LAB REPORT

Submitted by

Priyansh Agarwal [RA2111033010062]
Harsh Thite [RA2111033010056]
Khushi Padia [RA2111033010063]

Under the Guidance of

Dr.M.Vimaladevi

Assistant Professor(Sr.G), Department of Computing Technologies

In partial satisfaction of the requirements for the degree of

BACHELOR OF TECHNOLOGY
in
COMPUTER SCIENCE ENGINEERING

SCHOOL OF COMPUTING

COLLEGE OF ENGINEERING AND TECHNOLOGY

SRM INSTITUTE OF SCIENCE AND TECHNOLOGY

KATTANKULATHUR - 603203

MAY 2023

COLLEGE OF ENGINEERING & TECHNOLOGY
SRM INSTITUTE OF SCIENCE & TECHNOLOGY
S.R.M. NAGAR, KATTANKULATHUE – 603 203
Chengalpattu District

BONAFIDE CERTIFICATE

Register No. **RA2111033010062, RA2111033010063, RA2111033010056**

Certified to be the bonafide work done by **Priyansh Agarwal , Khushi Padia , Harsh Thite** of II Year/IV Sem B.Tech Degree Course in the **Practical Software Software Engineering and Project Management 18CSC206J** in **SRM INSTITUTE OF SCIENCE AND TECHNOLOGY,** Kattankulathur during the academic year 2022 – 2023.

LAB INCHARGE
Dr. M Vimaladevi
Assistant Professor
Department of Computational Intelligence
SRMIST – KTR.

Head of the Department
Dr. R Annie Uthea
Professor and Head
Department : CINTEL
SRMIST – KTR.

Date :

TABLE OF CONTENTS

ABSTRACT

Face recognition and detection is an important area of research in computer vision and machine learning. The aim of this technology is to automatically identify and locate human faces in images or videos, and then recognize them accurately. This technology has various applications, such as in security systems, biometric authentication, and social media tagging.

The challenges in face recognition and detection include variations in facial features, lighting conditions, pose variations, occlusions, and the presence of facial hair or accessories. To overcome these challenges, advanced machine learning techniques such as deep learning and computer vision algorithms are used.

This paper discusses the state-of-the-art techniques used in face recognition and detection, including face detection algorithms, feature extraction methods, and recognition algorithms. Additionally, the paper explores the datasets used for training and evaluation of the systems, as well as the evaluation metrics used to measure the performance of the systems.

LIST OF FIGURES

PROBLEM STATEMENT

The goal of face recognition and detection is to automatically detect human faces in images or videos and accurately recognize them. This technology has various applications, such as in security systems, biometric authentication, digital image processing, and social media tagging.

The challenges of face recognition and detection include variability in facial features, lighting conditions, pose variations, occlusions, and the presence of facial hair or accessories. Additionally, the system should be able to distinguish between identical twins or people with similar appearances.

To develop an effective face recognition and detection system, it is important to address these challenges by using advanced machine learning techniques, such as deep learning and computer vision algorithms. The system should also be able to handle large datasets, including diverse facial images and demographics, to ensure accuracy and robustness.

Chapter 2 Stakeholders and Process Model

2.1 STAKEHOLDERS

Stakeholders for a face recognition attendance system can vary depending on the target audience and the specific goals of the application. Here are some potential stakeholders:

2.1.1 Users: The primary stakeholders of the face recognition attendance system are the individual users who utilize the app to check the attendance. These can include individuals, school organizations, colleges, offices, etc. where the presence of a person has to be checked regularly.

2.1.2 Application Developers: The development team responsible for creating and maintaining the face recognition attendance system are stakeholders. They are involved in designing and implementing the features, ensuring the application's functionality, and addressing user feedback and issues.

2.1.3 Management and Product Owners: Stakeholders at the management level are responsible for overseeing the application's development, deployment, and overall success. They provide strategic direction, allocate resources, and make decisions regarding the application's roadmap and future enhancements.

2.1.4 Privacy Advocates: They will be concerned about the privacy implications of a face recognition-based attendance system. They will make sure that the data collection is happening in a lawful and secure manner.

2.1.5 Investors: In cases where face recognition attendance system receives funding or investment, the investors become stakeholders. They have a financial interest in the success and growth of the application and may have expectations for returns on their investment.

2.2 PROCESS MODEL

2.2.1 Selection of Methodology: Agile Model

For a simple project such as developing an Face recognition app, either Agile or Waterfall could work. However, based on the nature of the project, we would be using the Agile methodology.

Agile would allow for a faster and more iterative development process, where you can quickly test and validate your game mechanics and make changes as necessary. This would allow for a more flexible and adaptive approach, which is well-suited for a project with a lot of uncertainty or rapid changes in requirements.

In contrast, the Waterfall methodology may be too structured and time- consuming for a project like Spend Spot, and could slow down the development process. The Agile methodology would allow for a more flexible and adaptable approach, allowing the development team to respond quickly to any changes in the project.

Of course, the ultimate choice of methodology will depend on the specific requirements and constraints of your project, as well as the preferences of your development team**.**

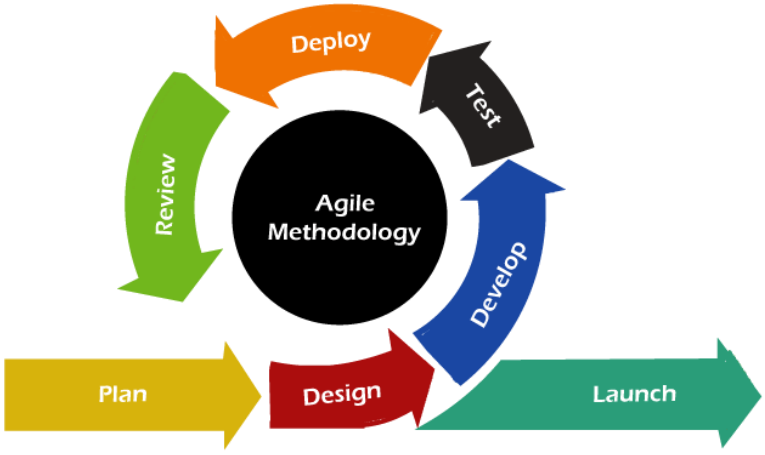


Fig 2.1 Representation of Agile Model

Chapter 3 Identifying Requirements

3.1 System Requirements:

3.1.1 A standalone computer(i3 5 gen, 8 gb ram or higher)

3.1.2 High-quality wireless camera to capture images

3.1.3 Secondary memory to store all images and database

3.1.4 Pycharm professional 2017.2.4 or higher

3.1.5 Python 3.5 or more

3.1.6 Windows 8 or higher

3.1.7 Latest version of all libraries

3.2 Functional Requirements:

System functional requirement describes activities and services that must provide.

3.2.1 A user must be able to manage student records.

3.2.2 An only authorized user must be able to use the system.

3.2.3 A system must be attached to wireless camera and face recognition should be smooth.

3.2.4 The administrator or the person who will be given the access to the system must login into the system before using it.

3.2.5 The information should be entered and managed properly.

3.3 Non-Functional Requirements:

3.3.1 The GUI of the system will be user-friendly.

3.3.2 The data that will be shown to the users will be made sure that it is correct and is available for time being. The system will be flexible to changes.

3.3.3 The system will be extended to changes and the latest technologies.

3.3.4 Efficiency and effectiveness of the system will be made sure.

3.3.5 The Accuracy of recognizing a face which belongs to users of the system is more than 83% whereas 96% for unknown people.

Chapter 4 Project Plan and Effort

4.1 Project Plan:

Focus Area	Details
Schedule Management	• Governance Framework • Project Team Structure • Roles & Responsibilities of Team • Project Closure
Scope Management	• Scope Statement • Requirement Management (Gathering, Control, Assumption, Constraint Stakeholder) • Define Deliverable • Requirement Change Control
Schedule Management	• Define Milestones • Schedule Control
Cost Management	• Estimate Effort • Assign Team • Budget Control
Quality Management	• Quality Assurance: Quality assurance will be managed including governance, roles and responsibilities, tools and techniques and reporting • Quality Control: Specify the mechanisms to be used to measure and control the quality of the work products
Resource Management	• Estimate and Manage the need • People: People & Skills Required • Finance: Budget Required • Physical: Facilities, IT Infrastructure

Stakeholder	• Identifying, • Analyzing • Engaging Stakeholders
Communication Management	Determine communication requirements, roles and responsibilities, tools and techniques. [Type of Communication, Schedule, Mechanism Recipient]
Risk Management	• Identifying • Analyzing • Prioritizing project risks
Procurement Management	Adhering to organization procurement process

4.2 Effort Estimation

4.2.1 Effort and Cost Estimation

Activity Description	Sub-Task	Sub-Task Description	Effort (in hours)	Cost in INR
Design the user screen	Interface Creation	Creating the interface for uploading data	4	2000
	Algorithm Building	Creating Algorithm to recognize objects, people , things	8	4000
	Testing	Implementing Algorithm in real life	4	2000
Identifying Data and Implementing it in real life situations	Real time Managing	Creating a DSBS for storing all real time data	12	6000
	Merging Algorithm	Merging real time data with Algorithm	2	1000

Effort (hr)	Cost (INR)
1	500

4.2.2 Infrastructure/Resource Cost [CapEx]

Infrastructure Requirement	Qty	Cost per qty	Cost per item
Laptop	2	80000	160000
Keyboard	2	2000	4000
Mouse	2	1500	3000

4.2.3 Maintenance and Support Cost [OpEx]

Category	Details	Qty	Cost per qty per annum	Cost per item
People	Network, System, Middleware and DB admin Developer , Support Consultant	3	2,000,000	6,000,000
License	Operating System Database Middleware IDE	10	10000	100,000
Infrastructures	Server, Storage and Network	20	20000	400,000

4.2.4 Project Team Formation

Name	Role	Responsibilities
Priyansh Agarwal	Key Business User (Product Owner)	Provide clear business and user requirements
Priyansh Agarwal	Project Manager	Manage the project
Harsh Thite	Business Analyst	Discuss and Document Requirements
Harsh Thite	Technical Lead	Design the end-to-end architecture
Khushi Padia	UX Designer	Design the user experience
Khushi Padia	Frontend Developer	Develop user interface
Harsh Thite	Backend Developer	Design, Develop and Unit Test Services/API/DB
Priyansh Agarwal	Cloud Architect	Design the cost effective, highly available and scalable architecture
Priyansh Agarwal	Cloud Operations	Provision required Services

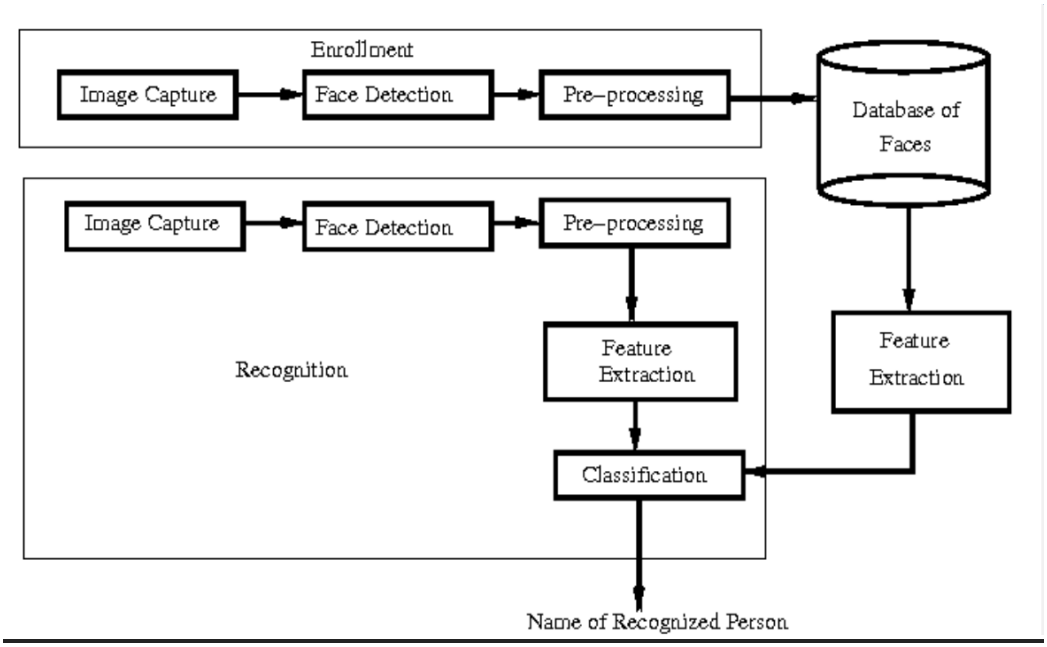
4.2.5 Responsibility Assignment Matrix

RACI Matrix		Team Members		
Activity	**Khushi Padia**	**Harsh Thite**	**Priyansh Agarwal**	**Key Business User**
1.Interface	R	A	C/I	R
2.Algorithm	C	C	A	R
3.Maintanence	C	R	I	R

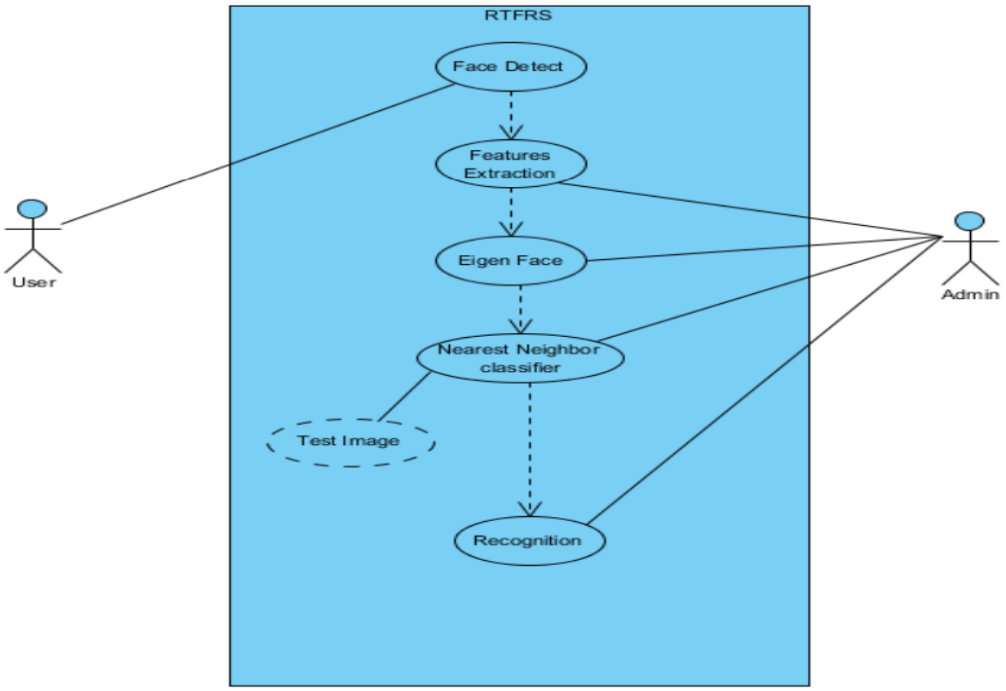
A	Accountable
R	Responsible
C	Consult
I	Inform

Chapter 5: SYSTEM ARCHITECTURE, USE CASE AND CLASS DIAGRAM

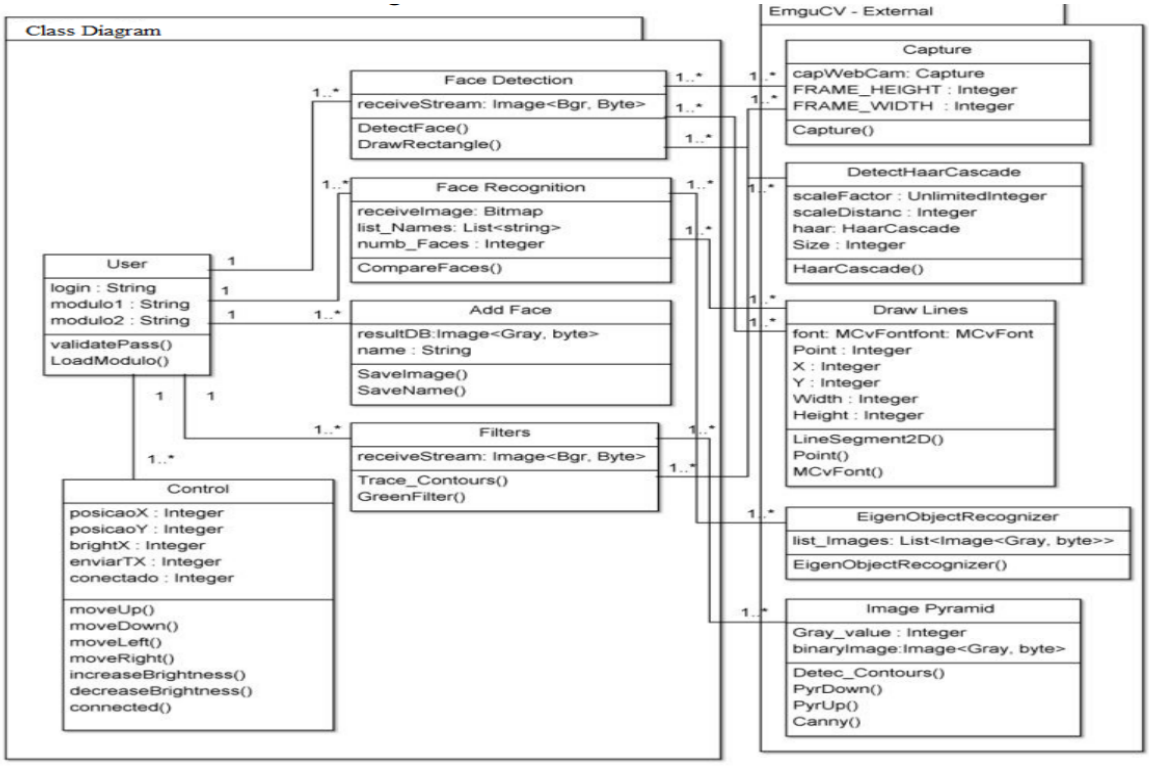
5.1 System Architecture:



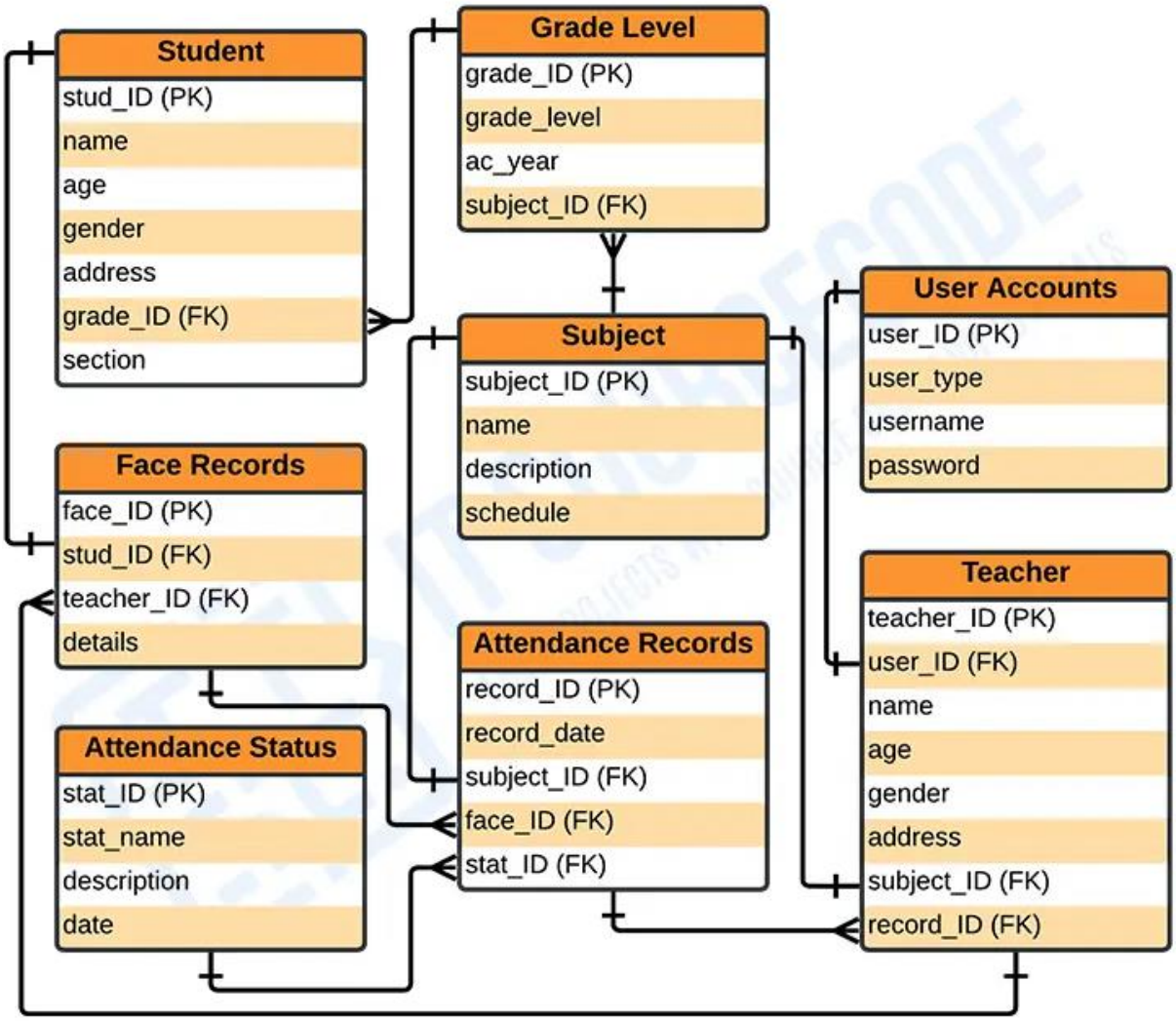
5.2 Use case:



5.3 Class Diagram:

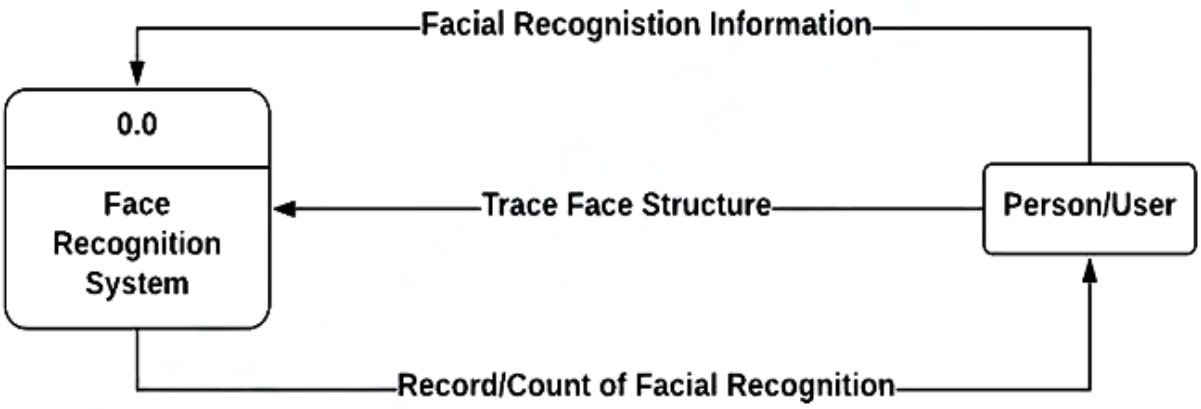


Chapter 6: Entity Relationship Diagram:

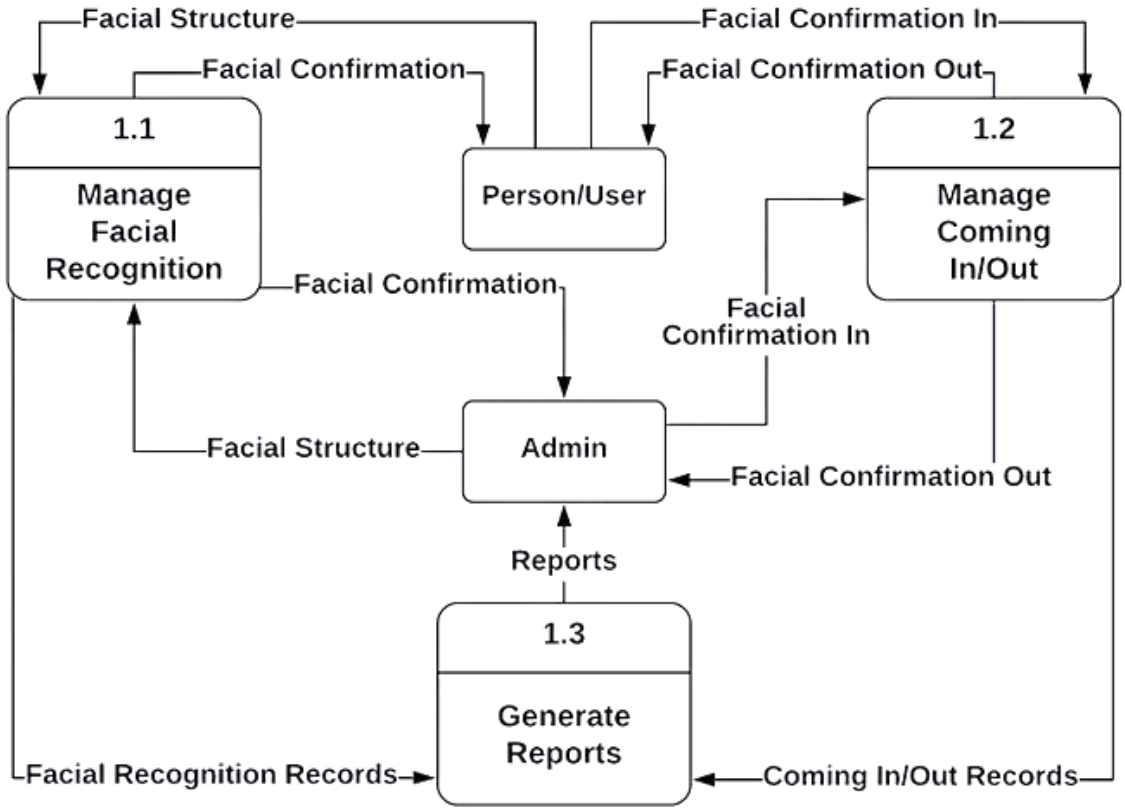


Chapter 7: Data Flow Diagram:

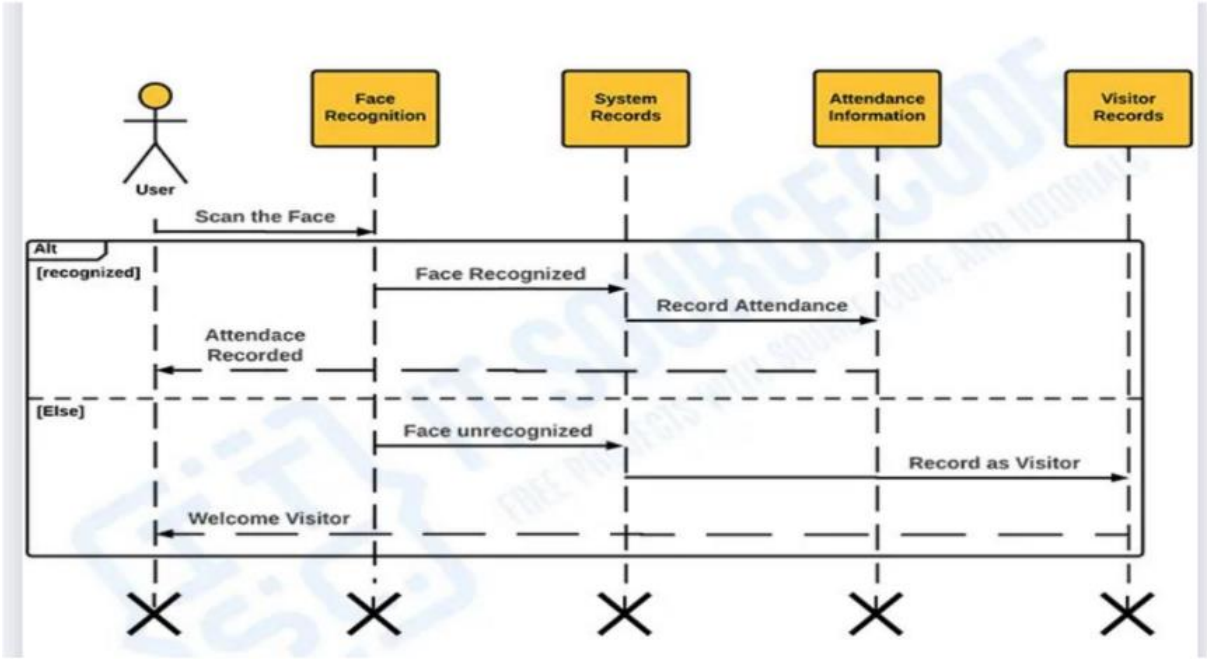
7.1 Data Flow Diagram for Level 0:



7.2 Data Flow Diagram for level 1:



Chapter 8: Sequence Diagram



Chapter 9: Development of Testing Framework/User Interface

9.1 Scope of Testing and User Interface

Some facial Recognition algorithms identify facial features by extracting landmarks, or features, from an image of the subject's face. For example, an algorithm may analyze the relative position, size, and/or shape of the eyes, nose, cheekbones, and jaw. These features arethen used to search for other images with matching features. Other algorithms normalize a gallery of face images and then compress the face data, only saving the data in the image that is useful for face Recognition. A probe image is then compared with the face data. One of theearliest successful systems is based on template matching techniques applied to a set of salientfacial features, providing a sort of compressed face representation. Recognition algorithms can be divided into two main approaches, geometric, which looks at distinguishing features, or photometric, which is a statistical approach that distills an image into values and compares thevalues with templates to eliminate variances

When it comes to managing attendance, there are different ways to inculcate it. However, if you need a touchless biometric system, this is it. You can opt for face Recognition attendance.Face Recognition attendance is the one where the user has to stand still in a fixed

position, in front of the camera for a retina scan, and thus, the identification is measured. The user has to take the face image from a certain distance. In this method of attendance tracking, you can havea contactless, and hygienic interface.

9.2 Test Case and Reporting

9.2.1 Testing Plan:

- Category Progress Against Plan Status
- Functional Testing
- Non-Functional Testing

9.2.2 Present Obstacles:

- Limited availability of resources, specifically the availability of team members fortesting.
- Delay in receiving feedback from stakeholders regarding test results.
- Integration testing with third-party applications is yet to be completed.

9.2.3 Seek help from stakeholders to remove obstacles/constraints:

- Request for additional resources to speed up the testing process.
- Communicate with stakeholders to expedite the feedback process.

9.3 Test Cases :

9.3.1 Functional Test Cases

Test ID (#)	Test Scenario	Test Case	Execution Steps	Expected Outcome	Actual Outcome	Status	Remarks
1	Admin sign up	To create an admin account	1. Start app 2. Fill in details for admin sign up	On sign up, it should create an admin account	It created an admin account on sign up	Pass	success
2	User registration	To register and store user information	1. Start the application 2. Admin login 3. User registration	Registration form should be fill up according to validation set up and any incorrect detail should be popped as warning	Registration form was filled successfully	Pass	success

3.	IP camera and configuration	To open IP camera after filling registration form	1. Start the application 2. Admin login 3. User registration 4. User face registration	IP camera should be connected and turned on after filling registration form.	IP camera was turned on after filling registration form.	Pass	success
4.	Registration of faces	To capture image of a particular user and close automatically	1. Start the application 2. Admin login 3. User registration 4. User face registration	After connecting IP camera to the system, the face should be detected and capture 100 images of the user and close automatically with a message "Registration Over".	Image of the user was captured and message was displayed	Pass	success
5.	Encoding registered faces	To get the folder names in training-dir as labels to be encoded in the form of machine learning.	1. Start the application 2. Admin login 3. Encode database	Get the folder names in training-dir as labels to be encoded in the form of machine learning and if a file with that name already exist then backup the old file.	The folder names in training-dir were encoded in the form of machine learning.	Pass	success
6.	entry in excel sheet	To enter today's date in the column and match name which are already printed at the time of registration with the matched faces.	1. Start the application 2. Admin login 3. Take enteries	Enter today's date in the column and match name which are already printed ay the time of registration with the matched faces.	Today's date was entered into the column and names were matched with the matched faces.	Pass	success

9.3.2 Non Functional Test Cases

Test ID (#)	Test Scenario	Test Case	Execution Steps	Expected Outcome	Actual Outcome	Status
1.	Performance testing	To test the system's ability to handle the load of multiple users logging in at the same time.	Made 15 users to login and register at the same time.	All faces to be registered without any issue.	All faces were registered without any issue.	Pass
2.	Security testing	To test the system's security features, such as password protection, encryption, and data privacy.	Made ensure that the system complies with industry standards and regulations.	Ensure that the system is secure, and that it protects the sensitive data and information of the users who interact with it.	System is secured and protects the sensitive data and information of the users.	Pass
3.	Compatibility testing	Test the system's compatibility with different devices, platforms, and operating systems.	Made ensure that the system works seamlessly with a wide range of hardware and software configurations.	Ensure that the system works seamlessly with a wide range of devices, platforms, and operating systems.	System works on wide range of devices, platforms.	Pass

Category	Progress Against Plan	Status	Comments
Functional Testing	Amber	In-Progress	All the test case of Face detection with attendance system has been passed and any issue or defect that was found during testing has been resolved and reported.
Non-Functional Testing	Green	Completed	The testing team has completed the non-functional testing phase, which included performance testing, security testing. The test plan and test cases for this phase were executed and any issue or defects found were reported and resolved.

Functional	ModuleID	Test case ID	Test case description	Test Case Coverage (%)	Test data	Expected result	Actual result	Status
Login	M1	TC01	Verify that user can login using correct credentials	100%	Valid credentials	Logged in	Success	Completed
Login	M2	TC02	Verify that user can't login using incorrect credentials	100%	Invalid credentials	Not loggedin	Success	Completed
Customer management	M3	TC03	Verify that a new person canbe added	90%	Customer details (name, picture)	Verified	Success	Completed
Customer management	M4	TC04	Verify a user can't put attendance more than once	90%	Same user who already put attendance	Verified	Success	Completed
Registration offaces	M5	TC05	Verify that capture image of	90%	User image	Verified	Success	Completed
			a particular userand close automaticall y					
Attendance inexcel sheet	M6	TC06	To enter today'sdate in the column and match name which are already printed at the time of registration withthe matched faces.	100%	User image, date and time	Verified	Success	Completed
View attendance	M7	TC07	To verify absentees viaemail	90%	Absentees email id	Verified	Success	Completed

Chapter 10: Architecture/Design/Framework/Implementation

A face detection and Recognition system typically involves four main components: the hardware for capturing the facial images, the software for processing the images and recognizing faces, the database for storing attendance records, and the user interface for accessing the attendance data.

10.1 Hardware

The hardware used for capturing facial images can be a camera, either standalone or embedded in a device such as a smartphone or tablet. The camera should be capable of capturing clear images of faces, ideally in varying lighting conditions.

10.2 Software

The software used for face detection and recognition can be built using various frameworks and libraries such as OpenCV, TensorFlow, and Keras. The software should be able to detect faces in an image, recognize individuals from their facial features, and track their movements within a given space. To improve accuracy, the system can be trained using machine learning algorithms.

10.3 Database

The attendance records can be stored in a database, either locally or on a remote server. The database should be designed to efficiently store and retrieve attendance records, and to handle multiple users accessing the data simultaneously.

10.4 User Interface

The user interface can be a web-based or mobile application that allows users to access the attendance records. The interface should be user-friendly and provide features such as viewing attendance reports, adding or deleting attendance records, and generating alerts for absentees.

10.5 Implementation:

10.5.1 Identify the requirements and constraints of the attendance system.
10.5.2 Choose appropriate hardware and software components based on the requirements.
10.5.3 Develop the software using the chosen framework or library.
10.5.4 Test the system and refine it based on feedback.
10.5.5 Deploy the system and train users on how to use it.
10.5.6 Monitor the system and perform maintenance tasks as necessary.

Overall, the implementation of a face detection attendance system requires careful planning and execution to ensure its accuracy, reliability, and security.

CONCLUSION

In conclusion, face recognition and detection is a rapidly evolving field that has seen significant advances in recent years. This technology has many important applications, from security systems and biometric authentication to social media tagging and digital image processing.

To develop effective face recognition and detection systems, researchers have addressed numerous challenges related to variability in facial features, lighting conditions, pose variations, occlusions, and the presence of facial hair or accessories. Advanced machine learning techniques such as deep learning and computer vision algorithms have been used to address these challenges.

However, it is important to also consider the ethical concerns related to face recognition and detection technology, including privacy violations, biases, and misuse. Careful consideration of these ethical concerns is necessary to ensure responsible development and deployment of these systems.

Looking forward, there are many opportunities for future research in this field. These include improving the accuracy and robustness of face recognition and detection systems, developing systems that can handle large and diverse datasets, and addressing ethical concerns through responsible design and deployment. Overall, face recognition and detection is an exciting and important field with the potential to transform many areas of our lives.

REFERENCES

1. Geeksforgeeks.com
2. javaTpoint.com

Appendix(Code)

```
import cv2
import numpy as np
import face_recognition
import os
from datetime import datetime
# from PIL import ImageGrab

path = 'ImagesBasic'
images = []
classNames = []
myList = os.listdir(path)
print(myList)
for cl in myList:
    curImg = cv2.imread(f'{path}/{cl}')
    images.append(curImg)
    classNames.append(os.path.splitext(cl)[0])
print(classNames)

def findEncodings(images):
    encodeList = []
    for img in images:
        img = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)
        encode = face_recognition.face_encodings(img)[0]
        encodeList.append(encode)
    return encodeList
```

```
def markAttendance(name):
    with open('Attendance.csv','r+') as f:
        myDataList = f.readlines()
        nameList = []
        for line in myDataList:
            entry = line.split(',')
            nameList.append(entry[0])
        if name not in nameList:
            now = datetime.now()
            dtString = now.strftime('%H:%M:%S')
            f.writelines(f'\n{name},{dtString}')

encodeListKnown = findEncodings(images)
print('Encoding Complete')

cap = cv2.VideoCapture(0)

while True:
    success, img = cap.read()
    #img = captureScreen()
    imgS = cv2.resize(img,(0,0),None,0.25,0.25)
    imgS = cv2.cvtColor(imgS, cv2.COLOR_BGR2RGB)

    facesCurFrame = face_recognition.face_locations(imgS)
    encodesCurFrame = face_recognition.face_encodings(imgS,facesCurFrame)

    for encodeFace,faceLoc in zip(encodesCurFrame,facesCurFrame):
        matches = face_recognition.compare_faces(encodeListKnown,encodeFace)
```

```
        faceDis = face_recognition.face_distance(encodeListKnown,encodeFace)
        #print(faceDis)
        matchIndex = np.argmin(faceDis)

        if matches[matchIndex]:
            name = classNames[matchIndex].upper()
            #print(name)
            y1,x2,y2,x1 = faceLoc
            y1, x2, y2, x1 = y1*4,x2*4,y2*4,x1*4
            cv2.rectangle(img,(x1,y1),(x2,y2),(0,255,0),2)
            cv2.rectangle(img,(x1,y2-35),(x2,y2),(0,255,0),cv2.FILLED)
            cv2.putText(img,name,(x1+6,y2-6),cv2.FONT_HERSHEY_COMPLEX,1,(255,255,255),2)
            markAttendance(name)

        if faceDis[matchIndex]< 0.50:
            name = classNames[matchIndex].upper()
            markAttendance(name)
        else: name = 'Unknown'
        #print(name)
        y1,x2,y2,x1 = faceLoc
        y1, x2, y2, x1 = y1*4,x2*4,y2*4,x1*4
        cv2.rectangle(img,(x1,y1),(x2,y2),(0,255,0),2)
        cv2.rectangle(img,(x1,y2-35),(x2,y2),(0,255,0),cv2.FILLED)
        cv2.putText(img,name,(x1+6,y2-6),cv2.FONT_HERSHEY_COMPLEX,1,(255,255,255),2)

    cv2.imshow('Webcam',img)
    cv2.waitKey(1)
```

```
<!DOCTYPE html>
<html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Login Page</title>
    <link rel="stylesheet" href="1.css">

</head>

<body>
    <header class="header">

        <div class="left">
            <img src="CODE.jpg">
        </div>

        <div class="mid">

            <li><a href="#">Home</a></li>
            <li><a href="#">About us</a></li>
            <li><a href="#">Contact us</a></li>
            <li><a href="#">Email</a></li>
        </div>

        <div class="right">
            <div class="search">
                <input type="text" name="Search" placeholder="Search">
            </div>

    </header>
    <div class="container">
        <center>
            <h1>
                Login
            </h1>
        </center>
        <div class="forms">
            Name<input type="Name" placeholder="Enter Name"><br>
            Password<input type="password" placeholder="Enter password"><br>
            <button class="btn">Submit</button>

        </div>
    </div>
</body>
```

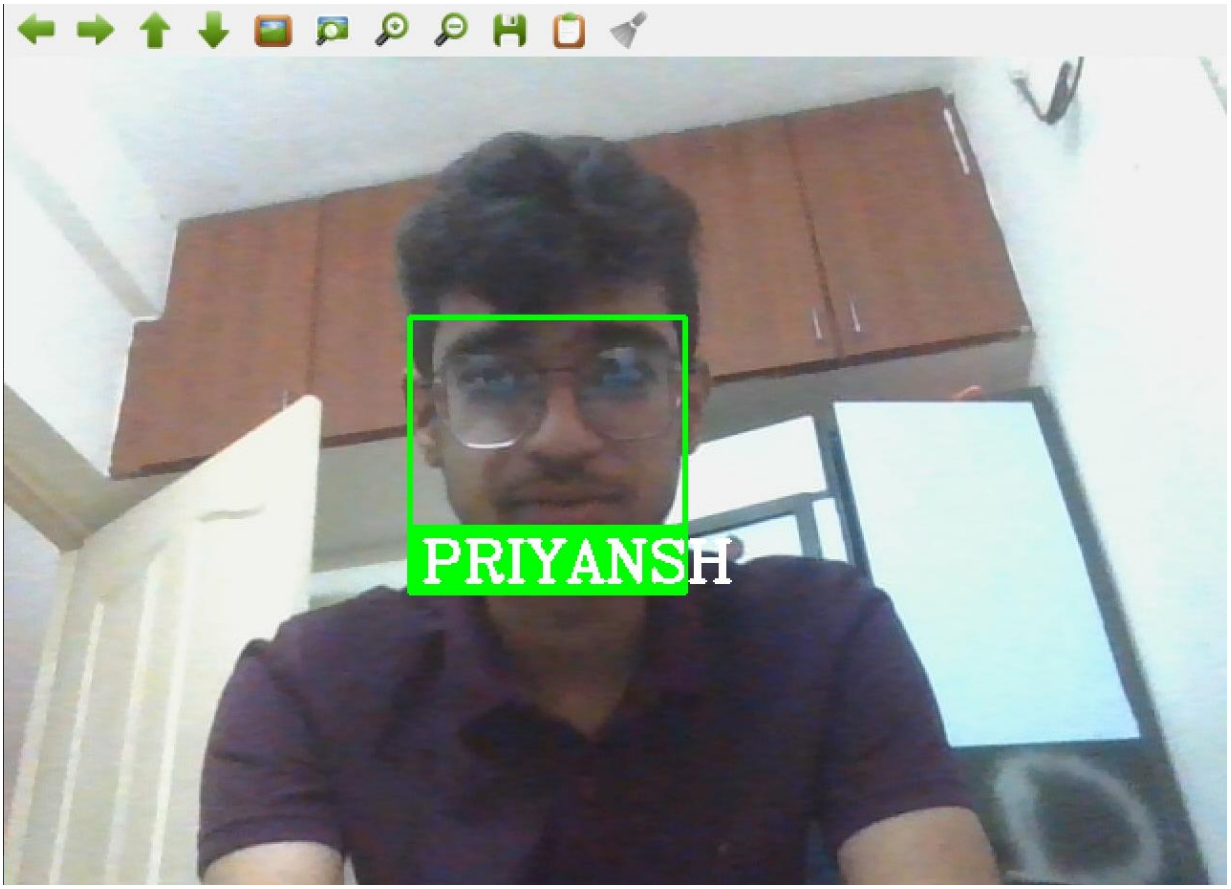
```css
body{
    margin: 0px;
    padding: 0px;
    color: white;
    background: url(bg2.jpg);
    height: 100%;
    background-size: 100%;
     background-repeat: no-repeat;
     }
.header{
    position: sticky;
    display: flex;
    height: 50px;
    flex-wrap: wrap;

}

.left{
    flex: 3;
    position: sticky;

}
.left img{

    width: 140px;
    left: 50px;
    top: 10px;
    padding: 5px;
}
.mid{

    display: flex;
    flex-wrap: wrap;
    flex-direction: row;
    flex: 7;
    justify-content: center;
    align-items: flex-start;

}
.right{
    flex: 1;
    }
li{

    list-style: none;
    margin: auto;
    border: 3px;
    margin: 13px 23px;
}
 li a{
    color: white;
    text-decoration: none;
    padding: 3px 3px;
}
 input{
    padding: 8px 39px;
    border: 3px 5px;
    border-radius: 43px;
    }
.search{
    float: right;
    padding: 12px 70px;
```

```css
li a:hover, li a:active{
   text-decoration: underline;
  color: gray;
}
.btn:hover{
    cursor: pointer;
    background-color: rgb(6, 77, 101);
    color: white;
}
.container{
border: 2px solid white;
margin: 118px 213px;
width: 33%;
padding: 59px 60px;
border-radius: 10px;
}
.forms input{
display: flex;
flex-wrap: wrap;
text-align: center;
margin: 6px 9px;
padding: 12px 65px;
}
.container button{
display: flex;
flex-wrap: wrap;
width: 15%;
margin: auto;
}
```

Sample Screenshots

Login Page

Recognition



ENTRY TIME-

NAMES	TIME
PRIYANSH	15:53:56
HARSH	16:34:22
KHUSHI	17:25:43