BIOMETERIC ATTENDANCE SYSTEM

A Project Report

Submitted in partial fulfillment of the
Requirements for the award of the Degree of

BACHELOR OF SCIENCE (INFORMATION TECHNOLOGY)

By

MARIAMMAL YADAV

Under the esteemed guidance of

GEETANJALI YATNALKAR

Lecturer

DEPARTMENT OF INFORMATION TECHNOLOGY

VIVEKANAND EDUCATION SOCIETY'S
COLLEGE OF ARTS, SCIENCE AND COMMERCE

Affiliated to University of Mumbai

CHEMBUR MUMBAI – 400071

MAHARASHTRA

2021-2022

Proforma for the Approval Project Proposal

PNR No.: …… Roll No.: 4

1. Name of the Student

MARIAMMAL YADAV

2. Title of the Project

BIOMETRIC ATTENDANCE SYSTEM

3. Name of the Guide

GEETANJALI YATNALKAR

4. Teaching Experience of the Guide

21 Years

5. Is this your first submission? Yes ☐ No ☐

Signature of Student Signature of Guide

Date: …….. Date: ……..

Signature of the Coordinator

Date: …….

VIVEKANAND EDUCATION SOCIETY'S
COLLEGE OF ARTS, SCIENCE AND COMMERCE

Affiliated to University of Mumbai

CHEMBUR-MAHARASHTRA-400071

DEPARTMENT OF INFORMATION TECHNOLOGY

CERTIFICATE

This is to certify that the project entitled, " **BIOMETERIC ATTENDANCE SYSTEM** ", is bonafide work of **MARIAMMAL YADAV** submitted in partial fulfillment of the requirements for the award of degree of BACHELOR OF SCIENCE in INFORMATION TECHNOLOGY from University of Mumbai.

Internal Guide **Coordinator**

External Examiner

Date: **College Seal**

ABSTRACT

In industrial and domestic applications attendance registering is important at each and every moment. Many faces a lot of problems due to lack of proper attendance monitoring system. In this project we use Fingerprint Sensor (R307) which senses the Fingerprint of a particular person; a led gets activated whenever a person places his finger on the sensor. Then the fingerprint is stored in cloud with id no. Many people can store their fingerprints. Then next time any person puts their finger on the sensor it checks there are any matching fingerprints or not. If his fingerprint matches with any of the stored fingerprints then the LCD display shows which person it is and the time & date of checking.

In this model, all the fingerprints are stored each and every time someone places his finger. User can connect the system wirelessly with the cloud and monitor the process. When the code is running on the computer, data sent by R307 fingerprint module is received and stored on the cloud and displayed in serial monitor and 16*2 LCD display module.

This study has mainly focused to develop IOT based biometric attendance system, that is able to keep record of attendance and count the data for daily purpose. In this project we are going to design Fingerprint Sensor Based Biometric Attendance System using Arduino. Simply we will be interfacing fingerprint sensor with Arduino, LCD Display & SD Card Module to design the desired project. In this project, we are using fingerprint Module and Arduino to take and keep attendance data and records.

This presents a simple and portable approach to student attendance in the form of an Internet of Things (IOT) based system that records the attendance using fingerprint based biometric scanner and stores them securely over cloud. This system aims to automate the cumbersome process of manually taking and storing student attendance records. It will also prevent proxy attendance, thus increasing the reliability of attendance records. The records are securely stored and can be reliably retrieved whenever required by the teacher.

Proper attendance recording and management has become important in today's world as attendance and achievement go hand in hand. Attendance is one of the work ethics valued by employers. Most of the educational institutions and government organizations in developing countries still use paper-based attendance method for maintaining the attendance records. There is a need to replace these traditional methods of attendance recording with biometric attendance system. The unique nature of fingerprint makes it ideal for use in attendance management systems. Besides being secure, Fingerprint based attendance system will also be environment friendly. Fingerprint matching is widely used in forensics for a long time. It can also be used in applications such as identity management and access control. This review incorporates the problems of attendance systems presently in use, working of a typical fingerprint-based attendance system, study of different systems, their advantages, disadvantages and comparison based upon important parameters.

ACKNOWLEDGEMENT

We thank the people who were a part of this project in numerous ways, people who gave their unending support right from the stage the project idea was conceived.

The four things that go on to make a successful endeavour are dedication, hard work, patience and correct guidance.

We would like to thank our Principal **Dr. Mrs. Anita Kanwar** who has always been the source of inspiration.

We are also thankful to **Mrs. Shital Patil** our In-charge coordinator who was very much kind enough to give us an idea and guide us throughout our project work.

We take this opportunity to offer sincere thanks to **[Geetanjali Yatlnalkar].**

We thankful to all teaching staff (I.T) who shared their experience and gave their suggestion for developing our project in better way.

We are also thankful to Mrs. Nandini for helping us out in Project Documentation.

DECLARATION

I hereby declare that the project entitled, "BIOMETERIC ATTENDANCE SYSTEM" done at place where the project is done, has not been in any case duplicated to submit to any other university for the award of any degree. To the best of my knowledge other than me, no one has submitted to any other university. The project is done in partial fulfilment of the requirements for the award of degree of BACHELOR OF SCIENCE (INFORMATION TECHNOLOGY) to be submitted as final semester project as part of our curriculum.

MARIAMMAL YADAV

Name and Signature of the Student

TABLE OF CONTENTS

CHAPTER 1: INTRODUCTION

Introduction

In the World of Technology, Biometrics plays an effective role in identifying Human beings. Through this project, we will develop a unique system that can identify students for attendance purpose using their fingerprints.

In this project, we are going to design a Fingerprint Sensor Based Biometric Attendance System using Arduino. Simply we will be interfacing fingerprint sensor with **Arduino Uno & Fingerprint module** to design the desired project. In this project, we used the fingerprint Module and Arduino to take and keep attendance data and records.

Biometric Attendance systems are commonly used systems to mark the presence in offices and schools. This project has a wide application in school, college, business organization, offices were marking of attendance is required accurately with time. By using the fingerprint sensor, the system will become more secure for the users.

Biometric techniques can be used to solve these problems. Biometric is derived from two Greek roots “bios” meaning life and “metrics” meaning measurement. Biometric technology identifies a person uniquely based on his/her characteristics which can be physiological or behavioural. Among the various biometric techniques, there are nine main biometric techniques which are widely used. These include fingerprint, face, hand vein, hand geometry, iris, retinal pattern, voice print, signature, and facial thermo grams. Comparison of different biometric techniques has shown that fingerprint biometric is a reliable,

mature and legally accepted biometric technique. Therefore, Fingerprint based attendance system can be used for identification of large number of students in universities and also for attendance monitoring of employees in organizations.

As you can see, the final product of this project can be built as a portable device. In the registration process, the user has to provide their fingerprint two times. That means each user has to take their finger away from the scanner and place it back on again in order to record. During the process of taking attendance, we need to press the start button (blue pushbutton) to start the process. The LCD display prompts the message “Place the Finger” to let us know the process has started. Then the user has to place their finger on the scanner to mark attendance. Once the fingerprint is matched, the message “Present” will prompt and the blue LED will turn ON. If there isn’t a match, the red LED will flash ON and the display shows the error message “Finger not Found.”

IoT based Biometric Attendance system using Arduino and Things board and Adafruit Fingerprint Sensor Library.

Few years back if you were to tell someone that the Geyser and bedroom lights in your home are connected to internet, they would be baffled and might even criticize it as over-engineered products. But today with the advent of IoT, Smart cities etc the idea no longer sounds strange, we have devices around us that have become smarter by being able to communicate with the internet.

1.1 Background

In the World of Technology, Biometrics plays an effective role in identifying Human beings. Through this project, you will develop a unique system that can identify students for attendance purpose using their fingerprints. In this project, we are going to design a Fingerprint Sensor Based Biometric Attendance System using Arduino. Simply we will be interfacing fingerprint sensor with Arduino, LCD Display & RTC Module to design the desired project. In this project, we are using the fingerprint Module and Arduino to take and keep attendance data and records. Biometric Attendance systems are commonly used systems to mark the presence in offices and schools. This project has a wide application in school, college, business organization, offices were marking of attendance is required accurately with time. By using the fingerprint sensor, the system will become more secure for the user.

1.2 Objectives

In developing this system, some project objectives had been specified. The main purpose of this project is to improve the current existing student attendance system that in use by most of the colleges by develop a fingerprint-based student attendance system. Some objectives of this project had been identified and listed below.

i. To replace the current existing student attendance system process to fully-computerized and automated student attendance system.
ii. To develop a desktop-based application that obtains the student fingerprint every time they attend the classes for attendance marking purpose.
iii. To develop a web-based student attendance system in displaying every student attendance result effectively.
iv. To generate reports regarding to the student attendance in order to assist the lecturer/staff in analyze and tracking the student attendance.
v. To eliminate the chances for student to ask their buddy sign attendance for them through the implementation of fingerprint attendance system.
vi. To provide easier method in evaluate and analyze the student performance based on their attendance since the system will record the attendance more accurately and efficiently with minimum possible error.

1.3 Purpose and scope

1.3.1 Purpose:

The main purpose of the microcontroller is to enroll and search the fingerprint. In enrolling, this controller read the template from the fingerprint sensor and enroll the ID number. This display the ID number on serial monitor. And then, this check

fingerprint with the stored template in the searching process. If the fingerprint is correct, the display valued are shown in excel. Otherwise, the controller don't give any output

1. The biometric attendance work is to capture the finger print of the user and mark the attendance for that particular date

2. Once the user keeps in fingerprint sensor, it will capture his/her fingerprint and displays his name with date and time.

3. The attendence captured will be stored in a file under that user's name.

4. The access for attendence record will be given to the faculty members.

5. This will be time saving and also be secure as there will be no proxy issues since fingerprint is captured.

1.3.2 Scope:

The future with biometric technologies is becoming faster, easier and more secure. Biometric attendance system using Arduino uno is very useful for many industries and offices. It's easy, cost effective and works very well. Hence the future scope of this technology is wide spread and quite essential in both domestic and industrial applications. Attendance systems are commonly used systems to mark the presence in offices and schools. From manually marking the attendance in attendance registers to using high-tech applications and biometric systems, these systems have improved significantly. The traditional process of manually taking and maintaining student attendance is highly inefficient and time consuming. The attendance monitoring system based on biometric authentication has a potential to streamline the whole process. An Internet of Things (IoT) based portable biometric attendance system can prove to be of great value to educational institutions in this regard as it

proves to be highly efficient and secure. The cost involved in making this system is quite less, when compared to conventional biometric attendance system. The use of cloud computing to store the attendance records makes all the data easy to access and retrieve as end when required by the teachers. The use of fingerprint scanner ensures the reliability of the attendance record. The system, due to its lack of complexity, proves to be easy to use and user friendly. Hence the future scope of this technology is wide spread and quite essential in both domestic and industrial applications

1.4 Applicability:

The purpose of the research is to constructs an online presence system that combines fingerprint modules. The fingerprint module is used as the system's main input as well as a security tool as an entrance to get access to the entire system. The fingerprint sensor can determine the fingerprint stored in the database with an average response time. Proper attendance recording and management has become important in today's world as attendance and achievement go hand in hand. Biometric student attendance system increases the efficiency of the process of taking student attendance. This paper presents a simple and portable approach to student attendance in the form of an Internet of Things (IOT) based system that records the attendance using fingerprint based biometric scanner and stores them securely. This system aims to automate the cumbersome process of manually taking and storing student attendance records. If we talk about the attendance system in universities and schools, lecturers did that work manually. Lecturers take the attendance and update it manually in the database. If we combine the fingerprint sensor with IOT (Internet of Things) than we can do it automatically and there is no need to do it by lecturers. We can use IOT and finger print sensor for better performance. IOT data is directly store on server in real time so we can

access it from anywhere and anytime which will provide us with better proficiency and flexibility

1.5 Organization of report:

The basic project idea is to design a Biometric attendance system to get data of the students who are present and absent. It will be authorized by the admin for collecting the date and student will be accessible for only marking the attendance.

CHAPTER 2 : SURVEY OF TECHNOLOGIES

2.1 Justification of selection of Technology

The Arduino software is easy-to-use for beginners, yet flexible enough for advanced users. To sum up, the Arduino is widely-used and supported by a vibrant community; It supports lots of microcontrollers; and it is open-sourced, enabling anyone from beginners to experts to quickly develop and even manufacture their products. The board has regular innovation and a bug fix in the design of the board to make the board suitable for the project's use. The Arduino UNO board is considered as the most used board and a standard board. When there is a lots of other micro-controller boards available, we choose Arduino because it is Less Expensive, Cross Platform which means one can install it in any OS like Linux, Windows ,Mac OS, Macintosh. And also it is simple and easy to learn even a armature electronic hobbyist or a school student can understand the concepts and learn it fast and do their own projects in a very short time

CHAPTER 3 : SYSTEM ANALYSIS

3.1 Existing System:

This system completes the requirement of the department, as this system contains easy accessibility for the students and the professors. Now fingerprint system are widely used everywhere, offices, university are accepting the new method of attendance. The advanced attendance involves the database which helps in recording the data and preserving the attendance sheet in the google sheet. This saves paper and space and also reduces the threat of records been misplaced.

Attendance system saves the time in the class which is very help for the professors. Even records the accurate time of student entering and leaving the class. It only needs wifi and few memories to store the data from the database. This includes the enrollment, remove, enable, disable and adding the new device to system. System also has the wide application.

3.2 Proposed System:

In this paper, the design and development of a portable classroom attendance system based on fingerprint biometric is presented. Among the salient aims of implementing a biometric feature into a portable attendance system is security and portability. The circuit of this device is strategically constructed to have an independent source of energy to be operated, as well as its miniature design which made it more efficient in term of its portable capability. Rather than recording the attendance in writing or queuing in front of class equipped with fixed fingerprint or smart card reader. This paper introduces a portable fingerprint based biometric attendance system which addresses the weaknesses of the existing paper-based

attendance method or long-time queuing. In addition, our biometric fingerprint-based system is encrypted which preserves data integrity.

3.3 Hardware Requirements:

3.3.1 Prototype Modelling

In this Fingerprint Sensor Based Biometric Attendance System using Arduino, we used a Fingerprint Sensor module to authenticate a true person or employee by taking their finger input in the system. Here we are using 4 push buttons to register new fingerprint or delete stored fingerprint or match stored fingerprint. The 4 push buttons are used as an input unit for these tasks. Similarly, RTC Module DS3231 is used for registering scanning/entering/existing time of the user.

The LCD displays the time record and every function happening via push button. Buzzer indicates different functions and happening whenever an interrupt is detected. The LED is used for power indication.

3.3.2 Main features of the prototype

The features of the developed prototype are:

LCD display (showing the username and time & date)

- Upto 256 fingerprints can be stored and checked when needed
- Fingerprint is stored in cloud digitally
- After fingerprint checking data is displayed in a serial monitor from Things board account

- Fingerprint data can be stored and deleted as many times as one wants
- Buzzer and LCD indicates fingerprint is stored and checked
- The date and time is shown when fingerprint is stored and checked along with username.

3.3.3 Overview of the Project

Working of this **fingerprint attendance system project** is fairly simple. First of all, the user needs to enrol fingerprints of the user with the help of push buttons. To do this, user need to press ENROLL key and then LCD asks for entering ID for the fingerprint to save it in memory by ID name. So now user needs to enter ID by using UP/DOWN keys. After selecting ID, user needs to press OK key (DEL key). Now LCD will ask to place finger over the fingerprint module. Now user needs to place his finger over finger print module and then the module takes finger image. Now the LCD will say to remove finger from fingerprint module, and again ask to place finger again. Now user needs to put his finger again and module takes an image and convert it into templates and stores it by selected ID into the finger print module's memory. Now the user will be registered and he/she can feed attendance by putting their finger over fingerprint module. By the same method, all the users will be registered into the system.

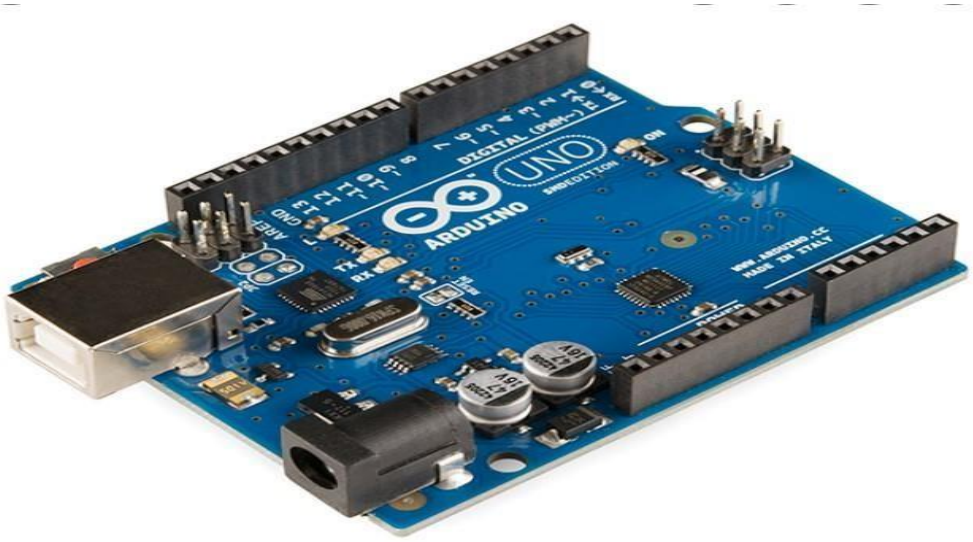
Now if the user wants to remove or delete any of the stored ID or fingerprint, then he/she need to press DEL key. Once delete key is pressed LCD will ask to select ID that need to be deleted. Now user needs to select ID and press OK key (same DEL key). Now LCD will let you know that fingerprint has been deleted successfully.

3.4 Hardware Requirements:

Serial No.	List of Components	Quantity
1	Arduino Uno	1
2	Fingerprint Module	1
3	16x2 LCD with 12C Module	1
4	Pushbuttons	2
5	LEDs	2
6	RTC Module	1
7	Breadboard and Jumper Wires	1

3.4.1 Arduino Uno

The main purpose of the microcontroller is to enroll and search the fingerprint. In enrolling, this controller read the template from the fingerprint sensor and enroll the ID number. This display the ID number on serial monitor. And then, this controller check the fingerprint with the stored template in the searching process. If the fingerprint is correct, the display valued are shown in excel. Otherwise, the controller don’t give any output.

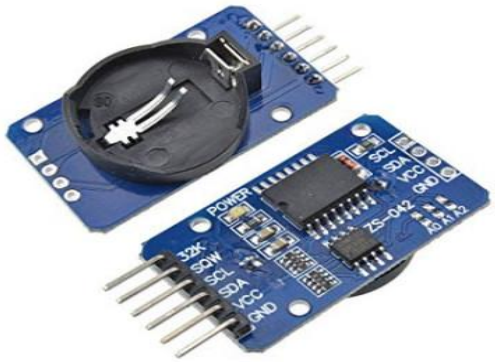


3.4.2 R305 Scanner

An optical fingerprint sensor is used in this system. This sensor read the fingerprint pattern. The scan image is converted as template and saved in memory.This is an interface which can be directly connected to the Arduino UART. TheR305/R307 fingerprint scanner has a TTL UART

R305/R307	Arduino Mega	Arduino Uno
GND	GND	GND
Vcc	5V	5V
Rx	18	3
Tx	19	2

3.4.3 DS3231 RTC Module



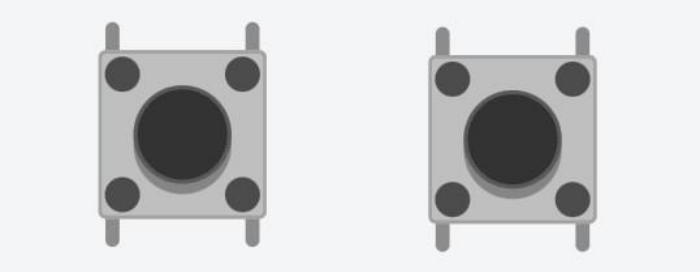
RTC modules are simply TIME and DATE remembering systems which have battery setup which in the absence of external power keeps the module running. This keeps the TIME and DATE up to date. So we can have accurate TIME and DATE from RTC module whenever we want

3.4.4 LCD With 12C Module

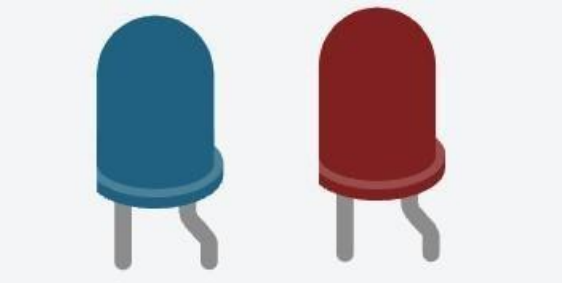
As the name implies, the LCD module communicates with Arduino through I2C communication.

I2C LCD	Arduino Mega	Arduino Uno
GND	GND	GND
Vcc	5V	5V
SDA	20	A4
SCL	21	A5

3.4.5 Pushbuttons



3.4.6 LEDs



3.5 Software Requirements:

All of the fingerprint sensor's functions are controlled by the Adafruit fingerprint sensor library. The library contains the methods and functions to drive the fingerprint scanner. Also DS3231 Library is used for RTC Module.

3.6 Planning and scheduling:

GANTT CHART

FIG A:

Task	Processed by	Start Date	Days	End date
Background	Mariammal	18-Jul	1	18-Jul
Objective	Monalisa	19-Jul	1	19-Jul
Purpose	Mariammal	20-Jul	2	21-Jul
Scope	Monalisa	21-Jul	2	22-Jul
Applicability	Mariammal	22-Jul	1	23-Jul
Definition and abbreviation	Monalisa	23-Jul	1	23-Jul
Organization of report	Mariammal	23-Jul	2	25-Jul

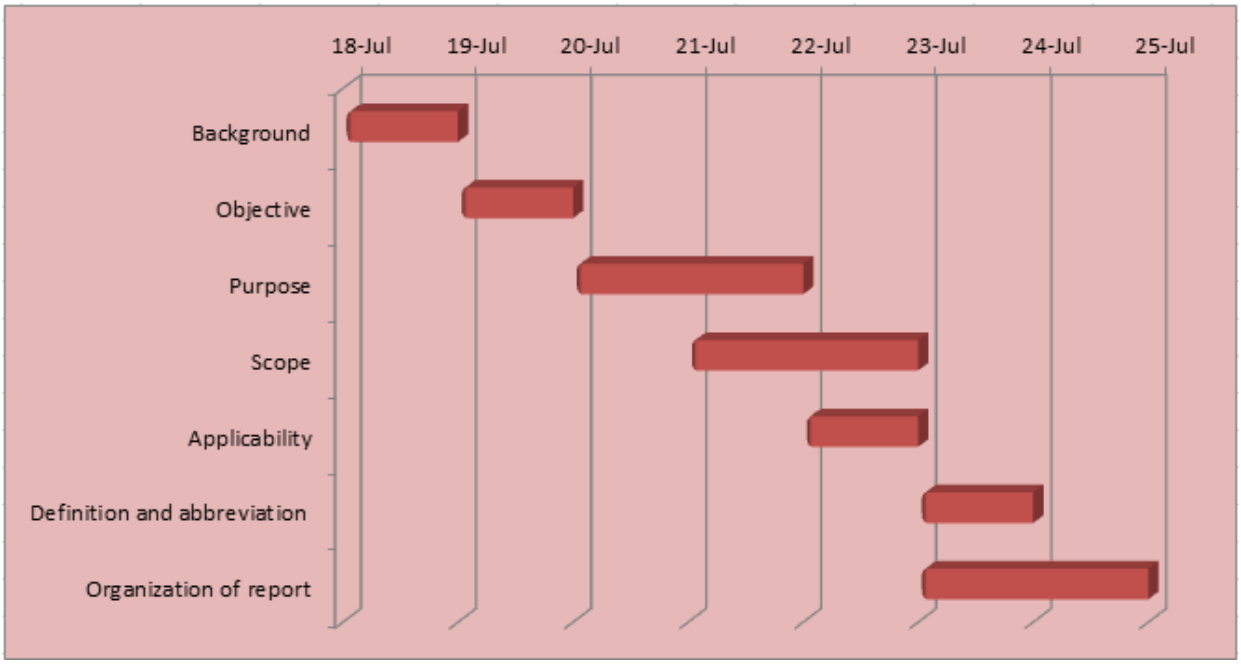


FIG B:

Task	Processed by	Start Date	Days	End date
Existing system	Mariammal	1-Aug	3	3-Aug
Proposed system	Monalisa	3-Aug	3	6-Aug
Requirement analysis	Mariammal	6-Aug	1	7-Aug
Hardware Requirement	Monalisa	8-Aug	3	10-Aug
Software requirement	Mariammal	11-Aug	2	12-Aug
Jusification of program	Monalisa	13-Aug	3	15-Aug

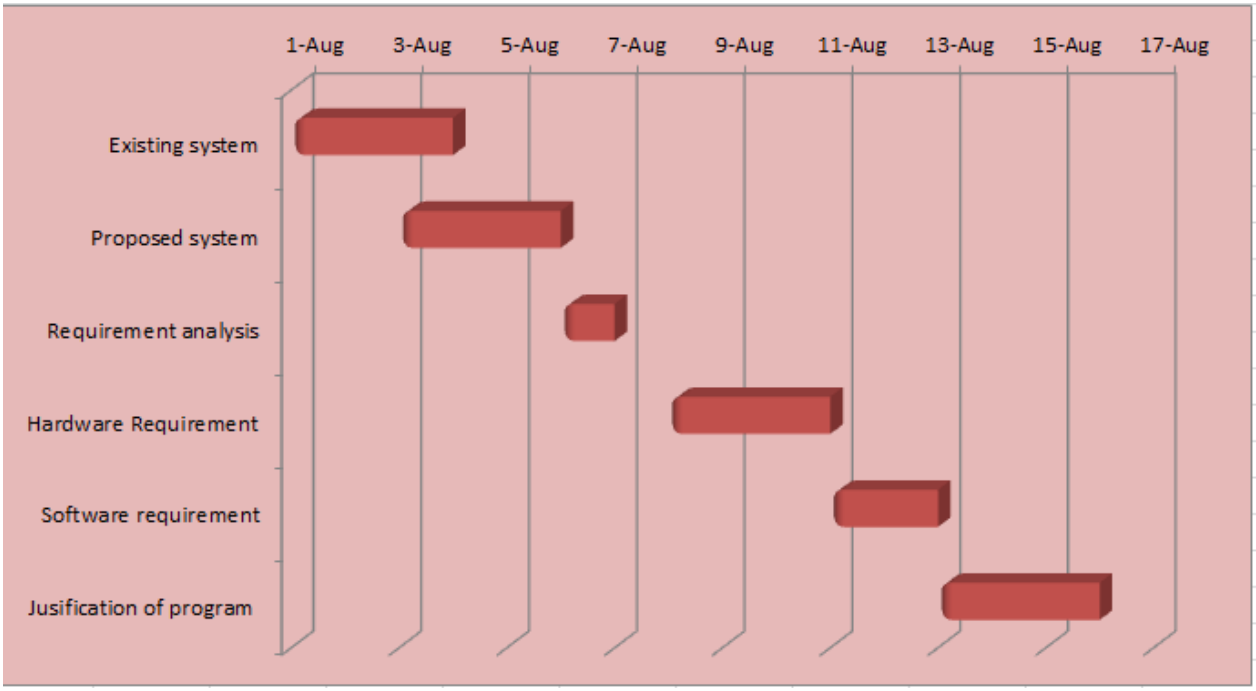


FIG C:

Task	Processed By	Start date	Days	End date
Module Design	Mariammal	20-Aug	3	22-Aug
Data Flow Diagram	Monalisa	23-Aug	3	26-Aug

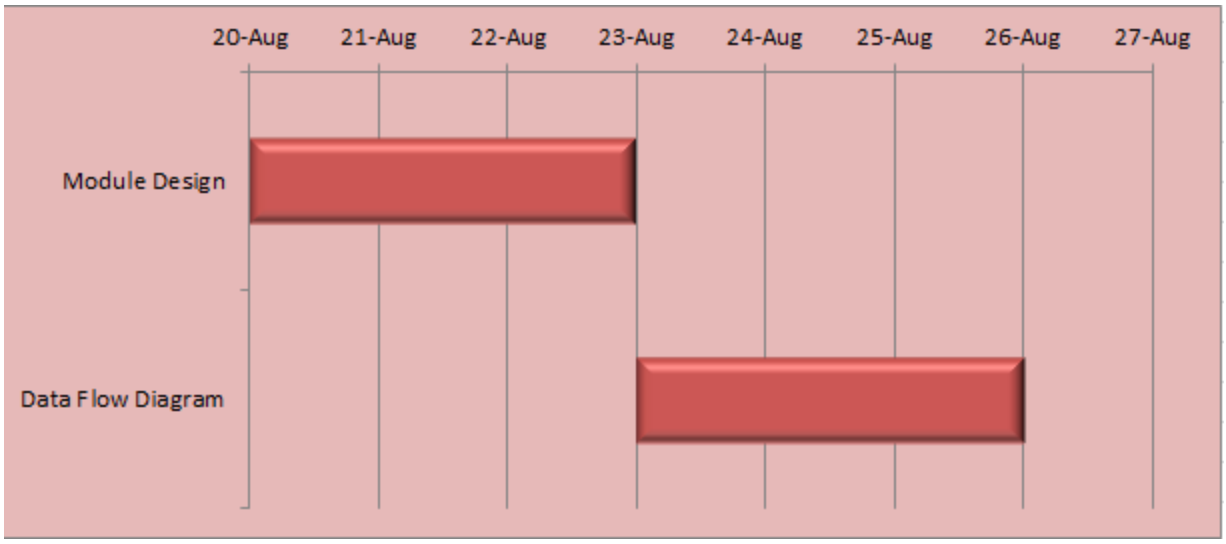
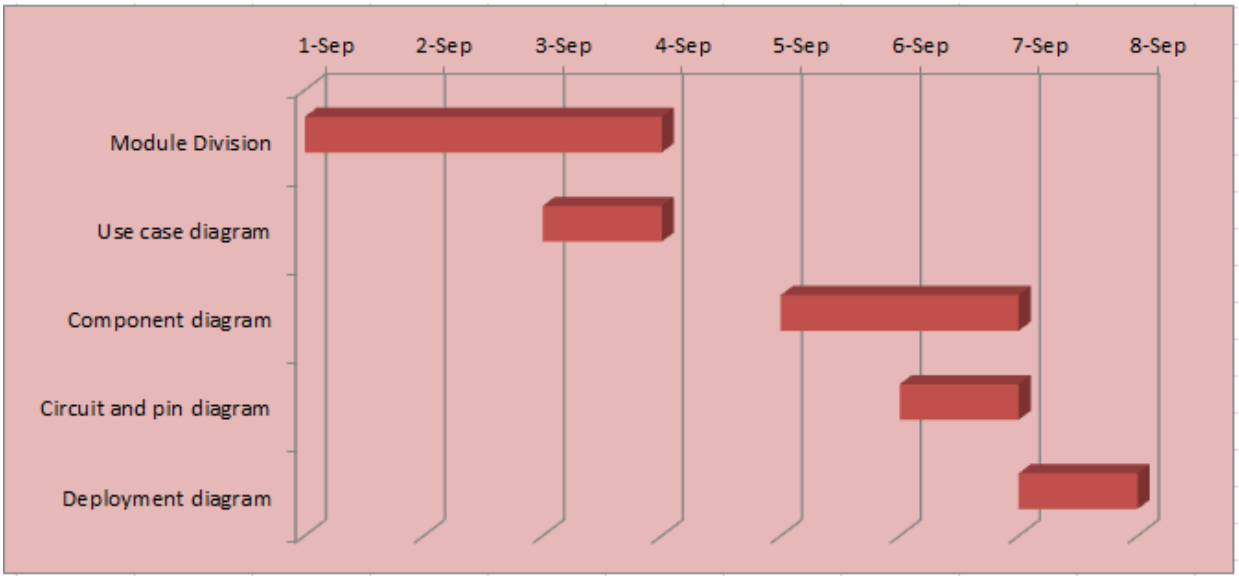


FIG D:

Task	Processed By	Start date	Days	End date
Module Division	Mariammal	1-Sep	3	3-Sep
Use case diagram	Monalisa	3-Sep	1	4-Sep
Component diagram	Mariammal	5-Sep	2	6-Sep
Circuit and pin diagram	Monalisa	6-Sep	1	7-Sep
Deployment diagram	Mariammal	7-Sep	1	8-Sep



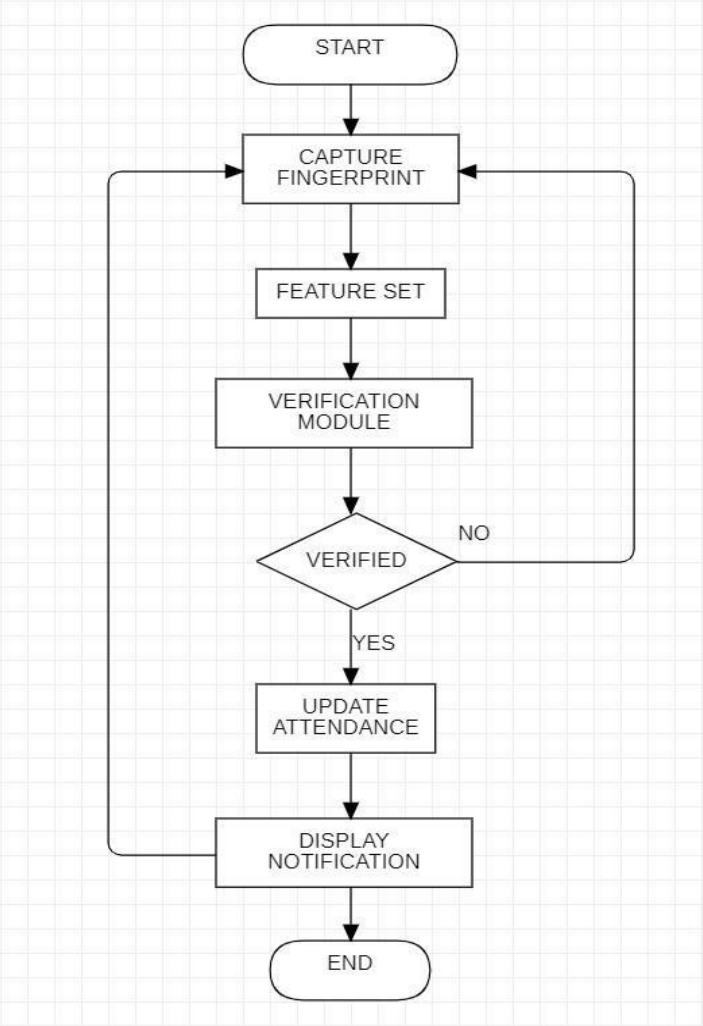
3.7 Justification of Platform:

In an educational system, the teachers call out the name of each and every student and mark the attendance. This causes time wastage during lecture time. This becomes more and more important where number of students in a class is very large. Managing the attendance data is also very difficult such a large group. The other way is that the teacher must pass the attendance sheet around the class for the students to sign. These methods have a major drawback because the students tend to answer or sign for their friends. In educational institutions, attendance and academic success are directly related. Therefore, to have a proper attendance management system is important. In developing countries, most of the educational institutions and government organizations still use paper-based attendance method to keep and save the attendance. Most employers value work attendance for their ethics. Biometrics is the emerging technology used for automatic identification of a person based on biological characters such as fingerprint, iris, facial recognition, etc. The fingerprint verification system is commonly used biometric technique.
Fingerprint based technique use computer to store and verify fingerprints

CHAPTER 4 : SYSTEM DESIGN

4.1 Module Division:

The main objective of our system is to reduce the human efforts and save time This system provides an advancement to the home appliances. The user can control the device with a mobile application with enabled wifi.



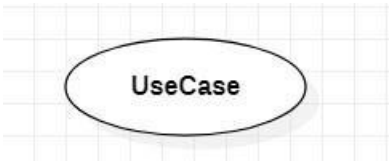
4.2 UML DIAGRAMS:

4.2.1 Use case:

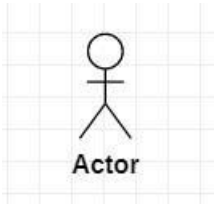
In the Unified Modelling Language (UML), a use case diagram can summarize the details of a system's users (also known as actors) and their interactions with the system.

The various components used in use case diagram are:

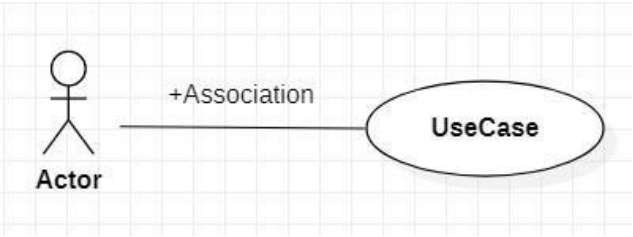
Use cases: Horizontally shaped ovals that represent the different uses that a user might have.



Actors: Stick figures that represent the people actually employing the use cases.

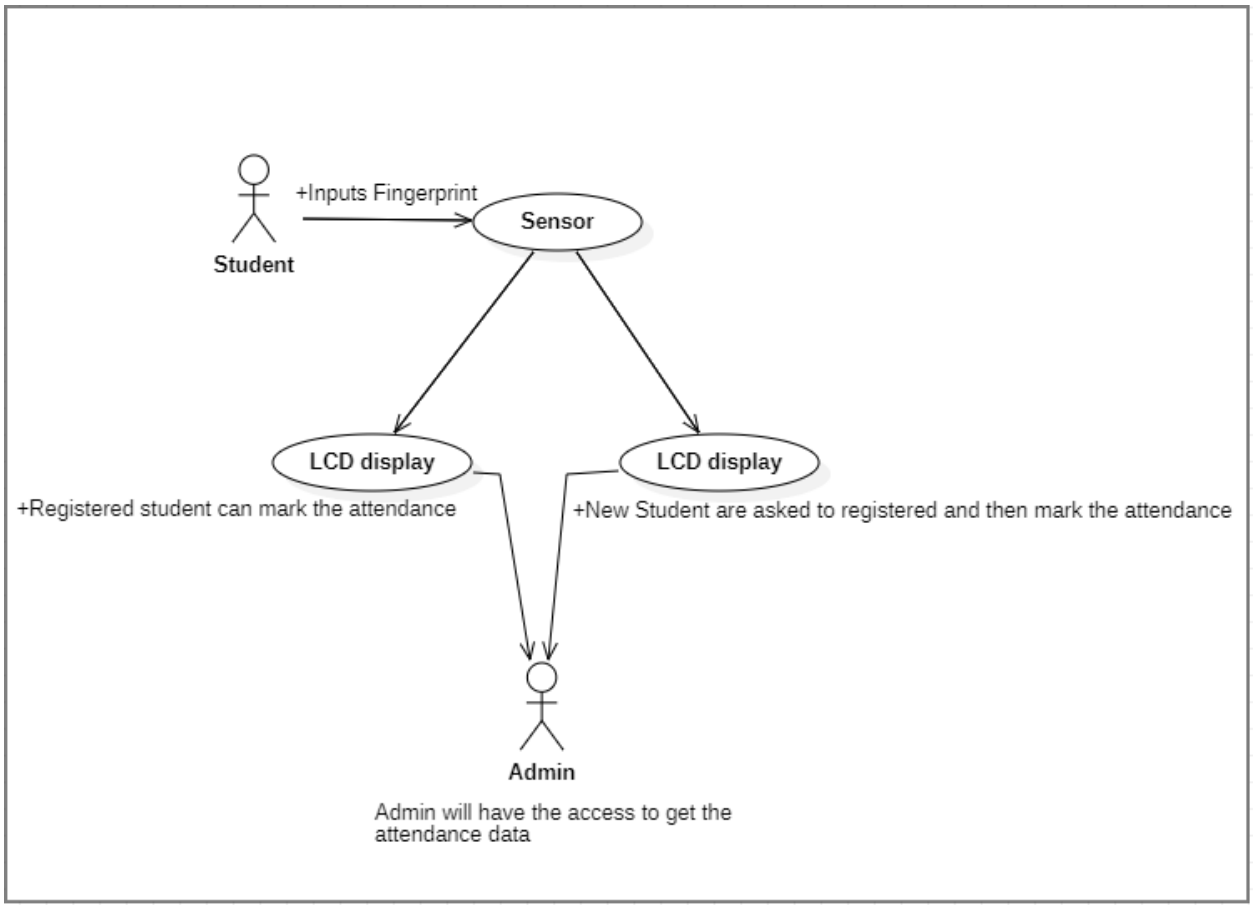


Associations: A line between actors and use cases. In complex diagrams, it is important to know which actors are associated with which use cases.



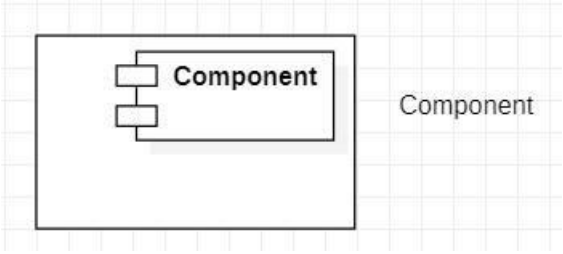
System boundary boxes: A box that sets a system scope to use cases. All use cases outside the box would be considered outside the scope of that system.

USE CASE DIAGRAM



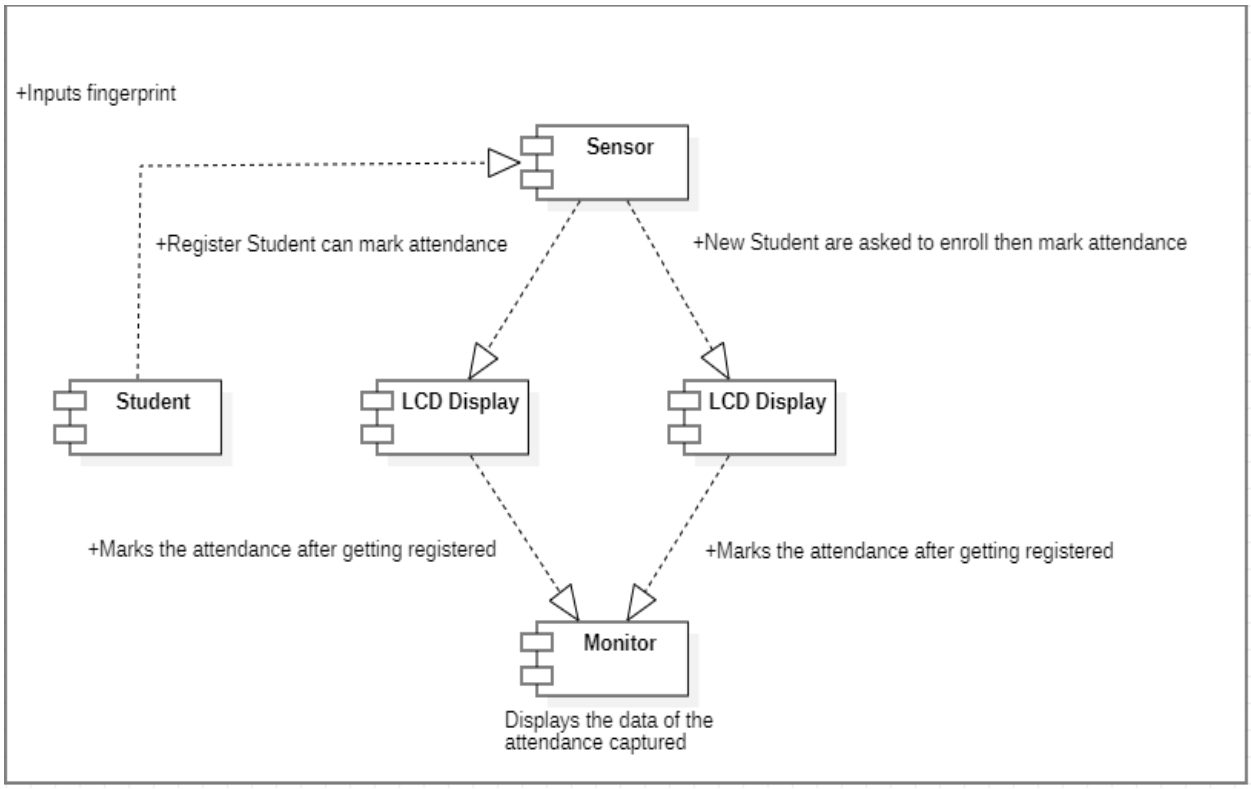
4.2.2 Component diagram:

Component Diagrams and Deployment Diagrams are closely related. Component Diagrams are used to describe the Components. A Component is a logical unit block of the system, a slightly higher abstraction that classes. It is represented as a rectangle with a smaller rectangle in the upper right corner with tabs or the word written above the name of the component to help distinguish it from a class.



Interface:

An interface (small circle or semi-circle on a stick) describes a group of operations used (Required or Created) (Provided) by components. A full circle represents an interface created or provided by the component. A semi-circle represents a required interface, like a person's input.

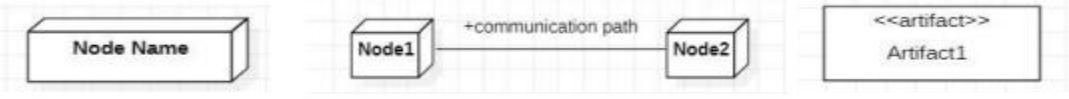


4.2.3 Deployment diagram:

Deployment diagrams are used for describing the hardware components, where software components are deployed. Deployment diagrams shows how they are deployed in hardware.

NODE

ARTEFACTS

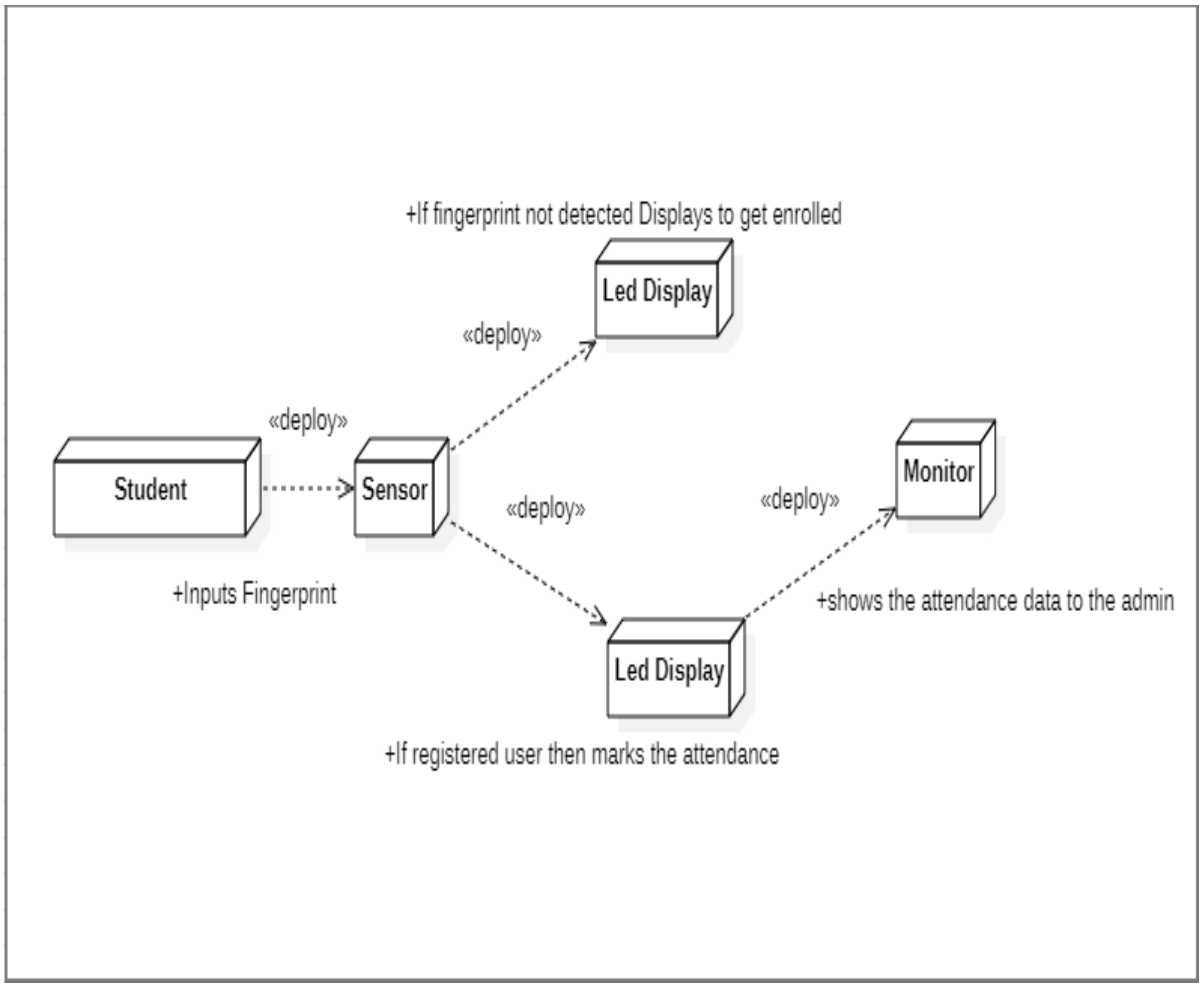


<<device>>

<<execution environment>>

Deployment Specifications

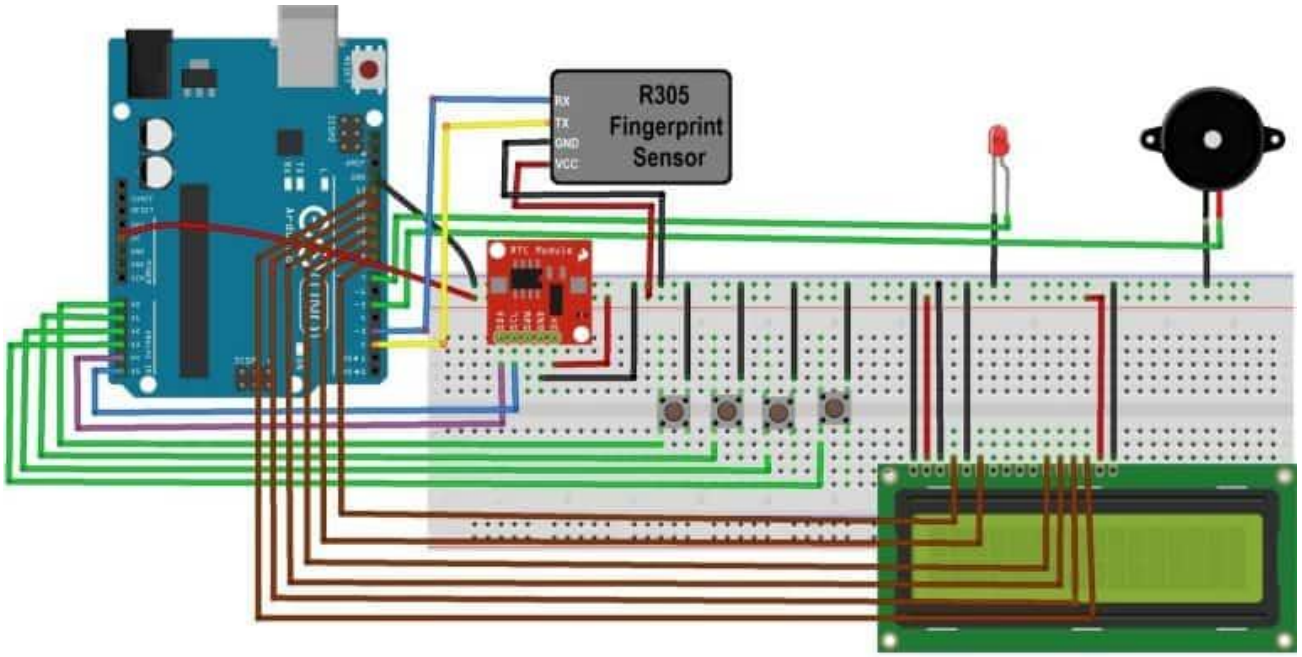
Deployment of Artefacts



4.3 Circuit Diagram

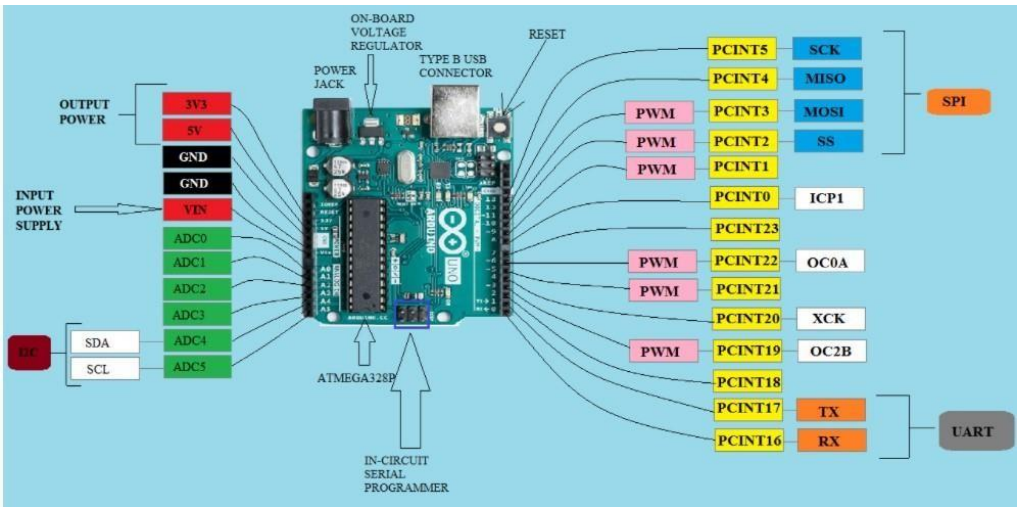
4.3.1Circuit Diagram:

A circuit diagram is **a graphical representation of an electrical circuit**. A circuit diagram also called an electrical diagram, elementary diagram or electronic schematic is defined as a simplified graphical representation of an electrical circuit.



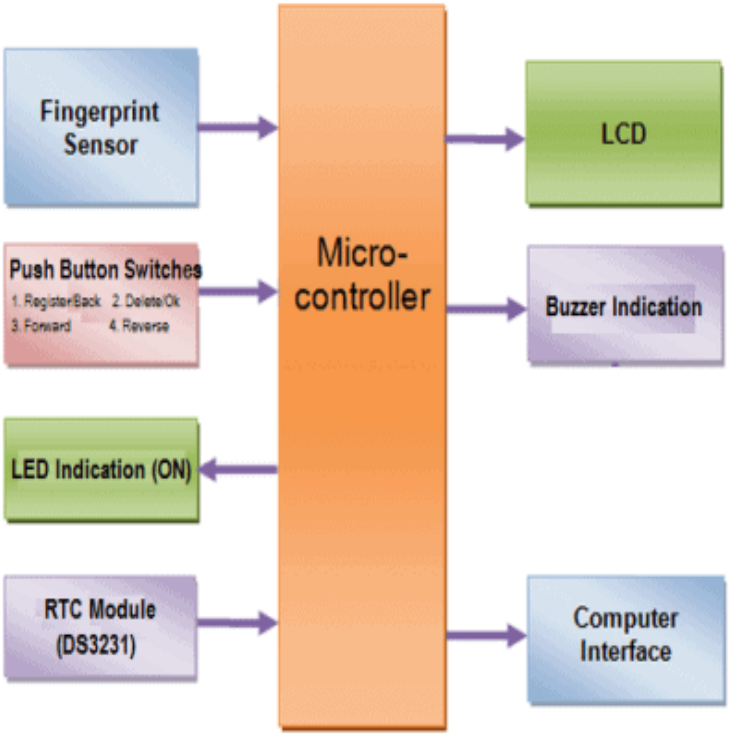
4.3.2 Pin Diagram:

In electronics, a pinout (sometimes written "pin-out") is a cross-**reference** between the contacts, or pins, of an electrical connector or electronic component, and their functions.



4.3.3 Block Diagram

In this Fingerprint Sensor Based Biometric Attendance System using Arduino, we used a Fingerprint Sensor module to authenticate a true person or employee by taking their finger input in the system. Here we are using 4 push buttons to register new fingerprint or delete stored fingerprint or match stored fingerprint. The 4 push buttons are used as an input unit for these tasks. Similarly, RTC Module DS3231 is used for registering scanning/entering/existing time of the user.The LCD displays the time record and every function happening via push button. Buzzer indicates different functions and happening whenever an interrupt is detected. The LED is used for power indication.


Fingerprint Sensor
Push Button Switches
1. Register/Back 2. Delete/Ok
3. Forward 4. Reverse
LED Indication (ON)
RTC Module (DS3231)
Micro-controller
LCD
Buzzer Indication
Computer Interface

CHAPTER 5 : IMPLEMENTATION AND TESTING

5.1 CODE

```
#include "Adafruit_Fingerprint.h" //fingerprint library header file
#include<EEPROM.h> //command for storing data
#include<LiquidCrystal.h> //lcd header file
LiquidCrystal lcd(8,9,10,11,12,13);
#include <SoftwareSerial.h>
SoftwareSerial fingerPrint(2, 3); //for tx/rx communication between arduino & r305 fingerprint sensor

#include <Wire.h>
#include "RTClib.h" //library file for DS3231 RTC Module
RTC_DS3231 rtc;

uint8_t id;
Adafruit_Fingerprint finger = Adafruit_Fingerprint(&fingerPrint);

#define register_back 14
#define delete_ok 15
#define forward 16
#define reverse 17
#define match 5
#define indFinger 7
#define buzzer 5

#define records 10 // 10 for 10 user
```

```
int user1,user2,user3,user4,user5,user6,user7,user8,user9,user10;

DateTime now;

void setup()
{
delay(1000);
lcd.begin(16,2);
Serial.begin(9600);
pinMode(register_back, INPUT_PULLUP);
pinMode(forward, INPUT_PULLUP);
pinMode(reverse, INPUT_PULLUP);
pinMode(delete_ok, INPUT_PULLUP);
pinMode(match, INPUT_PULLUP);
pinMode(buzzer, OUTPUT);
pinMode(indFinger, OUTPUT);
digitalWrite(buzzer, LOW);
if(digitalRead(register_back) == 0)
{
digitalWrite(buzzer, HIGH);
delay(500);
digitalWrite(buzzer, LOW);
lcd.clear();
lcd.print("Please wait !");
lcd.setCursor(0,1);
lcd.print("Downloding Data");
```

```
Serial.println("Please wait");
Serial.println("Downloding Data..");
Serial.println();

Serial.print("S.No. ");
for(int i=0;i<records;i++)
{
digitalWrite(buzzer, HIGH);
delay(500);
digitalWrite(buzzer, LOW);
Serial.print(" User ID");
Serial.print(i+1);
Serial.print(" ");
}
Serial.println();
int eepIndex=0;
for(int i=0;i<30;i++)
{
if(i+1<10)
Serial.print('0');
Serial.print(i+1);
Serial.print(" ");
eepIndex=(i*7);
download(eepIndex);
eepIndex=(i*7)+210;
download(eepIndex);
eepIndex=(i*7)+420;
```

```
download(eepIndex);
eepIndex=(i*7)+630;
download(eepIndex);
eepIndex=(i*7)+840;
download(eepIndex);
eepIndex=(i*7)+1050;
download(eepIndex);
eepIndex=(i*7)+1260;
download(eepIndex);
eepIndex=(i*7)+1470;
download(eepIndex);
eepIndex=(i*7)+1680;
download(eepIndex);
Serial.println();
}
}
if(digitalRead(delete_ok) == 0)
{
lcd.clear();
lcd.print("Please Wait");
lcd.setCursor(0,1);
lcd.print("Reseting.....");
for(int i=1000;i<1005;i++)
EEPROM.write(i,0);
for(int i=0;i<841;i++)
EEPROM.write(i, 0xff);
lcd.clear();
```

```
lcd.print("System Reset");
delay(1000);
}

lcd.clear();
lcd.print(" Fingerprint ");
lcd.setCursor(0,1);
lcd.print("Attendance System");
delay(2000);
lcd.clear();

digitalWrite(buzzer, HIGH);
delay(500);
digitalWrite(buzzer, LOW);
for(int i=1000;i<1000+records;i++)
{
if(EEPROM.read(i) == 0xff)
EEPROM.write(i,0);
}

finger.begin(57600);
Serial.begin(9600);
lcd.clear();
lcd.print("Finding Module..");
lcd.setCursor(0,1);
delay(2000);
if (finger.verifyPassword())
```

```
{
Serial.println("Found fingerprint sensor!");
lcd.clear();
lcd.print(" Module Found");
delay(2000);
}
else
{
Serial.println("Did not find fingerprint sensor :(");
lcd.clear();
lcd.print("Module Not Found");
lcd.setCursor(0,1);
lcd.print("Check Connections");
while (1);
}

if (! rtc.begin())
Serial.println("Couldn't find RTC");

// rtc.adjust(DateTime(F(__DATE__), F(__TIME__)));

if (rtc.lostPower())
{
Serial.println("RTC is NOT running!");
// following line sets the RTC to the date & time this sketch was compiled
rtc.adjust(DateTime(2018, 6, 7, 11, 0, 0));
// This line sets the RTC with an explicit date & time, for example to set
```

```
// June 7, 2018 at 11am you would call:
// rtc.adjust(DateTime(2018, 6, 7, 11, 0, 0));
}
lcd.setCursor(0,0);
lcd.print(" Press Match to ");
lcd.setCursor(0,1);
lcd.print(" Start System");
delay(3000);

user1=EEPROM.read(1000);
user2=EEPROM.read(1001);
user3=EEPROM.read(1002);
user4=EEPROM.read(1003);
user5=EEPROM.read(1004);
lcd.clear();
digitalWrite(indFinger, HIGH);

}

void loop()
{
now = rtc.now();
lcd.setCursor(0,0);
lcd.print("Time: ");
lcd.print(now.hour(), DEC);
lcd.print(':');
lcd.print(now.minute(), DEC);
```

```
lcd.print(':');
lcd.print(now.second(), DEC);
lcd.print(" ");
lcd.setCursor(0,1);
lcd.print("Date: ");
lcd.print(now.day(), DEC);
lcd.print('/');
lcd.print(now.month(), DEC);
lcd.print('/');
lcd.print(now.year(), DEC);
lcd.print(" ");
delay(500);
int result=getFingerprintIDez();
if(result>0)
{
digitalWrite(indFinger, LOW);
digitalWrite(buzzer, HIGH);
delay(100);
digitalWrite(buzzer, LOW);
lcd.clear();
lcd.print("ID:");
lcd.print(result);
lcd.setCursor(0,1);
lcd.print("Please Wait....");
delay(1000);
attendance(result);
lcd.clear();
```

```
lcd.print("Attendance ");
lcd.setCursor(0,1);
lcd.print("Registered");
delay(1000);
digitalWrite(indFinger, HIGH);
return;
}
checkKeys();
delay(300);
}

// dmyyhms - 7 bytes
void attendance(int id)
{
int user=0,eepLoc=0;
if(id == 1)
{
eepLoc=0;
user=user1++;
}
else if(id == 2)
{
eepLoc=210;
user=user2++;
}
else if(id == 3)
{
```

```
eepLoc=420;
user=user3++;
}
else if(id == 4)
{
eepLoc=630;
user=user4++;
}
else if(id == 5)
{
eepLoc=0;
user=user5++;
}
else if(id == 6)
{
eepLoc=840;
user=user5++;
}
else if(id == 7)
{
eepLoc=1050;
user=user7++;
}
else if(id == 8)
{
eepLoc=1260;
user=user8++;
```

```
}
else if(id == 9)
{
eepLoc=1470;
user=user9++;
}
else if(id == 10)
{
eepLoc=1680;
user=user8++;
}
/*else if(id == 5) // fifth user
{
eepLoc=840;
user=user5++;
}*/
else
return;

int eepIndex=(user*7)+eepLoc;
EEPROM.write(eepIndex++, now.hour());
EEPROM.write(eepIndex++, now.minute());
EEPROM.write(eepIndex++, now.second());
EEPROM.write(eepIndex++, now.day());
EEPROM.write(eepIndex++, now.month());
EEPROM.write(eepIndex++, now.year()>>8 );
EEPROM.write(eepIndex++, now.year());
```

```
EEPROM.write(1000,user1);
EEPROM.write(1001,user2);
EEPROM.write(1002,user3);
EEPROM.write(1003,user4);
// EEPROM.write(4,user5); // figth user
}

void checkKeys()
{
if(digitalRead(register_back) == 0)
{
lcd.clear();
lcd.print("Please Wait");
delay(1000);
while(digitalRead(register_back) == 0);
Enroll();
}

else if(digitalRead(delete_ok) == 0)
{
lcd.clear();
lcd.print("Please Wait");
delay(1000);
delet();
}
}
```

```
void Enroll()
{
int count=1;
lcd.clear();
lcd.print("Enter Finger ID:");

while(1)
{
lcd.setCursor(0,1);
lcd.print(count);
if(digitalRead(forward) == 0)
{
count++;
if(count>records)
count=1;
delay(500);
}

else if(digitalRead(reverse) == 0)
{
count--;
if(count<1)
count=records;
delay(500);
}
else if(digitalRead(delete_ok) == 0)
```

```
{
id=count;
getFingerprintEnroll();
for(int i=0;i<records;i++)
{
if(EEPROM.read(i) != 0xff)
{
EEPROM.write(i, id);
break;
}
}
return;
}

else if(digitalRead(register_back) == 0)
{
return;
}
}
}

void delet()
{
int count=1;
lcd.clear();
lcd.print("Enter Finger ID");
```

```
while(1)
{
lcd.setCursor(0,1);
lcd.print(count);
if(digitalRead(forward) == 0)
{
count++;
if(count>records)
count=1;
delay(500);
}

else if(digitalRead(reverse) == 0)
{
count--;
if(count<1)
count=records;
delay(500);
}
else if(digitalRead(delete_ok) == 0)
{
id=count;
deleteFingerprint(id);
for(int i=0;i<records;i++)
{
if(EEPROM.read(i) == id)
{
```

```
EEPROM.write(i, 0xff);
break;
}
}
return;
}

else if(digitalRead(register_back) == 0)
{
return;
}
}
}

uint8_t getFingerprintEnroll()
{
int p = -1;
lcd.clear();
lcd.print("finger ID:");
lcd.print(id);
lcd.setCursor(0,1);
lcd.print("Place Finger");
delay(2000);
while (p != FINGERPRINT_OK)
{
p = finger.getImage();
switch (p)
```

```
{
case FINGERPRINT_OK:
Serial.println("Image taken");
lcd.clear();
lcd.print("Image taken");
break;
case FINGERPRINT_NOFINGER:
Serial.println("No Finger");
lcd.clear();
lcd.print("No Finger Found");
break;
case FINGERPRINT_PACKETRECIEVEERR:
Serial.println("Communication error");
lcd.clear();
lcd.print("Comm Error");
break;
case FINGERPRINT_IMAGEFAIL:
Serial.println("Imaging error");
lcd.clear();
lcd.print("Imaging Error");
break;
default:
Serial.println("Unknown error");
lcd.clear();
lcd.print("Unknown Error");
break;
}
```

```
}

// OK success!

p = finger.image2Tz(1);
switch (p) {
case FINGERPRINT_OK:
Serial.println("Image converted");
lcd.clear();
lcd.print("Image converted");
break;
case FINGERPRINT_IMAGEMESS:
Serial.println("Image too messy");
lcd.clear();
lcd.print("Image too messy");
return p;
case FINGERPRINT_PACKETRECIEVEERR:
Serial.println("Communication error");
lcd.clear();
lcd.print("Comm Error");
return p;
case FINGERPRINT_FEATUREFAIL:
Serial.println("Could not find fingerprint features");
lcd.clear();
lcd.print("Feature Not Found");
return p;
case FINGERPRINT_INVALIDIMAGE:
```

```
Serial.println("Could not find fingerprint features");
lcd.clear();
lcd.print("Feature Not Found");
return p;
default:
Serial.println("Unknown error");
lcd.clear();
lcd.print("Unknown Error");
return p;
}

Serial.println("Remove finger");
lcd.clear();
lcd.print("Remove Finger");
delay(2000);
p = 0;
while (p != FINGERPRINT_NOFINGER) {
p = finger.getImage();
}
Serial.print("ID "); Serial.println(id);
p = -1;
Serial.println("Place same finger again");
lcd.clear();
lcd.print("Place Finger");
lcd.setCursor(0,1);
lcd.print(" Again");
while (p != FINGERPRINT_OK) {
```

```
p = finger.getImage();
switch (p) {
case FINGERPRINT_OK:
Serial.println("Image taken");
break;
case FINGERPRINT_NOFINGER:
Serial.print(".");
break;
case FINGERPRINT_PACKETRECIEVEERR:
Serial.println("Communication error");
break;
case FINGERPRINT_IMAGEFAIL:
Serial.println("Imaging error");
break;
default:
Serial.println("Unknown error");
return;
}
}

// OK success!

p = finger.image2Tz(2);
switch (p) {
case FINGERPRINT_OK:
Serial.println("Image converted");
break;
```

```
case FINGERPRINT_IMAGEMESS:
Serial.println("Image too messy");
return p;
case FINGERPRINT_PACKETRECIEVEERR:
Serial.println("Communication error");
return p;
case FINGERPRINT_FEATUREFAIL:
Serial.println("Could not find fingerprint features");
return p;
case FINGERPRINT_INVALIDIMAGE:
Serial.println("Could not find fingerprint features");
return p;
default:
Serial.println("Unknown error");
return p;
}

// OK converted!
Serial.print("Creating model for #"); Serial.println(id);

p = finger.createModel();
if (p == FINGERPRINT_OK) {
Serial.println("Prints matched!");
} else if (p == FINGERPRINT_PACKETRECIEVEERR) {
Serial.println("Communication error");
return p;
} else if (p == FINGERPRINT_ENROLLMISMATCH) {
```

```
Serial.println("Fingerprints did not match");
return p;
} else {
Serial.println("Unknown error");
return p;
}

Serial.print("ID "); Serial.println(id);
p = finger.storeModel(id);
if (p == FINGERPRINT_OK) {
Serial.println("Stored!");
lcd.clear();
lcd.print(" Finger Stored!");
delay(2000);
} else if (p == FINGERPRINT_PACKETRECIEVEERR) {
Serial.println("Communication error");
return p;
} else if (p == FINGERPRINT_BADLOCATION) {
Serial.println("Could not store in that location");
return p;
} else if (p == FINGERPRINT_FLASHERR) {
Serial.println("Error writing to flash");
return p;
}
else {
Serial.println("Unknown error");
return p;
```

```
}
}

int getFingerprintIDez()
{
uint8_t p = finger.getImage();

if (p != FINGERPRINT_OK)
return -1;

p = finger.image2Tz();
if (p != FINGERPRINT_OK)
return -1;

p = finger.fingerFastSearch();
if (p != FINGERPRINT_OK)
{
lcd.clear();
lcd.print("Finger Not Found");
lcd.setCursor(0,1);
lcd.print("Try Later");
delay(2000);
return -1;
}
// found a match!
Serial.print("Found ID #");
Serial.print(finger.fingerID);
```

```
return finger.fingerID;
}

uint8_t deleteFingerprint(uint8_t id)
{
uint8_t p = -1;
lcd.clear();
lcd.print("Please wait");
p = finger.deleteModel(id);
if (p == FINGERPRINT_OK)
{
Serial.println("Deleted!");
lcd.clear();
lcd.print("Finger Deleted");
lcd.setCursor(0,1);
lcd.print("Successfully");
delay(1000);
}

else
{
Serial.print("Something Wrong");
lcd.clear();
lcd.print("Something Wrong");
lcd.setCursor(0,1);
lcd.print("Try Again Later");
delay(2000);
```

```
return p;
}
}

void download(int eepIndex)
{

if(EEPROM.read(eepIndex) != 0xff)
{
Serial.print("T->");
if(EEPROM.read(eepIndex)<10)
Serial.print('0');
Serial.print(EEPROM.read(eepIndex++));
Serial.print(':');
if(EEPROM.read(eepIndex)<10)
Serial.print('0');
Serial.print(EEPROM.read(eepIndex++));
Serial.print(':');
if(EEPROM.read(eepIndex)<10)
Serial.print('0');
Serial.print(EEPROM.read(eepIndex++));
Serial.print(" D->");
if(EEPROM.read(eepIndex)<10)
Serial.print('0');
Serial.print(EEPROM.read(eepIndex++));
Serial.print('/');
if(EEPROM.read(eepIndex)<10)
```

```
Serial.print('0');
Serial.print(EEPROM.read(eepIndex++));
Serial.print('/');
Serial.print(EEPROM.read(eepIndex++)<<8 | EEPROM.read(eepIndex++));
}
else
{
Serial.print("--------------------------");
}

Serial.print(" ");
}
```

CHAPTER 6 : RESULT AND DISCUSSION

Working Explanation :

The working of the Fingerprint Sensor Based Biometric Attendance System. In this project, we have used a DS3231 RTC Module for time & date display. We used 1 LED for power indication, 1 buzzer for different function indication. We have interfaced 16*2 LCD which displays everything whenever the finger is placed or removed, or registering attendance or downloading data.

We have used 4 push buttons which are used to control the entire system. The functions of each button are :

1. Register/Back Button – Used for enrolling new fingerprint as well as reversing the back process or going back

2. Delete/OK Button – This Button is used for deleting the earlier stored fingerprint system as well as granting access as an OK selection.

3. Forward Button – Used for moving forward while selecting the memory location for storing or deleting fingerprints.

4. Reverse Button – Used for moving backward while selecting memory location for storing or deleting fingerprints.

CHAPTER 7 : CONCLUSION AND FUTURE

7.1 Conclusion:

In era of rapid technological developments, educational institutions need to have cost-effective and efficient systems in place such as biometric student attendance solutions to streamline operations and increase security. Biometric systems for student attendance provide a convenient way to check-in and check-out students and can bring numerous benefits to the education institutions beyond attendance tracking. Biometrics are also used for school lunch line point of service, in libraries, and any other educational scenario where accurate identification and security is needed. Biometric modalities make use of physical traits like fingerprints to identify individuals. It is possible to reset passwords but not fingerprints. Therefore, a high level of security is essential for protecting the fingerprint details of students. If hackers steal data from the server, it can cause a lot of problems for students. In conclusion, these are the main challenges that have to be overcome if biometric attendance systems are installed for student attendance monitoring in schools.

CHAPTER 8 : REFERENCES

- https://maker.pro/arduino/projects/how-to-make-a-fingerprint-based-attendance-system-with-arduino-and-r305
- https://how2electronics.com/fingerprint-biometric-attendance-system-arduino/#Downloading_Data
- https://www.youtube.com/watch?v=ZnjQe5RZXTA
- https://drive.google.com/file/d/1U36H4PWKjjFU_oz8ggm7EtOvGG9kTZBv/view