Logo

Description automatically generated**College of Computing and Informatics**

**Computer Science Department**

**University of Sharjah**

**Fall 2021/2022**

**Face recognition-based attendance system for primary school and kindergarten kids**

#### by

Reema Alharbi (U17100706)

Khuloud Khalid (U18105068)

#### Supervised by

Dr Ibrahim Abaker Hashem

Dec 2021

**ACKNOWLEDGEMENT**

We would like to precise our uncommon much obliged of appreciation to our instructor Dr Ibrahim Abaker Hashem  as well as our head department Dr Naveed Ahmed who gave us the brilliant opportunity to do this superb project on the subject Face recognition, which moreover made a difference in us in doing a part of Inquire about and we came to know around so numerous modern things we are truly grateful to them. Secondly, we would moreover to thank our guardians and companions who gave us the needed motivation to finalizing this project within the limited time frame.

**UNDERTAKING**

These days, biometric verification strategies start developing quickly as one of the promising verification strategies, other than the ordinary verification strategy. Nearly all biometrics innovations require a few activities by the client, which is the client must put reserves on the scanner to set the fingers or the hand geometry discovery. The user should standstill in a settled position before the camera for iris or retina identification reasons. The face recognition strategy has a few outside focal points compared to the other biometric strategies since this strategy can be done latently without express activity or ought to be held by the client since the confront picture can be gotten by the camera from a certain remove. This strategy can be particularly valuable for mission and supervision. This investigation would create and execute the face recognition framework comprised of four organize handles. All the analysis, design and system development have been accomplished by the undersigned. Moreover, this project has not been submitted to any other college or university.

* Reema Alharbi U17100706
* Khuloud Khalid U18105068

**ABSTRACT**

There are incidents of children being left behind on school buses that happen every year, which can result in their deaths, especially in summer, due to high temperatures. Research has shown that children's bodies appear to heat up three to five times faster than adults, while children are unable to cool down, too. Paediatric associations worldwide have further documented the negative effects of high temperature on kids, with children frequently falling victim to dehydration of hyperthermia and other risks that may be fatal.

With the evolution of technology, we aim to create a system that decreases the number of children dying after being left behind on buses. Since it is easier to take attendance using face recognition than a manual attendance system and to make sure parents feel safe about having their kids riding school buses, our system makes sure there is a communication between them and their child’s status of whether they entered or left the bus by receiving notifications.

The main objective of this project is to provide a system based on Face Recognition (FR) technology. Unlike the manual attendance system, our system will simplify and automates the process of documenting and monitoring the attendance of students. Machine learning algorithms will be adopted for facial recognition to improve the limitations of the present systems, it requires a high-quality camera to record student images, a database, and a mobile application. Our project will be implemented using python and MySQL technologies. This system will be responsible of monitoring student in school buses and make sure that all the students will reach home safely and on time.

**Table of Contents**

[LIST OF TABLES vii](#_bookmark0)

[LIST OF FIGURES viii](#_bookmark1)

[CHAPTER 0: General format guidelines ix](#_bookmark2)

[CHAPTER 1: Introduction (Heading 1) 1](#_bookmark5)

* 1. [Overview 1](#_bookmark6)
  2. [Project Motivation 1](#_bookmark7)
  3. [Problem Statement 1](#_bookmark8)
  4. [Project Aim and Objectives 1](#_bookmark9)
  5. [Project Scope 1](#_bookmark10)
  6. [Project Software and Hardware Requirements 2](#_bookmark11)
  7. [Project Limitations 2](#_bookmark12)
  8. [Project Expected Output 2](#_bookmark13)
  9. [Project, product, and schedule risks 2](#_bookmark15)

[CHAPTER 2: Related Existing System 3](#_bookmark16)

* 1. [Introduction 3](#_bookmark17)
  2. [Existing Systems 3](#_bookmark18)
  3. [Overall Problems of Existing Systems 3](#_bookmark19)
  4. [Overall Solution Approach 3](#_bookmark20)

[CHAPTER 3: Requirement Engineering and Analysis 4](#_bookmark21)

* 1. [Stakeholders 4](#_bookmark22)
  2. [Use Case Diagram 4](#_bookmark23)
     1. [Use Case Section 4](#_bookmark24)
  3. [Non-functional requirements 4](#_bookmark25)
  4. [Constraints 4](#_bookmark26)

[CHAPTER 4: Architecture and Design 5](#_bookmark27)

* 1. [Overview 5](#_bookmark28)
  2. [Software architecture 5](#_bookmark29)
     1. [Physical view 5](#_bookmark32)
     2. [Details of each component in a separate section 5](#_bookmark33)
  3. [Software design 5](#_bookmark34)
     1. [UML sequence/communication diagram 5](#_bookmark35)
     2. [Class diagram 5](#_bookmark36)
     3. [State transition diagram 5](#_bookmark38)
  4. [User interface design (prototype) 5](#_bookmark39)

[CHAPTER 5: Implementation Plan 6](#_bookmark40)

* 1. [Description of Implementation 6](#_bookmark41)
  2. [Attendance System 6](#_bookmark42)
  3. [User Interface (React Native) 6](#_bookmark42)
  4. [Programming language and technology 6](#_bookmark42)
  5. [IDE’s, Tools and Technologies 6](#_bookmark42)
  6. [Languages Used 6](#_bookmark42)
  7. [Facial Recognition Set Up 6](#_bookmark42)
  8. [Database Set Up 6](#_bookmark42)

[CHAPTER 6: Restrictions 7](#_bookmark44)

[CHAPTER 7: Conclusion and Results 8](#_bookmark48)

[CHAPTER 8: References 9](#_bookmark49)

[CHAPTER 9: Appendix 9](#_bookmark49)

# LIST OF TABLES

[TABLE 0-0 Schedule 14](#_bookmark4)

[TABLE 0-1 Existing Systems 16](#_bookmark4)

[TABLE 0-2 Stakeholders 19](#_bookmark4)

[TABLE 0-3 Use Case (1) 20](#_bookmark4)

[TABLE 0-4 Use Case (2) 21](#_bookmark4)

[TABLE 0-5 Use Case (3) 22](#_bookmark4)

[TABLE 0-6 Use Case (4) 23](#_bookmark4)

[TABLE 0-7 Non-Functional Requirements 25](#_bookmark4)

[TABLE 0-8 Physical View 30](#_bookmark4)

# LIST OF FIGURES

[FIGURE 0-0: Overall Solution Approach 18](#_bookmark3)

[FIGURE 0-1: Use Case (1) 20](#_bookmark3)

[FIGURE 0-2: Use Case (2) 21](#_bookmark3)

[FIGURE 0-3: Use Case (3) 22](#_bookmark3)

[FIGURE 0-4: Use Case (4) 23](#_bookmark3)

[FIGURE 0-5: Physical View Diagram 30](#_bookmark3)

[FIGURE 0-6: Attendance Sequence Diagram (1) 31](#_bookmark3)

[FIGURE 0-7: Database Sequence Diagram (2) 31](#_bookmark3)

[FIGURE 0-8: Face Recognition Cameras (FRC) Sequence Diagram (3) 32](#_bookmark3)

[FIGURE 0-9: Class Diagram 33](#_bookmark3)

[FIGURE 1-0: System State Transition Diagram (1) 34](#_bookmark3)

[FIGURE 1-1: System State Transition Diagram (2) 34](#_bookmark3)

[FIGURE 1-2: UI (1) 35](#_bookmark3)

[FIGURE 1-3: (a) 36](#_bookmark3)

[FIGURE 1-4: (b) 36](#_bookmark3)

[FIGURE 1-5: UI (2) 37](#_bookmark3)

[FIGURE 1-6: UI(3) 38](#_bookmark3)

[FIGURE 1-7: UI (4) 39](#_bookmark3)

[FIGURE 1-8: (a) 40](#_bookmark3)

[FIGURE 1-9: (b) 40](#_bookmark3)

[FIGURE 2-0: UI (5) 41](#_bookmark3)

[FIGURE 2-1: (a) 42](#_bookmark3)

[FIGURE 2-2: (b) 42](#_bookmark3)

[FIGURE 2-3: UI (6) 43](#_bookmark3)

[FIGURE 2-4: UI (7) 44](#_bookmark3)

[FIGURE 2-5: Implementation 45](#_bookmark3)

[FIGURE 2-6: Neural Network 46](#_bookmark3)

[FIGURE 2-7: Face Recognition(1) 47](#_bookmark3)

[FIGURE 2-8: Face Recognition (2) 47](#_bookmark3)

[FIGURE 2-9: Face Recognition (2) 47](#_bookmark3)

[FIGURE 3-0: Face Recognition (2) 47](#_bookmark3)

[FIGURE 3-1: Face Recognition (2) 47](#_bookmark3)

[FIGURE 3-2: Face Recognition (2) 47](#_bookmark3)

[FIGURE 3-3: Face Recognition (2) 47](#_bookmark3)

[FIGURE 3-4: Face Recognition (2) 47](#_bookmark3)

[FIGURE 3-5: Face Recognition (2) 47](#_bookmark3)

[FIGURE 3-6: Face Recognition (2) 47](#_bookmark3)

[FIGURE 3-7: Face Recognition (2) 47](#_bookmark3)

[FIGURE 3-7: Face Recognition (2) 47](#_bookmark3)

[FIGURE 3-8: Face Recognition (2) 47](#_bookmark3)

[FIGURE 3-9: Face Recognition (2) 47](#_bookmark3)

[FIGURE 4-0: Face Recognition (2) 47](#_bookmark3)

[FIGURE 4-1: Face Recognition (2) 47](#_bookmark3)

**CHAPTER 1: Introduction**

## Overview

The main objective of this project is to provide a system based on Face Recognition (FR) technology. Unlike the manual attendance system, our system will simplify and automates the process of documenting and monitoring the attendance of students. Machine learning algorithms will be adopted for facial recognition to improve the limitations of the present systems, it requires a high-quality camera to record student images, a database, and a mobile application. Our project will be implemented using React Native and Firebase technologies. This system will be responsible of monitoring student in school buses and make sure that all of the students will reach home safely and on time.

## Project Motivation

There are incidents of children being left behind on school buses that happen every year, which can result in their deaths, especially in summer, due to high temperatures. Research has shown that children's bodies appear to heat up three to five times faster than adults, while children are unable to cool down, too. Pediatric associations worldwide have further documented the negative effects of high temperature on kids, with children frequently falling victim to dehydration of hyperthermia and other risks that may be fatal.

With the evolution of technology, we aim to create a system that decreases the number of children dying after being left behind on buses. Since it is easier to take attendance using face recognition than a manual attendance system and to make sure parents feel safe about having their kids riding school buses, our system makes sure there is a communication between them and their child’s status of whether they entered or left the bus by receiving notifications.

## Problem Statement

* Fingerprint is harder for young students to understand how to use it.
* The database will not be able to handle a huge amount of all the student’s information.
* The notification is not supported by SMS.
* If the system did not work, teachers should overwrite attendance and mark as absent manually.
* The system cannot detect more than two faces at once.

## Project Aim and Objectives

With the evolution of technology, we aim to create a system that decreases the number of children dying after being left behind on buses. Our main objective is to create a successful system that helps in tracking the students attendance in a hands free-way and have a more accurate attendance system.

* Easy way to track students attendance.
* Attendance is easily reported to the school and to the parents.
* Have an integrated notification system.
* Providing a user-friendly UI for accessing the system
* The algorithm will be a combination of various proposed methods along with some other features.

## Project Scope

## The Face recognition process can be divided into two parts first is face verification and the second one is named face identification. The first one i.e., face verification system involves a one-to-one matching to confirm or deny a person’s identity claim. This system compares the captured face image against the person’s template(s) stored in the system. If the student presenting himself/herself to the system is the student, he/she is the same in the database then the system will consider him as attended, otherwise if the system will send a notification to the parents and the admission.

## Project Software and Hardware Requirements

* 4 GB RAM (Minimum)
* 80 GB HDD
* Dual Core processor
* CDROM (installation only). VGA resolution monitor
* Microsoft Windows 98/2000/NT with service pack 6 / XP with service pack 2/ Windows 7 with service pack 2
* SQL Server 2008 R2

## Project Limitations

There are incidents of children being left behind on school buses that happen every year, which can result in their deaths, especially in summer, due to high temperatures. Research has shown that children's bodies appear to heat up three to five times faster than adults, while children are unable to cool down, too. Paediatric associations worldwide have further documented the negative effects of high temperature on kids, with children frequently falling victim to dehydration of hyperthermia and other risks that may be fatal.

## Project Expected Output

This project proposes that the system takes attendance automatically using face recognition cameras, the cameras are placed inside the buses at several locations to capture the faces of the students from various angles. The faces of students are previously stored in the database system. After the capturing of the student's faces, the camera detects and recognizes the faces of the students from the images and compares it to the images stored in the database, if the face matched with the stored image then the student is marked as present and no notification is sent to the parents via the application made for the system, if the image is not a match or undetected then the student is marked as absent and a notification is sent to the parents and at the same time the image is stored in the database as a new image and marked as unknown before further calculations are done. Refer to architectural diagram in Figure 1.[3][2][1]

For the proposed system for the application, and database to communicate with each other, we will use React Native to program the Frontend of the system, while using a tool like Firebase database to find, change and store all the needed data for the students.

our system will perform the main services:

* consolidated student attendance in a day based on notifications sent to the parents via an application.
* monitoring school kids using facial recognition technology

## 

## Project, product, and schedule risks

Image quality affects how well facial-recognition algorithms work. **The image quality of scanning video is quite low compared with that of a digital camera.** Even high-definition video is, at best, 1080p (progressive scan); usually, it is 720p. These values are equivalent to about 2MP and 0.9MP, respectively, while an inexpensive digital camera attains 15MP. The difference is quite noticeable. When a face-detection algorithm finds a face in an image or in a still from a video capture, the relative size of that face compared with the enrolled image size**affects how well the face will be recognized**. An already small image size, coupled with a target distant from the camera, means that the detected face is only 100 to 200 pixels on a side. Further, having to scan an image for varying face sizes is a processor-intensive activity. Most algorithms allow specification of a face-size range to help eliminate false positives on detection and speed up image processing. **The relative angle of the target’s face influences the recognition score profoundly.** When a face is enrolled in the recognition software, usually multiple angles are used (profile, frontal and 45-degree are common). Anything less than a frontal view affects the algorithm’s capability to generate a template for the face. The more direct the image (both enrolled and probe image) and the higher its resolution, the higher the score of any resulting matches. Even though high-definition video is quite low in resolution when compared with digital camera images, it still occupies significant amounts of disk space. Processing every frame of video is an enormous undertaking, so usually only a fraction (10 percent to 25 percent) is run through a recognition system. To minimize total processing time, agencies can use clusters of computers. However, adding computers involves considerable data transfer over a network, which can be bound by input-output restrictions, further limiting processing speed.

Ironically, humans are vastly superior to technology when it comes to facial recognition. But humans can only look for a few individuals at a time when watching a source video. A computer can compare many individuals against a database of thousands.

**CHAPTER 2: Related Existing System**

## Introduction

Programming has been popular in the past few years for solving problems with electronics and computers. Using the expertise and skills we learned at university, we established a system for identifying faces. In the last few years, programmers have experienced several improvements in their programming abilities. Several systems have been developed, and there has been a lot of face recognition. Such as Biometric Face Recognition , Facial Recognition With AWS, Face Recognition system for Detect Critical Events Faster.[6]

When the current system was put into action, it was found to have certain weaknesses. For example, the existing system did not concentrate on allowing parents to easily contact authorities if anything went wrong while they were tracking their children's attendance via the app or within the time they had scheduled. The shortcomings of the current system prompted the development of a new system for contacting school administration or, at the very least, bus drivers.[6]

## Existing Systems

## Table 0-1

|  |  |
| --- | --- |
| The differences between the current system and the proposed system  **Current system**   * **There is no direct communication** * **A parent should only watch one child at a time** * **• The current system's efficiency was poor.** * **• The device does not keep track of active users.** * **The machine will only alert if you open and check it** | **Proposed system**   * **The system that will be built here is a chat facility and a direct call system.** * **It's a machine that's run from a central location.** * **All local clients are connected to the centralized server vis lan** * **The centralized server is accessible via LAN to all local parents** * **If the camera is not scanning or is impaired, attendance may be taken using a finger print.** * **there is a two way communication between different parents** * **It helps users to identify other users who are logged in.** |

The current system needs an internet connection; however, the proposed system does not require an internet connection to use the app for monitoring and instead requires a LAN connection. This device is helpful to those who cannot afford a home internet connection. For example, not every bus has internet access, and not every family does as well.[7]

## Overall Problems of Existing Systems & Solutions

## Our existing system would include a chat function as well as a direct call system; it is a computer that is operated from a central location, despite the fact that all clients are connected to the centralized server via LAN. If the cameras are not scanning or are impaired, attendance may be taken with a finger print. In other cases, there is two-way contact between different parents, and it often aids users in identifying other logged-in users.

## First, there is an existing system named Avalon Face Recognition, Temperature Monitoring, and Access Control System, which only saves time via the device and costs about $2,499 “. Even with a face mask on, the device captures and recognizes the human face, tests body temperature, and monitors the locking/unlocking/alarming for safe access. [8]

## However, our face recognition will be available at an affordable price, making it ideal for parents and all. Avalon faces recognition will be available for entering homes, offices, unlike ours, which will be available for students and buses.[8]

## to begin with , the innovatrica company has a SmartFace, which is a scalable facial recognition platform that can detect and monitor faces in multiple IP camera streams in parallel. [9]

## uniqueness “ Recognizes faces in a variety of places, including those partly obscured by hair, hands, or objects, and works accurately with photographs taken in poor lighting conditions. [9]

## Our face recognition can improve to an excellent scanner for faces even when there is no light because it will have its own lights and detect. Innovatrica can recongize it even in low light, but ours will be able to work even when there is no light at all. [9]

## Lastly, Thermal Face Detection Face Detection SDK For Thermal Cameras & Systems is available from Luxand. Rapidly and accurately detect human faces in thermal camera pictures. With the ability to differentiate real people from pictures and 3D masks, you can take your temperature, check for fever, or simply upgrade your security system. [10]

## Eventually, Luxand's face recognition is good for pendamic in Covid19; our system will improve to check faces even with multiple faces or face shields, as well as checking body fever and not allowing the door to open unless there is no fever for the protection of the children. [10]

## Overall Solution Approach

Figure 0-0

# CHAPTER 3: Requirement Engineering and Analysis

## Stakeholders

The environment in which our system operates in is dynamic, by the participants who are either affect or be affected by the system. The primary stakeholders in our system are Schools, Parents, Nurseries, Faculty and Colleges. And the secondary stakeholders are the Students.

Table 0-2

stakeholders (students, lecturers, faculties, and academic

operational staffs).

stakeholders (students, lecturers, faculties, and academic

operational staffs).

stakeholders (students, lecturers, faculties, and academic

operational staffs).

stakeholders (students, lecturers, faculties, and academic

operational staffs).

|  |  |
| --- | --- |
| **Stakeholders** | **Type** |
| Schools | Primary Stakeholder |
| Nurseries | Primary Stakeholder |
| Students (Children) | Secondary Stakeholder |
| Faculty | Primary Stakeholder |
| Colleges | Primary Stakeholder |
| Parents | Primary Stakeholder |

* The Schools will be the internal stakeholders and the Parents will be the external stakeholders.

## Use Case Diagram

### Use Case Section

Normal Flow for each use case including action, precondition, post-condition and other sections as you learnt in requirements engineering course.

Diagram

Description automatically generated

|  |  |
| --- | --- |
| Use Case Name: | Login use case |
| Brief Description: | This use case diagram shows the overview of the log in process. |
| Actors: | School(Admin) , Parents |
| Preconditions: | Registration in school system or through the application is required to access the application |
| Post Conditions: | School and Parents can view the relevant home page and receive notifications after logging in.UC736.28 |
| Basic Flow: | 1. School/Parents Register 2. School/Parents log in 3. School/Parents log out |
| Alternative Flow: | Registration through the school system |
| Priority: | High |
| Frequency of Use: | High |
| Special Requirements: | Registration |
| Assumptions: | If log in is not successful, registration in school system is required. |

Figure 0-1

Table 0-3: Use Case (1)

**Diagram

Description automatically generated Figure 0-2**

Table 0-4: Use Case (2)

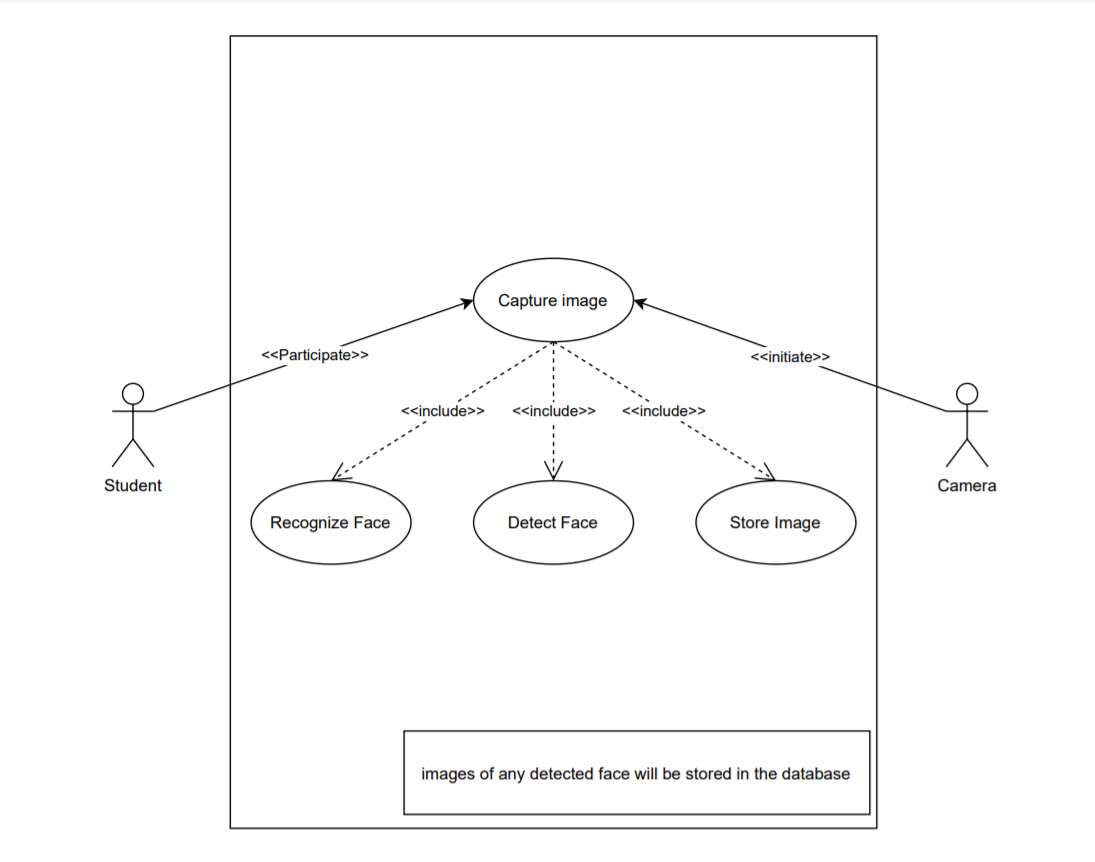
|  |  |
| --- | --- |
| Use Case Name: | Attendance use case |
| Brief Description: | This use case diagram shows the overview of the whole FRAS with its main actors who can log in to the system to view the attendance and receive notifications from the FRA system. |
| Actors: | School(Admin) , Parents, FR system(Database) |
| Preconditions: | Registration in school system is required to access the FRAS. |
| Post Conditions: | School and Parents can view the relevant home page and receive notifications after logging in.UC736.28 |
| Basic Flow: | 1. School/Parents log in 2. View student attendance 3. School/Parents receive notifications via application |
| Alternative Flow: | None |
| Priority: | High |
| Frequency of Use: | High |
| Special Requirements: | None |
| Assumptions: | If log in is not successful, registration in school system is required.  Parents and School Admins will receive notifications if students are absent. |

**Diagram

Description automatically generated Figure 0-3**

Table 0-5: Use Case (3)

|  |  |
| --- | --- |
| Use Case Name: | Database Use Case |
| Brief Description: | This use case shows the management of the database system that is being managed manually by the School(Admin) as its main actor. |
| Actors: | School(Admin) |
| Preconditions: | Logging in as an admin gives you access to manage the attendance database system. |
| Post Conditions: | Managing the database manually in case of unexpected occurrences of errors that might result from the FRAS.UC736.28 |
| Basic Flow: | 1. Log in as school admin 2. Can view relevant homepage depending on their level of access 3. Can view attendance database 4. Can view attendance reports 5. Can manage attendance database(Add, Delete, or Modify attendance) |
| Alternative Flow: | None |
| Extension Points: | None |
| Priority: | High |
| Frequency of Use: | High |
| Business Rules: | None |

****

**Figure 0-4**

Table 0-6: Use Case (4)

|  |  |
| --- | --- |
| Use Case Name: | Face Recognition Cameras (FRC) Use Case |
| Brief Description: | This use case shows the overview process of captured the images of the students to the matching process(Recognizing and detecting) |
| Actors: | Student, Camera |
| Preconditions: | Students images must be stored in the database beforehand. |
| Post Conditions: | A student will be marked as present if the FRC recognized their face, if not, the student will be marked as absent and their image will be stored in the database.UC736.28 |
| Basic Flow: | 1. Image is captured then goes through face detection 2. Detected face goes through Face recognition and stored in database 3. Recognized face gets stored in database |
| Alternative Flow: | None |
| Priority: | High |
| Frequency of Use: | High |
| Business Rules: | None |
| Special Requirements: | None |
| Assumptions: | If captured face is recognized the database marks the attendance, if not it will mark as absent. |

## Non-functional requirements

Specify the non-functional requirements of this project that can be divided into two main categories:

1. Execution qualities, such as safety, security and usability, which are observable during operation (at run time).
2. Evolution qualities, such as testability, maintainability, extensibility and scalability, which are embodied in the static structure of the system.

Table 0-7: Non-Functional Requirements

|  |  |
| --- | --- |
| **Category** | **Subcategory** |
| **Usability** | **Accessibility**  Not Applicable |
| **Aesthetics**  Not Applicable |
| **UI Consistency** The system shall be compatible with different system resolutions and different versions of operating systemsExamples of OSs: IOS, Android, Windows Phone OS |
| **Ergonomics**  Not Applicable |
| **Reliability** | **Availability**   1. The system should be operational 24/7 days depends on the network availability. |
| **Robustness**   1. For every invalid input from the users, the system shall display a meaningful error message explaining what format input is expected. 2. All the errors regarding the networks, Database server systems will be logged to log file and the user will be notified with specific error message. 3. Exceptions regarding database transactions or business logic will be logged to a log file and the user will be notified with specific error message. |
| **Accuracy**   1. Date and time of the attendance marking should be calculated at all times. |
| **Fault Tolerance**  Not Applicable |
| **Safety**  Not Applicable |
| **Security**   1. The system shall provide a multi-level security access based on the user profiles. |
| **Correctness**   1. The system shall validate each of the data entry type based on user input. |
| **Performance** | **Throughput**   1. The system shall accommodate 50 data entries per minute. |
| **Response Time**   1. Average system response time should be less than five seconds for each of web page. 2. Average time of returning a list of database shall not be more than 10 seconds. |
| **Recovery Time**   1. Response-time of this system should be less than 5 seconds. 2. Average repair time shall be less than one hour*.* |
| **Startup/Shutdown time**  Not Applicable |
| **Capacity**   1. The system shall accommodate 50 concurrent users. |
| **Utilization of resources.**  Not Applicable |
| **Supportability** | **Adaptability**   1. The supported server platform for database is the current approved versions of Windows. |
| **Maintainability**   1. An error log containing information about all critical errors shall be accessible to the system administrator. |
| **Compatibility**   1. After the system is in production, subsequent version of the system shall be backward-compatible. All transactions entered in previous versions shall be available in the new version. |
| **Configurability**  Not Applicable |
| **Upgradeability**   1. No installation on a user workstation shall be required. All system upgrades and new releases should be done on the server. |
| **Installability**   1. Installing a new version of the system shall not require any installation on user’s workstation. |
| **Scalability**  Not Applicable |
| **Portability**   1. Changing the system database in the future shall require rewriting application logic. |
| **Reusability**  Not Applicable |
| **Interoperability**  Not Applicable |
| **Compliance**  Not Applicable |
| **Localizability**  Not Applicable |
| **Design Constraints** | Not Applicable |
| **Implementation Requirement** | Software :  Operating System  Database |
| **Interface Requirement** | **User Interface**   1. The system shall provide the menu from each displayed Javascript page for user navigation. |
| **Hardware Interface**  Not applicable |
| **Software Interface**  Not Applicable |
| **Communication Interface**  Not Applicable |
| **Business Rules** | Not Applicable |
| **Documentation Requirement** | Basic documentation need to be provided for all the respective development activities. |
| **Licensing and Legal Requirements** | 1. Windows 10, Linux or Later 2. MySQL Enterprise 5 |
| **Post Development Requirement** | Training Requirements as stated below :   |  |  | | --- | --- | | **Training Type** | **Target Audience** | | **End-User Training:-** A total of 2-days System and Administration Training, will be provided during the project | Schools and Parents | |
| **Technology Transfer Requirements**  Not Applicable |
| **Maintenance Requirements**  Not Applicable |

## Constraints

List the conditions and restrictions of this project that must be satisfy.

Different issues can impact the specification, design, or implementation of the system such as:

1. Local hardware and software space (a server is needed to create the database (DB)).
2. Migration of existing data could be difficult.
3. Personal change.
4. Lack of team expertise.
5. Unwilling users.
6. Lack of top management support.
7. Conflicting preferences between users and system developers.
8. Inappropriate user interface.
9. Continual requests for change.
10. Lack of time management

# CHAPTER 4: Architecture and Design

## Overview

This project proposes that the system takes attendance automatically using face recognition cameras, an interface is provided for users to interact with the system to have a visual sense of the students’ attendance, the cameras are placed inside the buses at several locations to capture the faces of the students from various angles. The faces of students are previously stored in the database system. After the capturing of the student's faces, the camera detects and recognizes the faces of the students from the images and compares it to the images stored in the database, if the face matched with the stored image then the student is marked as present and no notification is sent to the parents via the application made for the system, if the image is not a match or undetected then the student is marked as absent and a notification is sent to the parents via the application and at the same time the image is stored in the database as a new image and marked as unknown before further calculations are done.[3][2][1]

## Software architecture

### Physical view

### The software architecture starts of when a user logs in the system which can be a Parent or a School Admin, after logging in the application will refer the user to their homepage depending on their access level, both users can get to view the results of the attendance based on the matching process but only the admin can view and modify the database. The captured images of the students face acts as the input for the matching process that can send a notification to both users.

Diagram

Description automatically generated

Figure 0-5: Physical View Diagram

* + 1. Details of each component in a **separate section**.

Table 0-8

|  |  |  |
| --- | --- | --- |
| view | Components | users |
| Physical view | Database, cameras, Application, Training model | School Systems, Parents, Students (Children) |

## Software design

### UML sequence/communication diagram

### This sequence diagram shows the interaction between the users and the system after logging in and receiving notifications.

Diagram

Description automatically generated

Figure 0-6: Attendance Sequence Diagram

### This sequence diagram shows the interaction of the school admin with the system, the school admin can manage attendance (Add, Delete, Update) ,school admin can also view the report and view the attendance.

Diagram

Description automatically generated

Figure 0-7: Database Sequence Diagram

### This sequence diagram shows the process of the face recognition cameras, the camera will capture a student’s face, the captured image will be stored whether the face is recognized or not. The face will be detected and recognized then stored as a recognized face.

Diagram

Description automatically generated Figure 0-8: Face Recognition Cameras (FRC) Sequence Diagram

### Class diagram

This class diagram shows the relationships between the main entities of the system, the diagram has 6 classes, System, Parents, Image, School Admin, Student, and Camera. It shows their methods and attributes.

Diagram

Description automatically generated

Figure 0-9: Class Diagram

### State transition diagram

### This state chart shows the process of capturing an image using the face recognition cameras that is then sent to the matching process then the training model which also stores the image in the database and sends alerts on required conditions.

Diagram

Description automatically generated

Figure 1-0: System State Transition Diagram(1)

### This state Chart shows how each of the actors can access the system, the user can access by logging in with their ID & Password, if they are authorized the system will check their level of access if not they will be denied access. Once they are logged in the users can view attendance and receive alerts and some and access database if they are school admins.

Diagram

Description automatically generated Figure1-1: System State Transition Diagram (2)

## User interface design (prototype)

**Splash page:**

A picture containing text, iPod, electronics, cellphone

Description automatically generatedFigure 1-2: UI (1)

* This is the splash page of the app it contains the logo and the name of the app. The user is required to Login or Register in order to access the app.

**Login page:**

Graphical user interface, text, application, chat or text message

Description automatically generatedGraphical user interface, text, application, chat or text message

Description automatically generated

Figure 1-3: (a)

Figure 1-4: (b)

* This is the login page. The user can login to their account by choosing their level of access and are required to enter their ID and password to login.

**Registration page:**

Graphical user interface, text, application, chat or text message

Description automatically generated

Figure 1-5: UI (2)

* This is the Registration page, the user is required to register in the application to access the attendance tracking system. The user is required to enter their, Name, Phone number, ID, Email & Password.

**Admin home page:**

Graphical user interface, application

Description automatically generatedFigure 1-6: UI (3)

* This page represents the homepage for Admin. The Admin can view attendance, receive notifications about the student’s attendance, view report, view database, and manage the attendance manually.

**Parents home page:**

Graphical user interface, application

Description automatically generatedFigure 1-7: UI (4)

* This page represents the homepage for Parents. The Parents can receive notifications and view their child’s attendance.

**Notifications:**

Graphical user interface, text, application, chat or text message

Description automatically generatedGraphical user interface, text, application, chat or text message

Description automatically generatedFigure 1-8: (a)

Figure 1-9 : (b)

* Notifications will be sent to the Parents and to the School Faculty

**Reports:**

Graphical user interface, application

Description automatically generated

Figure 2-0: UI (5)

* This page represents the Reports in the systems on a weekly basis, this page can only be viewed by the Admin.

Present is shown in Green.

Absent is shown in Red.

Leave is shown in Orange.

**Manage Attendance:**

Graphical user interface, text, application, chat or text message

Description automatically generatedGraphical user interface, text, application, chat or text message

Description automatically generated

Figure 2-1: (a)

Figure 2-2 : (b)

* The school admin can manage the attendance of each student in this page, by clicking the “edit” button it refers them to another page where they can change the student’s attendance for that week.

**Student Attendance:**

Graphical user interface, text, application, chat or text message

Description automatically generated

Figure 2-3: UI (6)

* This page shows the attendance of each student. By clicking the “view” button, it refers the user to the student’s attendance for the month. This page is only shown to the school admin.

**Student Attendance:**

Graphical user interface, application, calendar

Description automatically generated

Figure 2-4: UI (7)

To view the interactive UI prototype of the application visit:

https://www.figma.com/proto/hTe4JgYRFx5Omj3RODLruX/RKM?node-id=0%3A1&scaling=scale-down&page-id=0%3A1

* This page can be viewed by the School Admin and the Parents. It shows the students overall attendance throughout the month.

# CHAPTER 5: Implementation Plan

## Description of Implementation

Initially, we will create a dataset of the students before the recognition process, this includes the student's name, ID number, grade, and images in various poses, these images are given as an input to the system.[2]

Face recognition will detect a face and then identifies of those detected face images with the existing database looking for a match which then can be communicated with an external application that tells the school or parents if the student is present or absent.[2]

Diagram

Description automatically generated

Figure 2-5: Implementation

## Attendance system

A facial recognition attendance system employs facial recognition technology to automatically recognize and verify a person based on their facial traits for the students. The data is recorded and stored in real-time by the system. The system take the attendance by face recognition of the students, coordinates are determined and then the system maps the endpoints and intersections of the face recognition.

We're utilizing a native react app to do face recognition for an attendance system. The attendance face recognition application will help students to:

1. Recognize when students enter the bus.
2. This application helps administration and parents in keeping track of their children & students.
3. knows if students are late or absent
4. The time when students arrive at school or enter buses will be recorded.

## User Interface (React Native)

User Interface Design for any application should be very simple. We should have only a few clicks or navigation among the features when using the application to avoid hassle. In this application, there are two main screens, the Login and Home screens. The login page is the first page which appears when the user uses the application. In that page, if he is a new user, he can sign up or if he is an existing user, he can login with the credentials.

If the user wants to use the RKM application, they must download the application from the play store, install and register it by providing login information. Once, they register the registered information is stored on the server and can be validated, checking the valid credentials for the next time they login with the application. The next screen is the homepage where the users can select features and use the app.

A screenshot of a cell phone

Description automatically generated with medium confidenceA person jumping in the air

Description automatically generated with medium confidenceDiagram

Description automatically generatedThese are the splash pages that will appear only the first time a user opens the application. Figure 2-6: User Interface (React Native)

Graphical user interface

Description automatically generatedGraphical user interface

Description automatically generatedThe user then will be directed to the Start screen and will be asked to choose the level of access, after that they will either login or signup. Figure 2-7: User Interface (React Native)

Chart

Description automatically generatedGraphical user interface

Description automatically generated with medium confidenceSignup and Login pages. Figure 2-8: User Interface (React Native)

After selecting student profile, new page appears i.e. Student Profile in which courses are shown along with the attendance percentages. Figure 2-9: Attendance system interface

Graphical user interface, application

Description automatically generatedGraphical user interface, application

Description automatically generated

## Programming language and technology

For the proposed system for the application, and database to communicate with each other, we will use React Native to program the backend of the system, while using a tool like Firebase database to find, change and store all the needed data for the students, we will be using React Native and JavaScript programming languages as a frontend for the users, like the school or the parents. we will also be using some hardware like a high-quality camera and a mobile for testing.[2]

## IDE’s, Tools and Technologies:

* AWS Rekognition API
* Android Studio: It consists of all Android SDK tools to design, develop, maintain, test, debug and publish our app.
* Android Software Development Kit (SDK)
* SDK Manager: It is one of the main tools to maintain the updates of all the installed components required to run the project. It also notifies us when the project is not compatible with device or any other compatibility issues and to download any component that is required.
* Firebase will be used to connect to the database and host the application.
* Visual studio code
* Java SE Development Kit (JDK)
* Expo Cli quickstart
* NPM Package manager
* Node.js 12
* Android Virtual Devices (AVDs)

## Languages used:

* React Native was mainly used for creating the frontend of the project, the language is flexible enough to maintain code complexity, test, implementation, integration and support.
* Javascript
* Node.js

## Facial Recognition Set Up

Before we start writing our app, we would like to mention the API we will be using for face detection. AWS Rekognition API provides face detection and face recognition functionality via a cloud-based API. This allows us to send an HTTP request containing either an image or a URL of an existing image on the web and receive data about any faces detected in the image. It analyzes millions of images and videos within minutes and augment human visual review tasks with artificial intelligence (AI). We will be creating an attendance app with facial recognition features. It will have both server and client-side (React Native) components.

In this progress of the project, the facial detection in the app is responsible for the following:

* Scanning and connecting to AWS Rekognition peripheral.
* Asking for the user’s name.
* Asking the user to take a selfie to check if their face is registered.

There will be three pages to test the facial recognition system:

* First page will ask if the user wants to register a new face ID or to verify one.
* A page to register a new face ID
* Another page is for asking the user to take a selfie to verify if their face is registered.

Graphical user interface

Description automatically generatedGraphical user interface, application

Description automatically generated

When you connect to a peripheral, it will ask for your full name, you can access your camera and gallery by clicking on the *Capture Image* button. the image is sent to the API to check if the face is similar to one that is previously registered.

Figure 3-0: Face Recognition pages interface

## Database Set Up

A database is a logically organized collection of structured data stored electronically in a computer system. A database is usually administered by a database management system.

Data may then be accessed, managed, updated, regulated, and organized.

A database is a data structure that organizes and stores data. A database for a face recognition system, for example, might have tables for attendance data, leave time, and everything else about each student. Each of these tables would have its own set of fields that are relevant to the data it stores.

In a database setting, a database management system is used to store, manipulate, and manage data. Most data base management system packages allow users to run SQL queries to execute tasks such as database creation, data storage, and data updating.

Figure 3-1: Database

Graphical user interface, text, application, email

Description automatically generated

Figure 3-2: Database

Graphical user interface, text, application, email

Description automatically generated

Text

Description automatically generatedFigure 3-3: firebase configurations

# CHAPTER 6: Restrictions

The functionality to preview the captured image is implemented using a library called **react-native-image-picker**that enables to capture a picture from the device's camera or to upload an image from the gallery, the library was successfully installed but we have faced some issues getting the files to interact with our current project files.

Figure 3-4: Implementation of the image picker in the Face.js file

Text

Description automatically generated

Text

Description automatically generated

Figure 3-5: **AndroidManifest.xml** file

Adding these permissions to the **AndroidManifest.xml**file is what will enable to capture a picture from the device's camera or to upload an image from the gallery.

Additionally, connecting the database from firebase to our application did not work, so due to time restrictions, we weren’t able to implement them on time.

# CHAPTER 7: Conclusion and Results

To conclude, our project supplies a system that, through face recognition technology, facilitate and automates the proves of documenting and controlling the attendance of students, our technology uses facial recognition methods to estimate the limitation of the present systems, it progresses with the best type of cameras to be able to record student images perfectly, and finally, with this system we have created an application using React Native that provide all the necessities for users to use the system in the most simple way possible, this feature helps parents & school admins to take attendance whenever they enter or leave the busses and send information between parents safely quickly, and this will ensure that the student will be safer and none of them will be missing. Although we had some restrictions that prevent us from completing the application we believe if we had more time, the application will work perfectly.

**CHAPTER 8: References**

[1] Bharadwaj, R., & Rao, T. (2019). Attendance Management Using Facial Recognition. In International Journal of Innovative Technology and Exploring Engineering (IJITEE) (pp. 2278–3075). Retrieved from website: https://www.ijitee.org/wp-content/uploads/papers/v8i6/F3965048619.pdf

[2] Bhatti, K., Mughal, L., Khuhawar, F., & Memon, S. (2018). Smart Attendance Management System Using Face Recognition. EAI Endorsed Transactions on Creative Technologies, 5(17), 159713. https://doi.org/10.4108/eai.13-7-2018.159713

[3] https://www.facebook.com/jason.brownlee.39. (2019, May 30). A Gentle Introduction to Deep Learning for Face Recognition. Retrieved from Machine Learning Mastery website: https://machinelearningmastery.com/introduction-to-deep-learning-for-face-recognition/

[4] Face Recognition System. (n.d.). Retrieved from www.cdac.in website: <https://www.cdac.in/index.aspx?id=cs_bi_Face_Recognition>

[5] Edgell, J. (2013). 4 Limitations of Facial Recognition Technology. Retrieved from Technology Solutions That Drive Government website: <https://fedtechmagazine.com/article/2013/11/4-limitations-facial-recognition-technology>

[6] Bézier, P. E. (1971). Example of an Existing System in the Motor Industry: The Unisurf System. Proceedings of the Royal Society of London. Series A, Mathematical and Physical Sciences, 321(1545), 207–218. Retrieved from <https://www.jstor.org/stable/77846?seq=1>

[7] dnraj. (2012, March 22). project. Retrieved March 14, 2021, from <https://www.slideshare.net/dnraj/project-12123327>

[8] Avalon Face Recognition, Temperature Monitoring & Access Control System - Standard. (n.d.). Retrieved April 11, 2021, from avalonnetworks.com website: https://avalonnetworks.com/avalon-face-recognition-temperature-monitoring-access-control-system-standard

[9] SmartFace Platform | Facial Recognition Platform. (n.d.). Retrieved April 11, 2021, from Innovatrics website: https://www.innovatrics.com/face-recognition-solutions/?gclid=CjwKCAjw07qDBhBxEiwA6pPbHgOx6Bh8aS87aa1uJbZatYXLpD\_xW9hgguzQITNu-FwKN5KnmYvnZxoCkK8QAvD\_BwE

[10] Detect and Recognize Faces and Facial Features with Luxand FaceSDK. (2021). Retrieved 11 April 2021, from https://www.luxand.com/facesdk/?utm\_source=google&utm\_medium=cpc&utm\_campaign=face-recognition&gclid=CjwKCAjw07qDBhBxEiwA6pPbHrRS3YzM1Naa3x6o5Xj4O9sTJ5FwYJ7VBwkCLqwAANVdwYxYDctaIBoCajEQAvD\_BwE

[11] Jamil, N., Lqbal, S., & Iqbal, N. (2019). Face recognition using neural networks. Proceedings. IEEE International Multi Topic Conference, 2001. IEEE INMIC 2001. Technology for the 21st Century. https://doi.org/10.1109/inmic.2001.995351

[12] Wolf, L. (2009). Face Recognition, Geometric vs. Appearance-Based. Encyclopedia of Biometrics, 347–352. <https://doi.org/10.1007/978-0-387-73003-5_92>

[13] Engineering@ZenOfAI. (2020, January 20). Face Recognition App In React Native using AWS Rekognition. Retrieved November 11, 2021, from ZenOf.AI website: https://medium.com/zenofai/face-recognition-app-in-react-native-using-aws-rekognition-c10b188a6413

[14] How to Create a Face Detection App With React Native. (n.d.). Retrieved from Code Envato Tuts+ website: https://code.tutsplus.com/tutorials/how-to-create-a-face-detection-app-with-react-native--cms-26491

**CHAPTER 9: Appendix**

Text

Description automatically generated

Figure 3-6: App.js code shows screen navigations between pages of the App.

Text

Description automatically generated

Figure 3-7: Face.js includes codes of the facial recognition set up.

Text

Description automatically generated

Figure 3-8: Implementation

Text

Description automatically generated

Figure 3-9: RegisterScreen.js



Figure 4-0: Implementation



Figure 4-1: Implementation