALTER TABLE students MODIFY COLUMN Phone_no Char(11)

INSERT INTO students VALUES ('19I-0828','Huda Shoaib','Shoaib Rabbani
Khan','A','EE','i190828@nu.edu.pk', 03324883888,'F')
INSERT INTO students VALUES ('18I-0817','Ahsan Basharat','Basharat
Hussain','A','EE','i180817@nu.edu.pk',03405022738,'M')
INSERT INTO students VALUES
('19I-0835','Aahad Yousaf Raja','Muhammad
Basharat','A','EE','i190835@nu.edu.pk',03335510956,'M'),
('19I-0807','Mustafa Amjad','Syed Ali Amjad','A','EE','i190807@nu.edu.pk',03370648886,'M'),
('19I-0772','zeeshan Ansar','Ansar Mehmood','A','EE','i190772@nu.edu.pk',03348634492, 'M'),
('19I-0847','Aisam Irshad','Muhammad Irsad','A','EE','i190847@nu.edu.pk',03335223378,'M'),
('19I-0851','Tayyaba Azam','Azam Khan','A','EE','i190851@nu.edu.pk',03055900606, 'F'),
('19I-0842','Subata Khan','Masroor Ahmad Khan','A','EE','i190842@nu.edu.pk',03351338380,
'F')
INSERT INTO students VALUES
('19I-0755','Hamza Nisar','Nisar Ahmad','A','EE','i190755@nu.edu.pk',03335394493,'M'),

```

Appendix

('19I-0766','Basit Shabbir','Shabbir Hussain','A','EE','i190766@nu.edu.pk',03071229885,'M'),
('19I-0850','Zayan Shafi','Zulfiqar Ali','A','EE','i190850@nu.edu.pk',03155718001,'M'),
('19I-0852','Umair khan','Muhammad Ali
Khan','A','EE','i190852@nu.edu.pk',03335281115,'M'),
('19I-0751','Maryam Naveed','Naveed Zaffar','A','EE','i190751@nu.edu.pk',03005178644,'F')
INSERT INTO students VALUES
('19I-0762','Khuzaima Ahmed','Ahmed khalil','A','EE','i190762@nu.edu.pk',03145988957,'F'),
('19I-0765','Wardah Malik','Usman Malik','A','EE','i190765@nu.edu.pk',03315579993,'F'),
('19I-0813','Rabab Hussain Rizvi','Hussain Rizvi','A','EE','i190813@nu.edu.pk',03349990992,
'F'),
('19I-0881','Hassan Khan','Sheryar Khan','A','EE','i190881@nu.edu.pk',0331064886,'M'),
('19I-0892','Mauzzama Aslam','Muhammad Aslam
Khan','A','EE','i190892@nu.edu.pk',03326269666,'F'),
('19I-0904','Abdullah Zahid','Zahid Sheikh','A','EE','i190904@nu.edu.pk',03051110088,'M'),
('20I-2313','Shiza Jamil','Jamil Ahmad','A','EE','i202313@nu.edu.pk',03337694553,'F')
INSERT INTO students VALUES
('19I-0795','Muhammad Arij','Akmal khan ','A','EE','i190795@nu.edu.pk',03145900475,'M'),
('20I-1009','Samra Shahzad','Shahzad Saleem','A','EE','i201009@nu.edu.pk',03004479923,'F'),
('20I-1027','Neha Rizwan','Rizwan Malik','A','EE','i201009@nu.edu.pk',03441112351,'F'),
('20I-1028','Abeera Abassi','Taha Abassi','A','EE','i201028@nu.edu.pk',0300064997,'F'),
('20I-1029','Naqi Aslam','Aslam Butt','A','EE','i201029@nu.edu.pk',03316289333,'M'),
('20I-1034','Hadi Shahzad','Shahzad Sheikh','A','EE','i201034@nu.edu.pk',03052340081,'M'),
('20I-2757','Sultan Khan','Khizr Khan','A','EE','i202575@nu.edu.pk',03315004443,'M')
INSERT INTO students VALUES
('21I-0905','Muhammad Abdullah','Ayan khan
,','A','EE','i210905@nu.edu.pk',03005910481,'M'),
('21I-0946','Zohaib Sadat','Sadat Saleem','A','EE','i210946@nu.edu.pk',03011179921,'M'),
('21I-1001','Adil Mubashir','Mubashir Malik','A','EE','i211001@nu.edu.pk',03001101241,'M'),
('21I-1004','Zain Malik','Azher Malik','A','EE','i211004@nu.edu.pk',0321464492,'M'),
('21I-1533','Minahil Irsahad','Irshad Butt','A','EE','i211533@nu.edu.pk',03006289133,'F'),
('21I-2491','Arkam khan','Minhas Khan','A','EE','i212491@nu.edu.pk',03212340281,'M'),
('21I-2995','Maham Jamil','Jamil Khawaja','A','EE','i212995@nu.edu.pk',03325109943,'F')
INSERT INTO students VALUES
('22I-1794','Feroza Khan','Mustafa khan ','A','EE','i221794@nu.edu.pk',03225910483,'F'),
('22I-1835','Fatima Baig','Saleem Baig','A','EE','i221835@nu.edu.pk',03001179122,'F'),
('22I-1841','Barera Fatima','Furqan Malik','A','EE','i221841@nu.edu.pk',03323431534,'F'),
('22I-2198','Eeshah Zulfikar','Zulfiqae Malik','A','EE','i222198@nu.edu.pk',0300464396,'F'),
('22I-2215','Laiba Rehan','Rehan Meer','A','EE','i222215@nu.edu.pk',03014639033,'F'),
('29I-1054','Abdul Rehman','Rehman Khan','A','EE','i291054@nu.edu.pk',03002340281,'M')

Appendix

```
CREATE TABLE Department (  
    Dept_ID CHAR(5) NOT NULL,  
    Dept_Name VARCHAR(20) NOT NULL,  
    Dept_block CHAR(1) NOT NULL,  
    Total_Floors INT(11) NOT NULL,  
    PRIMARY KEY(Dept_ID),  
    CONSTRAINT STUDENT_DEPT_FK  
    FOREIGN KEY (Dept_ID) REFERENCES students(Dept_ID)  
    ON DELETE CASCADE ON UPDATE CASCADE  
);  
ALTER TABLE department MODIFY COLUMN Dept_Name VARCHAR(40);  
  
INSERT INTO department VALUES  
( 'EE','Electrical Engineering','B',2),  
( 'CS','Computer Science','C',6),  
( 'BBA','Bachelor of Business Administration','A',2),  
( 'AnF','Accounting and Finance','A',2),  
( 'Robo','Robotics','B',2),  
( 'DS','Data Science','C',6),  
( 'AI','Artificial Intelligence','C',6),  
( 'SE','Software Engineering','C',6)
```

```
CREATE TABLE Teachers(  
    Teacher_Name VARCHAR(20) NOT NULL,  
    Teacher_ID VARCHAR(10) NOT NULL,  
    Phone_no INT(11) NOT NULL,  
    Email VARCHAR(25) NOT NULL,  
    Gender CHAR(1) NOT NULL,  
    Dept_ID CHAR(5) NOT NULL,  
    Course_No VARCHAR(15) NOT NULL,  
    PRIMARY KEY(TEACHER_ID),  
    CONSTRAINT Teacher_DEPT_FK FOREIGN KEY(Dept_ID) REFERENCES  
department(Dept_ID)  
    ON DELETE CASCADE ON UPDATE CASCADE  
);  
  
ALTER TABLE teachers MODIFY COLUMN Teacher_ID INT(5);  
ALTER TABLE teachers MODIFY COLUMN Phone_no Char(11);  
  
INSERT INTO teachers VALUES  
( 'Shehzad Ahmad',34107,03215667898,'shehzad.ahmad@nu.edu.pk','M','EE','PF0019'),  
( 'Arshad Hassan',34110,03450652254,'arshad.hassan@nu.edu.pk','M','EE','PF0019'),
```

Appendix

```
('Mukhtar Ullah',34097,03469546289,'mukhtar.ullah@nu.edu.pk','M','EE','BME007'),  
( 'Moomal Bukhari',34159,03008567488,'moomal.bukhari@nu.edu.pk','F','EE','IPC450'),  
( 'Umer Baig',10864,03007866393,'omer.baig@nu.edu.pk','M','DS','DSA872'),  
( 'Sidra Tariq',10223,03224883237,'sidra.tariq@nu.edu.pk','F','BBA','ACC657'),  
( 'Ibrar Khan',34230,03241117832,'ibrar.khan@nu.edu.pk','M','EE','DB0768')
```

```
CREATE TABLE Courses (  
    Course_No VARCHAR(15) NOT NULL,  
    Course_Name VARCHAR(20) NOT NULL,  
    Teacher_ID VARCHAR(10) NOT NULL,  
    Dept_ID CHAR(5) NOT NULL,  
    PRIMARY KEY(Course_No),  
    CONSTRAINT Courses_Teacher_FK FOREIGN KEY(Teacher_ID) REFERENCES  
teachers(Teacher_ID)  
    ON DELETE CASCADE ON UPDATE CASCADE,  
    CONSTRAINT Courses_dept_FK FOREIGN KEY(Course_No) REFERENCES  
teachers(Course_No)  
    ON DELETE CASCADE ON UPDATE CASCADE  
);  
  
ALTER TABLE courses MODIFY COLUMN Course_Name VARCHAR(40);  
ALTER TABLE courses DROP FOREIGN KEY Courses_dept_FK ;  
ALTER TABLE courses ADD FOREIGN KEY Courses_dept_FK (Dept_ID) REFERENCES  
department(Dept_ID)  
    ON DELETE CASCADE ON UPDATE CASCADE;  
  
INSERT INTO courses VALUES  
( 'PF0019','Programming Fundamental',34107,'EE'),  
( 'EMT125','Electromagnetic Theory',34110,'EE'),  
( 'BME007','Basic Mechanical Engineering',34097,'EE'),  
( 'IPC450','Interprocess Communication',34159,'EE'),  
( 'DSA872','Datastructure and Algorithm',10864,'DS'),  
( 'ACC657','Accounting',10223,'BBA'),  
( 'DB0768','Database',34230,'EE')
```

```
CREATE TABLE Room_Schedule(  
    Course_No VARCHAR(15) NOT NULL,  
    Section CHAR(1) NOT NULL,  
    Teacher_ID VARCHAR(10) NOT NULL,  
    Day VARCHAR(9) NOT NULL,  
    start_time Time NOT NULL,  
    end_time Time NOT NULL,  
  
    CONSTRAINT sched_course_FK FOREIGN KEY(Course_No)REFERENCES  
courses(Course_No)
```

```

ON DELETE CASCADE ON UPDATE CASCADE

```

```

);
INSERT INTO room_schedule VALUES
('PF0019','A',34107,'Monday','13:00:00','14:00:00')
INSERT INTO room_schedule VALUES
('EMT125','A',34110,'Monday','14:05:00','15:05:00'),
('BME007','A',34097,'Tuesday','09:00:00','10:00:00'),
('IPC450','A',34159,'Tuesday','10:05:00','11:05:00'),
('DSA872','A',10864,'Wednesday','9:00:00','10:00:00'),
('ACC657','A',10223,'Wednesday','13:00:00','14:00:00'),
('DB0768','A',34230,'Thursday','9:00:00','10:00:00')

```

```

CREATE TABLE Enrollment (
  Roll_No VARCHAR(8) NOT NULL,
  Course_No VARCHAR(15) NOT NULL,
  PRIMARY KEY(Course_No, Roll_No),

  CONSTRAINT Enroll_Students_FK FOREIGN KEY(Roll_No) REFERENCES
students(Roll_No)
  ON DELETE CASCADE ON UPDATE CASCADE,
  CONSTRAINT Enroll_Courses_FK FOREIGN KEY(Course_No) REFERENCES
courses(Course_No)
  ON DELETE CASCADE ON UPDATE CASCADE
);

```

```

INSERT INTO enrollment
VALUES('19I-0828', 'PF0019'),('18I-0817', 'PF0019'),('19I-0835', 'PF0019'),('19I-0842',
'PF0019'),('19I-0807', 'PF0019'),('19I-0772', 'PF0019'),('19I-0847', 'PF0019'),('19I-0851',
'PF0019'),('19I-0775', 'PF0019'),('19I-0766', 'PF0019'),('19I-0850', 'PF0019'),('19I-0795',
'PF0019'),('19I-0852', 'PF0019'),('19I-0751', 'PF0019'),('19I-0828', 'DB0768'),('18I-0817',
'DB0768'),('19I-0835', 'DB0768'),('19I-0842', 'DB0768'),('19I-0807', 'IPC450'),('19I-0772',
'IPC450'),('19I-0847', 'IPC450'),('19I-0851', 'IPC450'),('19I-0775', 'IPC450'),('19I-0766',
'BME007'),('19I-0850', 'BME007'),('19I-0795', 'BME007'),('19I-0875', 'BME007'),('19I-0751',
'BME007')
,('19I-0762', 'PF0019'),('19I-0765', 'PF0019'),('19I-0813', 'PF0019'),('19I-0881', 'PF0019'),('19I-
0892', 'PF0019'),('19I-904', 'PF0019'),('20I-2313', 'PF0019'),('19I-0762', 'EMT125'),('19I-0765',
'EMT125'),('19I-0813', 'EMT125'),('19I-0881', 'DSA872')
INSERT INTO enrollment
VALUES('19I-0795', 'PF0019'),('20I-1009', 'PF0019'),('20I-1027', 'PF0019'),('20I-1029',
'PF0019'),('20I-1028', 'PF0019'),
('20I-1034', 'PF0019'),('20I-2757', 'PF0019'),('21I-0905', 'PF0019'),('21I-0946', 'PF0019'),('21I-
1001', 'PF0019'),('21I-1004', 'PF0019'),('21I-2491', 'PF0019'),('21I-2995', 'PF0019'),('22I-1794',
'PF0019'),('22I-1835', 'PF0019'),('22I-1841', 'PF0019'),
('22I-2198', 'PF0019'),('22I-2215', 'PF0019'),('29I-1054', 'PF0019')

```

Appendix

```
DROP TABLE IF EXISTS `loginform`;
CREATE TABLE IF NOT EXISTS `loginform` (
  `ID` int(11) NOT NULL AUTO_INCREMENT,
  `User` varchar(50) NOT NULL,
  `Pass` varchar(50) NOT NULL,
  PRIMARY KEY (`ID`)
) ENGINE=MyISAM AUTO_INCREMENT=2 DEFAULT CHARSET=latin1;

--
-- Dumping data for table `loginform`
--

INSERT INTO `loginform` (`ID`, `User`, `Pass`) VALUES
(1, 'admin', 'fast123');

CREATE TABLE students_attendance (
  Roll_No VARCHAR(8) NOT NULL,
  Student_Name VARCHAR(50) NOT NULL,
  Course_No VARCHAR(15) NOT NULL,
  Attendance VARCHAR(10) NOT NULL,
  CONSTRAINT attendance_student_FK FOREIGN KEY(Roll_No) REFERENCES
students(Roll_No),
  CONSTRAINT courses_student_FK FOREIGN KEY(Course_NO) REFERENCES
courses(Course_No)
);
```

A3 Project Codes (Machine Learning)

A3.1 Project Codes (Machine Learning(Data preprocessing))

```
Mounting the drive
"""

from google.colab import drive
drive.mount('/content/drive')

"""Face Extraction (MTCNN)"""

!pip install mtcnn

"""Importing all libraries"""

import cv2 as cv
import os
import numpy as np
```

```

import tensorflow as tf
import matplotlib.pyplot as plt
os.environ['TF_CPP_MIN_LOG_LEVEL'] = '2'
from mtcnn.mtcnn import MTCNN
import imgaug.augmenters as iaa
import matplotlib.pyplot as plt

""""Extracting faces from images and storing cropped images in drive through MTCNN""""

class FACELOADING:
    def __init__(self, directory):
        self.directory = directory
        self.target_size = (160,160)
        self.X = []
        self.Y = []
        self.detector = MTCNN()
        self.augmentation = iaa.Sequential([
            iaa.Fliplr(0.5), # horizontally flip 50% of the images
            iaa.Affine(rotate=(-10, 10)) # rotate images between -10 and 10 degrees
        ])

    def extract_face(self, filename):
        img = cv.imread(filename)
        img = cv.cvtColor(img, cv.COLOR_BGR2RGB)
        x,y,w,h = self.detector.detect_faces(img)[0]['box']
        x,y = abs(x), abs(y)
        face = img[y:y+h, x:x+w]
        face_arr = cv.resize(face, self.target_size)
        return face_arr

    def load_faces(self, dir):
        FACES = []
        for im_name in os.listdir(dir):
            try:
                path = dir + im_name
                single_face = self.extract_face(path)
                augmented_face = self.augmentation.augment_image(single_face) # Apply data
augmentation
                FACES.append(single_face)
                FACES.append(augmented_face)
            except Exception as e:
                pass
        return FACES

```

```

def load_classes(self):
    for sub_dir in os.listdir(self.directory):
        path = self.directory + '/' + sub_dir + '/'
        FACES = self.load_faces(path)
        labels = [sub_dir for _ in range(len(FACES))]
        print(f"Loaded successfully: {len(labels)}")
        self.X.extend(FACES)
        self.Y.extend(labels)

    return np.asarray(self.X), np.asarray(self.Y)

def plot_images(self):
    plt.figure(figsize=(18,16))
    for num,image in enumerate(self.X):
        ncols = 3
        nrows = len(self.Y)//ncols + 1
        plt.subplot(nrows,ncols,num+1)
        plt.imshow(image)
        plt.axis('off')

"""Giving our dataset as input to the above class for face extraction
"""

faceloading= FACELOADING("/content/drive/MyDrive/train1/train")
X,Y=faceloading.load_classes()

"""Showing the loaded images"""

plt.figure(figsize=(70,68))
for num,image in enumerate(X):
    ncols = 20
    nrows = len(Y)//ncols + 1
    plt.subplot(nrows,ncols,num+1)
    plt.imshow(image)
    plt.axis('off')

```

A3.2 Project Codes (Machine Learning (Feature Extraction))

```

"""Feature Extraction through Google Facenet"""

!pip install keras-facenet

from keras_facenet import FaceNet
embedder = FaceNet()

```

```

def get_embedding(face_img):
    face_img = face_img.astype('float32') # 3D(160x160x3)
    face_img = np.expand_dims(face_img, axis=0)
    # 4D (Nonex160x160x3)
    yhat= embedder.embeddings(face_img)
    return yhat[0] # 512D image (1x1x512)

EMBEDDED_X = []

for img in X:
    EMBEDDED_X.append(get_embedding(img))

EMBEDDED_X = np.asarray(EMBEDDED_X)

np.savez_compressed('/content/drive/MyDrive/train1/faces_embeddings_final_done.npz',
EMBEDDED_X, Y)
#print(EMBEDDED_X)

# face recognition part II
#IMPORT
import cv2
import cv2 as cv
import numpy as np
import os
os.environ['TF_CPP_MIN_LOG_LEVEL']='2'
import tensorflow as tf

#INITIALIZE
faces_embeddings =
np.load("/content/drive/MyDrive/train1/faces_embeddings_final_done.npz")
EMBEDDED_X=faces_embeddings['arr_0']
Y=faces_embeddings['arr_1']

```

A3.3 Project Codes (Machine Learning(Model Training))

```

"""SVM Model for recognition"""

from sklearn.preprocessing import LabelEncoder

encoder = LabelEncoder()
encoder.fit(Y)
Y = encoder.transform(Y)

from sklearn.model_selection import KFold, cross_val_score

```

Appendix

```
from sklearn.svm import SVC
from sklearn.metrics import
accuracy_score, confusion_matrix, classification_report, mean_absolute_error, precision_score,
recall_score, f1_score

print("[INFO] Performing SVM Training .....")
# Define the number of folds for k-fold cross validation
k = 200

# Create a k-fold object
kf = KFold(n_splits=k, shuffle=True, random_state=1000)

# Initialize an empty list to store the accuracy scores
accuracy_list = []
precision_list = []
recall_list = []
f1_list = []
all_classes = np.unique(Y)
num_classes = len(all_classes)
list_training_error=[]
list_testing_error=[]
ypreds_train = np.array([])
ypreds_test = np.array([])
y_true = np.array([])

# Initialize an empty confusion matrix with appropriate shape
overall_cm = np.zeros((num_classes, num_classes), dtype=int)

# Loop through each fold
for train_index, test_index in kf.split(EMBEDDED_X):
    # Split the data into train and test sets for this fold
    X_train, X_test = EMBEDDED_X[train_index], EMBEDDED_X[test_index]
    y_train, y_test = Y[train_index], Y[test_index]

    # Initialize a SVM classifier with your choice of hyperparameters
    svm = SVC(kernel='rbf', C=0.2, probability=True)

    # Fit the SVM classifier on the training data
    svm.fit(X_train, y_train)
```


A3.4 Project Codes (Machine Learning (Model Evaluation))

```
# Use the SVM classifier to make predictions on the test data
ypreds_test = svm.predict(X_test)
ypreds_train = svm.predict(X_train)

# Calculate evaluation metrics
precision = precision_score(y_test, ypreds_test, average='macro', zero_division=0)
recall = recall_score(y_test, ypreds_test, average='macro', zero_division=0)
f1 = f1_score(y_test, ypreds_test, average='macro', zero_division=0)
accuracy = accuracy_score(y_test, ypreds_test)

# Append metrics to lists
precision_list.append(precision)
recall_list.append(recall)
f1_list.append(f1)
accuracy_list.append(accuracy)

# Calculate the confusion matrix for this fold
cm = confusion_matrix(y_test, ypreds_test, labels=all_classes)
overall_cm += cm

#Plotting the errors
fold_training_error=mean_absolute_error(y_train,ypreds_train)
fold_testing_error=mean_absolute_error(y_test,ypreds_test)
list_training_error.append(fold_training_error)
list_testing_error.append(fold_testing_error)

# Calculate average metrics
avg_precision = np.mean(precision_list)
avg_recall = np.mean(recall_list)
avg_f1 = np.mean(f1_list)
avg_accuracy = np.mean(accuracy_list)

print("Average Precision:", avg_precision)
print("Average Recall:", avg_recall)
print("Average F1-score:", avg_f1)
print("Average Accuracy:", avg_accuracy)
print("Classification Report (Average):")

# perform cross-validation with MAE as the evaluation metric
mae_scores = -1 * cross_val_score(svm, EMBEDDED_X, Y, cv=kf,
scoring='neg_mean_absolute_error')
```

Appendix

```
# print the mean MAE score across the folds
print('Mean MAE:', mae_scores.mean())

# plot the mean MAE score across the folds
plt.subplot(1,2,1)
plt.plot(range(1,kf.get_n_splits()+1),np.array(list_training_error).ravel(),'o-')
plt.xlabel('Number of Folds')
plt.xlabel('Training Error')
plt.title('Training Error Accross Folds')
plt.tight_layout()

plt.subplot(1,2,2)
plt.plot(range(1,kf.get_n_splits()+1),np.array(list_testing_error).ravel(),'o-')
plt.xlabel('Number of Folds')
plt.xlabel('Testing Error')
plt.title('Testing Error Accross Folds')
plt.tight_layout()

import seaborn as sns
print("[INFO] Confusion Matrix:")
# Plot the overall confusion matrix
sns.heatmap(overall_cm, annot=True, fmt='d', cmap='Blues')
plt.xlabel('Predicted')
plt.ylabel('True')
plt.show()

""""Saving the trained Model""""

import pickle
#save the model
with open('/content/drive/MyDrive/train1/newmodel.pkl','wb') as f:
    pickle.dump(svm,f)
```

A4 Project Codes (Web Portal Connection)

```
$servername = "database-1.cary0mwjpcqv.ap-south-1.rds.amazonaws.com";
$username = "admin";
$password = "huda123huda";
$database = "attendance";
$conn = mysqli_connect($servername, $username, $password, $database);
if (!$conn){
die("Sorry we failed to connect:" .mysqli_connect_error());
}
else{
echo "Connection was successful<br>";
}
```

Appendix B: Budget

S.No	Items	Price/Unit	Unit	Cost
1	ESP 32 camera OV2640	Rs 1500	4	Rs 6000
2	220V to 3V adapter charger	Rs 500	4	Rs 2000
3	ESP 32 cam antenna	Rs.200	4	Rs.800
3	FDTI Programmer	Rs 350	4	Rs 1400
4	Female to male jumper wires	Rs 100	2	Rs 200
5	Female to female jumper wires	Rs 100	1	Rs 100
6	Amazon Web Services	Rs 6000	-	Rs 8000
7	Miscellaneous	Rs 5000	-	Rs 5000
8	Shipping (If any)	Rs 1000	-	Rs 1000
	Total			Rs 24500

Appendix C: Minutes of Meetings

Meeting No	Agenda Item	Action Item	Date
1	Discussion of FYP1 presentation.	<ul style="list-style-type: none"> Expansion of Dataset Discussion on the limitations of the project and its solution. 	09-12-2022
2	Progress Meeting	<ul style="list-style-type: none"> Report on ESP32 camera with AWS Research on ML Model 	23-2-2023
3	Discussion of ML Model	<ul style="list-style-type: none"> Discussion on the difficulties faced in ML Model 	15-3-2023
4	Progress of Mid Evaluation	<ul style="list-style-type: none"> Finalized ML model was shown Checking the Front-end of Web-portal Approval of Mid Evaluation slides 	28-03-2023
5	Discussion of ML Model	<ul style="list-style-type: none"> Discussion on the overfitting of ML Model 	10-04-2023
6	Progress Meeting	<ul style="list-style-type: none"> Checking the model overfitting Showing integration of RDS with Website 	20-04-2023
7	Discussion of AWS Account	<ul style="list-style-type: none"> Discussion on suspension of AWS account 	17-05-2023
8	Progress of Final Evaluation	<ul style="list-style-type: none"> Discussion on unsolved issue Approval of Final Evaluation slides 	18-05-2023
9	Final Presentation Discussion	<ul style="list-style-type: none"> Discussion on Panel remarks 	19-05-2023
10	Final Report	<ul style="list-style-type: none"> Approval of final report 	11-06-2023

References

- [1] Gottfried, M., Chronic Absenteeism and Its Effects on Students' Academic and Socioemotional Outcomes. 2014 Journal of Education for Students Placed at Risk (JESPAR), 19(2), pp.53-75.
- [2] GATETE Marcel, HARUBWIRA Flaubert, "A Development of an Online Student Attendance Management Information System: Case Study "University of Tourism, Technology, and Business Studies",” International Journal of Scientific Research in Computer Science and Engineering, Vol.10, Issue.1, pp.32-48, 2022.
- [3] J. Geeganage et al., "Precise Integrated Contactless Attendance Tracking, Recording and Analyzing System," 2022 19th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications, and Information Technology (ECTI-CON), 2022, pp. 1-4, doi: 10.1109/ECTI- CON54298.2022.9795538.
- [4] F. Cahyono, W. Wirawan and R. Fuad Rachmadi, "Face Recognition System using Facenet Algorithm for Employee Presence," 2020 4th International Conference on Vocational Education and Training (ICOVET), 2020, pp. 57-62, doi: 10.1109/ICOVET50258.2020.9229888.
- [5] Patel, Sakshi & Kumar, Ravi. "Face Recognition based smart attendance system using IOT." 2018 INTERNATIONAL JOURNAL OF COMPUTER SCIENCES AND ENGINEERING, pp. 871-877, doi: 10.26438/ijcse/v6i5.871877.
- [6] V. Perumal, "Face Recognition in Video Streams and its Application in Freedom Fighters Discovery - A Machine Learning Approach," 2020 IEEE International Conference on Machine Learning and Applied Network Technologies (ICMLANT), 2020, pp. 1-5, doi: 10.1109/ICMLANT50963.2020.9355979.
- [7] S. Seal, A. Sen, R. Mukerjee and A. K. Das, "An approach towards development of automated attendance system using face detection and recognition," 2020 11th IEEE Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON), 2020, pp. 0333-0340, doi: 10.1109/IEMCON51383.2020.9284817.