A Project Report

on

Fingerprint Based Biometric Attendance System using Arduino

Submitted in partial fulfillment of the requirement for the award of the degree of

Bachelor of Technology in Computer Science Engineering

Under The Supervision of

Dr.Amit Kumar Goel

professor

Submitted By

Shubham kumar:-18SCSE1010720
Tarneet Singh:-18SCSE1010099

SCHOOL OF COMPUTING SCIENCE AND ENGINEERING
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

GALGOTIAS UNIVERSITY, GREATER NOIDA

INDIA

December, 2021

SCHOOL OF COMPUTING SCIENCE AND ENGINEERING

GALGOTIAS UNIVERSITY, GREATER NOIDA

CANDIDATE'S DECLARATION

We hereby certify that the work which is being presented in the project, entitled **"FINGERPRINT BASED BIOMETRIC ATTENDANCE SYSTEM USING ARDUINO"** in partial fulfillment of the requirements for the award of the **Btech CSE** submitted in the School of Computing Science and Engineering of Galgotias University, Greater Noida, is an original work carried out during the period of 4 months, December 2021, under the supervision of **Dr Anit kumar Goel**, Department of Computer Science and Engineering, School of Computing Science and Engineering , Galgotias University, Greater Noida

The matter presented in the project has not been submitted by us for the award of any other degree of this or any other places.

Shubham kumar, 18SCSE1010720

Tarneet singh, 18SCSE1010099

This is to certify that the above statement made by the candidates is correct to the best of my knowledge.

Dr.Amit Kumar Goel

Professor

CERTIFICATE

The Final Thesis/Project/ Dissertation Viva-Voce examination of Shubham kumar:- 18SCSE1010720, Tarneet singh:- 18SCSE1010099 has been held on _____ and his work is recommended for the award of Btech CSE.

Signature of Examiner(s) **Signature of Supervisor(s)**

Signature of Project Coordinator **Signature of Dean**

Date: 27th December, 2021

Place: Greater Noida

Acknowledgement

It is our great fortune that we have got opportunity to carry out this project work under the supervision of **Dr. Amit kumar Goel** in the Department of **Computer Science Engineering**, Galgotias University (GU), Yamuna Expy, Opposite, Buddha International Circuit, Sector 17A, Greater Noida, approved by AICTE. We express our sincere thanks and deepest sense of gratitude to our guide for his constant support, unparalleled guidance and limitless encouragement.

We wish to convey our gratitude to Dr Amit kumar Goel, Professor, Department of Ccomputer Science And Engineering, Galgotias University and to the authority of GU for providing all kinds of infrastructural facility towards the research work.
Thanks to the fellow members of our group for working as a team.

1. SHUBHAM KUMAR (18SCSE1010720)

2. TARNEET SINGH (18SCSE1010099)

CSE

B.TECH

4TH YEAR. 7TH SEMESTER.

GALGOTIAS UNIVERSITY

BATCH OF 2018-2022.

Contents

Abstract

Biometric understudy participation framework expands the productivity of the most common way of taking understudy participation. This paper presents a straightforward and compact way to deal with understudy participation as an Internet of Things (IOT) based framework that records the participation utilizing unique mark based biometric scanner and stores them safely over cloud. This framework intends to computerize the unwieldy course of physically taking and putting away understudy participation records. It will likewise forestall intermediary participation, hence expanding the unwavering quality of participation records. The records are safely put away and can be dependably recovered at whatever point needed by the educator.

Introduction

Marking personal attendance is a tedious and time-consuming task. Considering these limitations, the model is designed in such a way that the attendees can be marked without suffering. Using this method of going to college can be beneficial at many levels: -
a) Saves time for professor, so that they can use all the talk time rather than take time to be present
b) It removes the dependency of daily transfers into the system

The main idea is to make a hardware product using arduino board. A fingerprint scanner will be attached to the board that will help students place their fingerprints before each class. It will therefore function as a biometric system.

When a student puts on fingerprints, they must work to be accepted. For example, if a student from another class tries to install a class fingerprint where he or she is not registered, the system will reject and request valid fingerprints to unlock the system. Student fingerprints will be taken twice. Just before entering the classroom and just after the class is over.
In the software section, the information that will be downloaded from the biometric system will be stored on the website. The website will be updated regularly.
Each student will have a unique student id that will be posted on the website.
To view the number of classes visited by a student, an application will be made. This app will allow students, professors and student caregivers to view their presence. This however will be a viewing app only. No one can cheat data.

The designed system automatically performs the entire institutional participation process. Ensures that data is secure and secure by avoiding access to unauthorized users

List of tables

Table Project costing

	A	B
1	**Product**	**Cost**
2	Breadboard	160
3	Jump Wire	100
4	Arduino Uno	570
5	Leds	20
6	Resistors	20
7	Push Buttons	20
8	RTC Module	230
9	Biometric Fingerprint Scanner	1400
10	LCD Display	800
11	Buzzer	120
12	Total	3440

Figure: Project Costing

List of figures

Flow diagram

Flow diagram represents a set of dynamic relationship in a system. Before a model is created, a flow diagram is designed for understanding the better of the system.

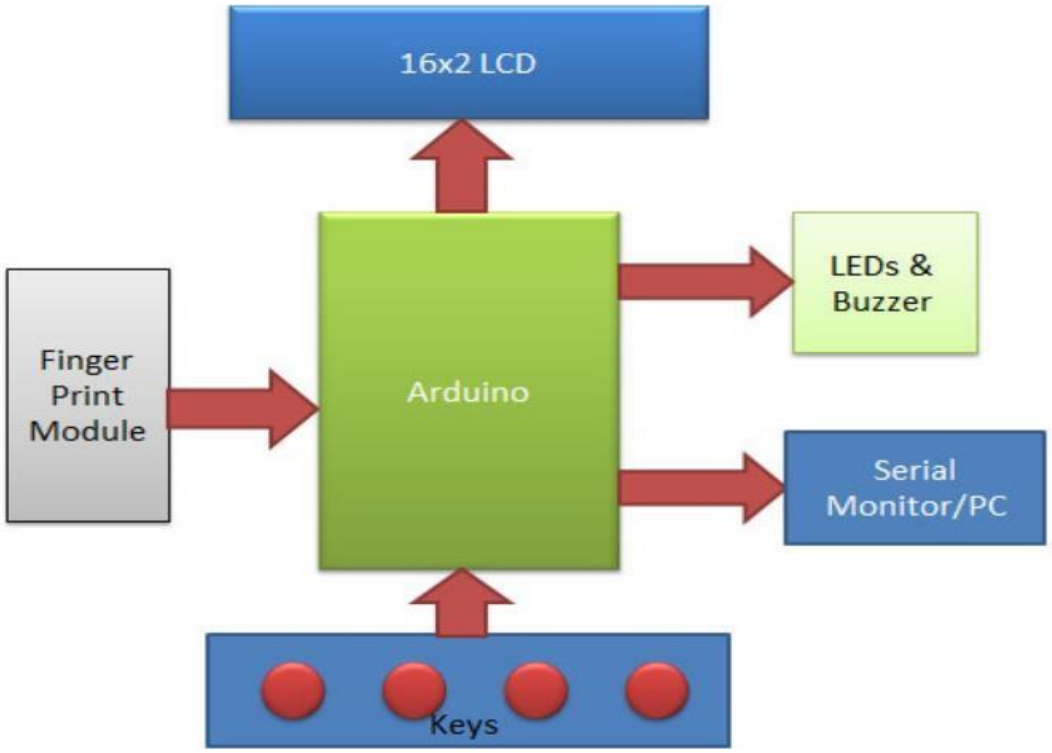


Figure: Flow Diagram

ER Diagram

ER diagram is a non-technical design method which works on a conceptual level based on the perception of the real world.

`To display the contents of a data stored, ER diagrams are used. It is used for the construction of a relational database. Therefore, ER diagram shows the relationship between the entity sets.

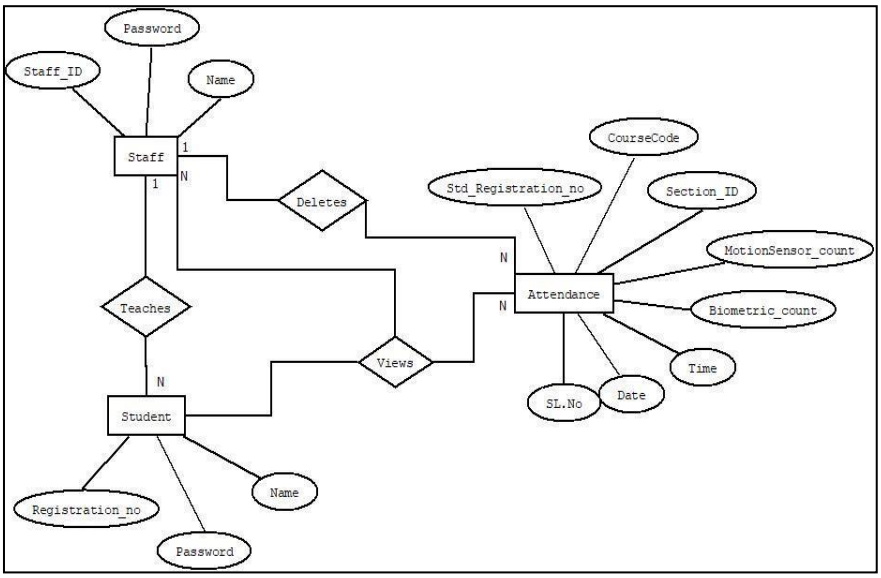


Figure: Entity relationship diagram for attendance management system

Data Flow Diagram (DFD)

DFD shows the flow of the data through the system and also used for modelling the requirements. The path of the decision represents the logical expression. It is used to represent the system at different levels of abstraction.

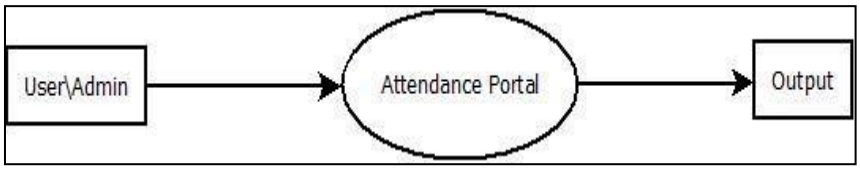
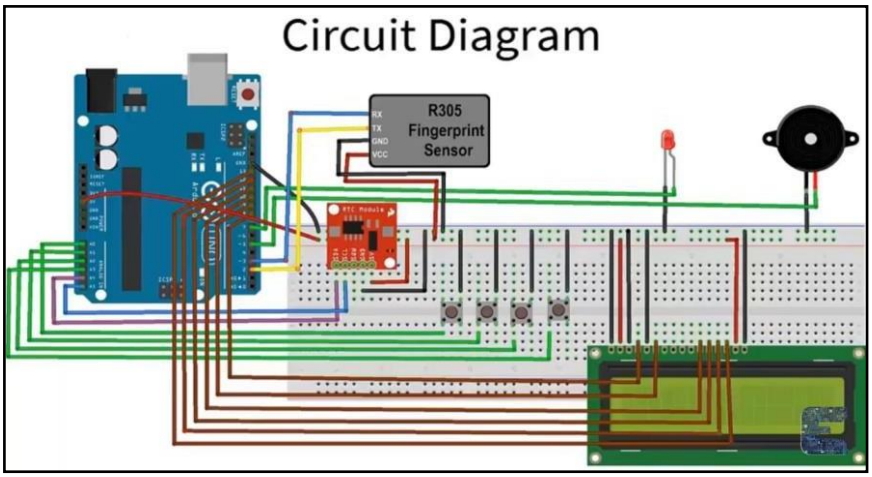
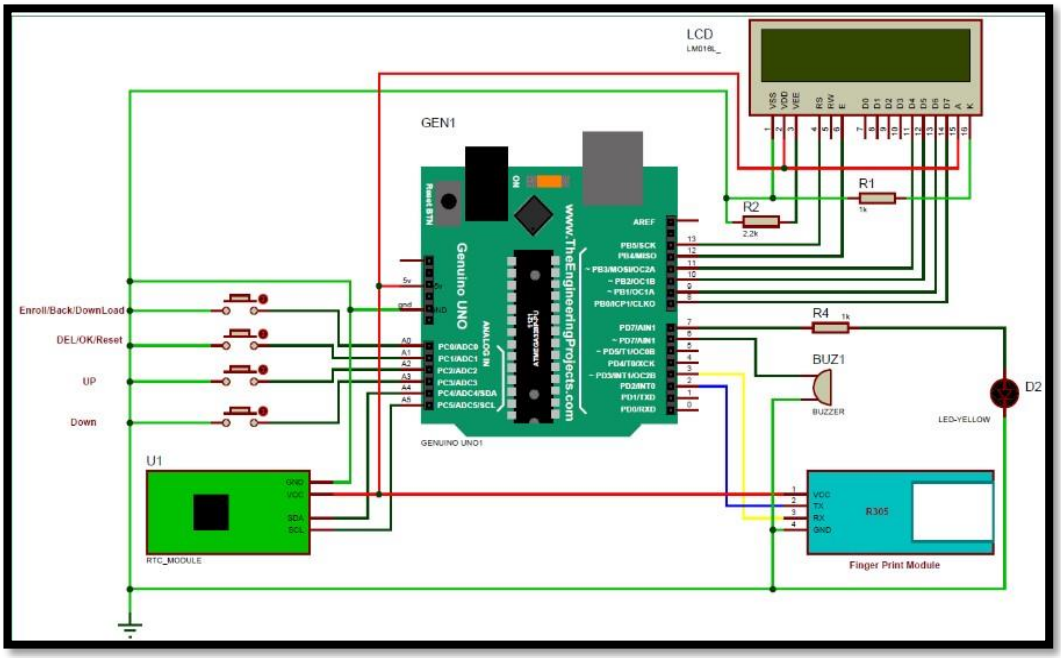


Figure: DFD for attendance management system

Circuit Diagram For Fingerprint Scanner



Detailed Circuit Diagram For Fingerprint Scanner



Tools and technologies

- Breadboard
- Arduino
- Fingerprint scanner
- RTC module
- LCD display

HARDWARE

Biometric scanner

A scanner allows the student and the professor to enter their fingerprint. Once thefingerprint is detected, it is searched in the database. Once the data is found the attendance is marked else invalid fingerprint

SOFTWARE

MIT application

MIT app inventor is an innovative beginner's introduction to programming and appcreation that transforms the complex language of text-based coding into visual, drag-and-drop building blocks

Php

It is a scripting language which is used to create connectivity to a database. Many operations are developed such as collection of data, generate dynamic page content, or send and receive cookies. Three main areas where PHP scripts are used:

i. Server-side scripting
ii. Command line scripting
iii. Writing desktop applications

Xampp Server

It is free and open source. Simple, lightweight apache distribution that makes it extremely easy for developers to create a local web server for testing and development purposes. All that is required is a web server application, database and scripting language (PHP) to make transition from a local test server to a live server.

Database

Data can be facts related to any object inconsideration. A database is a systematiccollection of data. Since the data in a database is organised, makes the data management easy. Database management system is a collection of programs which enables its users to access database, manipulate data and help in the representation of data.

MySQL

Structured Query Language (SQL) is a standard language to deal with relational databases. It can be effectively used to insert, search, update, and delete databaserecords. It is a relational database management system.

Hardware Description

ARDUINO UNO- **Arduino Uno** is a microcontroller board based on the ATmega328P datasheet. It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz ceramic resonator (CSTCE16M0V53-R0), a USB connection, a power jack, an ICSP header and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started.. You can tinker with your Uno without worrying too much about doing something wrong, worst case scenario you can replace the chip for a few dollars and start over again

"Uno" means one in Italian and was chosen to mark the release of Arduino Software (IDE) 1.0. The Uno board and version 1.0 of Arduino Software (IDE) were the reference versions of Arduino, now evolved to newer releases. The Uno board is the first in a series of USB Arduino boards, and the reference model for the Arduino platform; for an extensive list of current, past or outdated boards see the Arduino index of boards.

Figure: ARDUINO UNO

Microcontroller: The high-performance Microchip picoPower 8-bit AVR RISCbased microcontroller combines 32KB ISP flash memory with read-while-write capabilities, 1024B EEPROM, 2KB SRAM, 23 general purpose I/O lines, 32 general purpose working registers, three flexible timer/counters with compare modes, internal and external interrupts, serial programmable USART, a byte-oriented 2-wire serial interface, SPI serial port, a 6-channel 10-bit A/D converter (8-channels in TQFP and QFN/MLF packages), programmable watchdog timer with internal oscillator, and five software selectable power saving modes. The device operates between 1.8-5.5 volts.

FINGERPRINT SCANNER (R307): R307 Fingerprint Module consists of optical fingerprint sensor, high-speed DSP processor, high-performance fingerprint alignment algorithm, highcapacity FLASH chips and other hardware and software composition, stable performance, simple structure, with fingerprint entry, image processing, fingerprint matching, search and template storage and other functions.

Features:

- Perfect function: independent fingerprint collection, fingerprint registration, fingerprint comparison (1: 1) and fingerprint search (1: N) function.
- Small size: small size, no external DSP chip algorithm, has been integrated, easy to install, less fault.
- Ultra-low power consumption: low power consumption of the product as a whole, suitable for low-power requirements of the occasion.
- Anti-static ability: a strong anti-static ability, anti-static index reached 15KV above.
- Application development is simple: developers can provide control instructions, self fingerprint application product development, without the need for professional knowledge of fingerprinting.
- Adjustable security level: suitable for different applications, security levels can be set by the user to adjust.
- Finger touch sensing signal output, low effective, sensing circuit standby current is very low, less than 5uA.

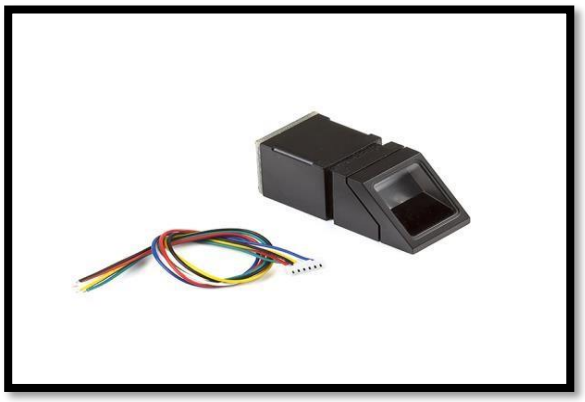


Figure: FINGERPRINT SENSOR

16*2 LCD DISPLAY:

- Operating Voltage is 4.7V to 5.3V
- Current consumption is 1mA without backlight
- Alphanumeric LCD display module, meaning can display alphabets and numbers
- Consists of two rows and each row can print 16 characters.
- Each character is build by a 5×8 pixel box
- Can work on both 8-bit and 4-bit mode
- It can also display any custom generated characters • Available in Green and Blue Backlight.

LCD modules are very commonly used in most embedded projects, the reason being its cheap price, availability and programmer friendly. Most of us would have come across these displays in our day to day life, either at PCO's or calculators. The appearance and the pinouts have already been visualized above now let us get a bit technical.

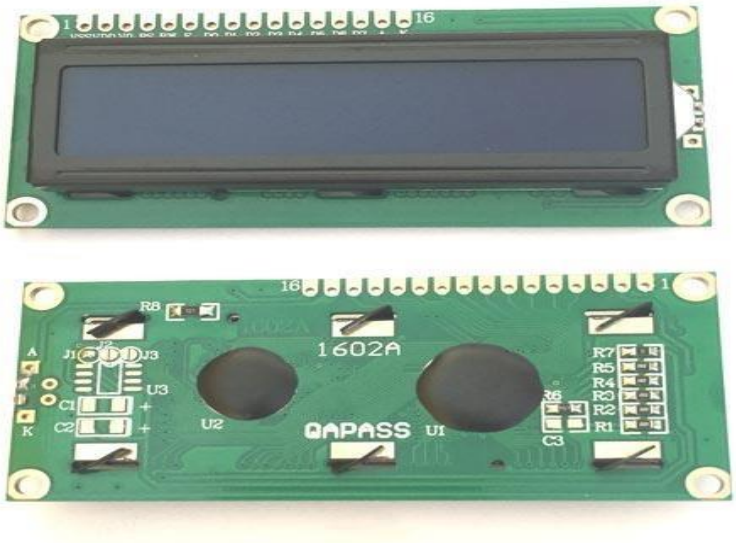
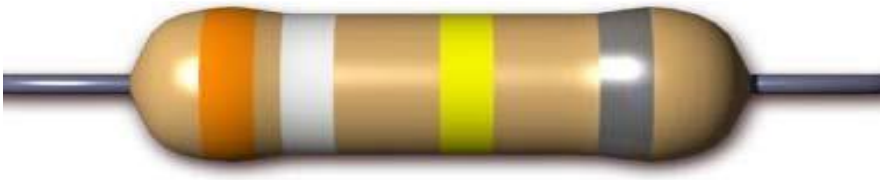


Figure: 16*2 LCD panel

PUSH BUTTON: A **push-button** or simply **button** is a simple switch mechanism to control some aspect of a machine or a process. Buttons are typically made out of hard material, usually plastic or metal. The surface is usually flat or shaped to accommodate the human finger or hand, so as to be easily depressed or pushed. Buttons are most often biased switches, although many un-biased buttons (due to their physical nature) still require a spring to return to their unpushed state.

Figure: Push Button

Resistor:



Resistance is the opposition of a material to the current. It is measured in Ohms Ω. All conductors represent a certain amount of resistance, since no conductor is 100% efficient. To control the electron flow (current) in a predictable manner, we use resistors. Electronic circuits use calibrated lumped resistance to control the flow of current. Broadly speaking, resistor can be divided into two groups viz. fixed & adjustable (variable) resistors. In fixed resistors, the value is fixed & cannot be varied. In variable resistors, the resistance value can be varied by an adjuster knob. It can be divided into (a) Carbon composition (b) Wire wound (c) Special type. The most common type of resistors used in our projects is carbon type. The resistance value is normally indicated by color bands. Each resistance has four colors, one of the band on either side will be gold or silver, this is called fourth band and indicates the tolerance, others three band will give the value of resistance (see table). For example if a resistor has the following marking on it say red, violet, gold. Comparing these colored rings with the color code, its value is 27000 ohms or 27 kilo ohms and its tolerance is ±5%. Resistor comes in various sizes (Power rating).The bigger the size, the more power rating of 1/4 watts. The four color rings on its body tells us the value of resistor.value.

buzzer:

A buzzer or beeper is an audio signaling device, which may be mechanical, electromechanical, or piezoelectric. Typical uses of buzzers and beepers include alarm devices, timers and confirmation of user input such as a mouse click or keystroke. A piezoelectric element may be driven by an oscillating electronic circuit or other audio signal source, driven with a piezoelectric audio amplifier. Sounds commonly used to indicate that a button has been pressed are a click, a ring or a beep.

Figure: Buzzer

BREADBOARD: A breadboard is a rectangular plastic board with a bunch of tiny holes in it. These holes let you easily insert electronic components to **prototype** (meaning to build and test an early version of) an electronic circuit, like this one with a battery, switch, resistor, and an LED (light-emitting diode). Modern breadboards are made from plastic, and come in all shapes, sizes, and even different colors. While larger and smaller sizes are available, the most common sizes you will probably see are "full-size," "half-size," and "mini" breadboards. Most breadboards also come with tabs and notches on the sides that allow you to snap multiple boards together. However, a single half-sized breadboard is sufficient for many beginner-level projects.

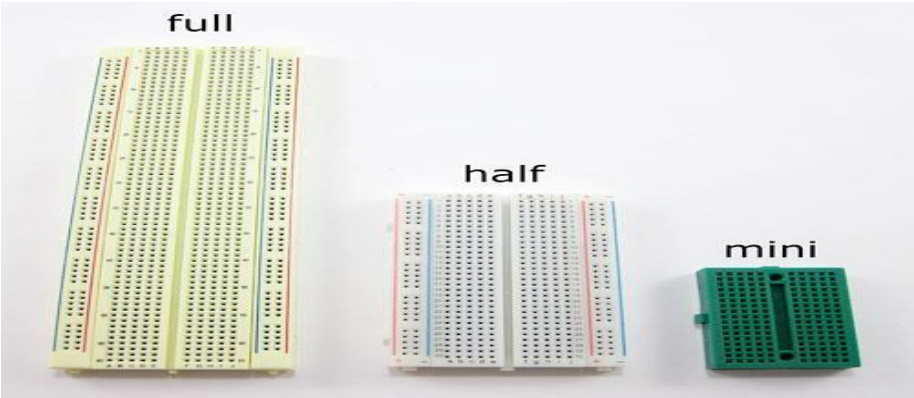


Figure: Breadboard

RTC MODULE: At the heart of the module is a low-cost, extremely accurate RTC chip from Maxim – DS3231. It manages all timekeeping functions and features a simple two-wire I2C interface which can be easily interfaced with any microcontroller of your choice. The chip maintains seconds, minutes, hours, day, date, month, and year information. The date at the end of the month is automatically adjusted for months with fewer than 31 days, including corrections for leap year (valid up to 2100).

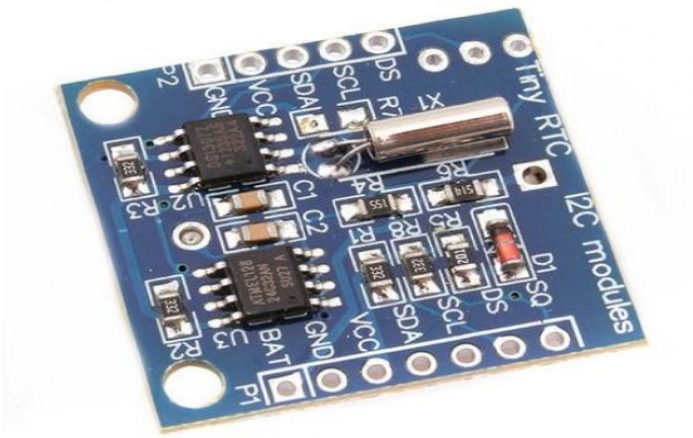


Figure: RTC Chip

The clock operates in either the 24-hour or 12-hour format with an AM/PM indicator. It also provides two programmable time-of-day alarms.

Literature Survey

M A Muchtar, Seniman, D Arisandi, S Hasanah “Attendance fingerprint identification system using arduino and single board computer” 2 nd International conference on computing and applied informatics 2017 2nd International Conference on Computing and Applied Informatics 2017 IOP Publishing IOP Conf. Series: Journal of Physics: Conf. Series 978 (2018) 012060 doi :10.1088/1742- 6596/978/1/012060 2nd International Conference on Computing and Applied Informatics 2017 IOP Publishing IOP Conf. Series: Journal of Physics: Conf. Series 978 (2018) 012060 doi :10.1088/1742-6596/978/1/012060 Abstract. Fingerprint is one of the most unique parts of the human body that distinguishes one person from others and is easily accessed. This uniqueness is supported by technology that can automatically identify or recognize a person called fingerprint sensor. Yet, the existing Fingerprint Sensor can only do fingerprint identification on one machine. For the mentioned reason, we need a method to be able to recognize each user in a different fingerprint sensor. The purpose of this research is to build fingerprint sensor system for fingerprint data management to be centralized so identification can be done in each Fingerprint Sensor. The result of this research shows that by using Arduino and Raspberry Pi, data processing can be centralized so that fingerprint identification can be done in each fingerprint sensor with 98.5 % success rate of centralized server recording. Nur Izzati Zainal, Khairul Azami Sidek, Teddy Surya Gunawan, Hasmah Mansor, and Mira Kartiwi “Design and Development of portable classroom attendance system based on Arduino and fingerprint biometric" Department of Electrical and computer engineering,kulliyah of engineering, department of information systems , kulliyyah of information and communications technology, international Islamic university Malaysia, Kuala Lampur, Malaysia 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 17 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. Khin San Myint, Chan Mya Mya Nyein “Fingerprint Based Attendance System Using Arduino” Department of Electronic Engineering, Technological University Design and Development of Portable Classroom Attendance System Based on Arduino and Fingerprint Biometric Design and Development of Portable Classroom Attendance System Based on Arduino and Fingerprint Biometric Design and Development of Portable Classroom Attendance System Based on Arduino and Fingerprint Biometri Abstract- Attendance system is required in many different places such as offices, companies, schools, organizations and institutions, etc. There are many attendance systems to take attendance. But, every place need to have a good system. This paper describes one of the attendance systems. The main objective of this paper is to study and construct the attendance system using fingerprint module. In this system, Arduino UNO controller and PLX DAQ tool are the main components to display the record on Excel. Karthik Vignesh E, Shanmuganathan S , A.Sumithra S.Kishore and P. Karthikeyan “A Foolproof Biometric Attendance Management System” Information Technology, Velammal College of Engineering and Technology ,Velammal College of Engineering and Technology, Madurai, Tamil Nadu, India. In this paper, we proposed a system which maintains the attendance records of students automatically. Manual entering of attendance in log books becomes a difficult task and it also wastes the time. Reading out the names of each student, each hour destroys the precious time. So we designed an efficient module that comprises of a fingerprint sensor to manage the attendance records of students. Our module enrolls the student’s as well as staff’s fingerprints. This 18 enrolling is a onetime process and their fingerprints will be stored in the fingerprint sensor. During enrolling of fingerprints alone we require a system since it is a onetime

process. You can have your own roll number as your fingerprint id which will be unique for each student and staff. After enrolling process gets completed you can disconnect the module from the system and insert a 9v battery into the module. This will provide power when the module is not connected with the system. Then the module can be taken to the class and the presence of students can be get. The presence of each students will be updated in a database and the data will be passed to the server using Wi-Fi. If a student is absent for a particular class automatically a SMS will be sent to their parents. If a student is absent continuously for more than three days a message intimating the parents to meet the HOD will be sent automatically. So everything here gets automated. Also a unique username and password for staff members are given in a website we create and the website can display the student's details, their attendance percentage which makes the work simple. Also mails and messages can be sent by the staff members using that site to intimate any urgent messages to the parents. Lia Kamelia, Eki Ahmad Dzaki Hamidi, Wahvudin Darmalaksana, Afit Nugraha, "Real-Time Online Attendance System Based on Fingerprint and GPS in the Smartphone" UIN Sunan Gunung Djati, Department of Electrical Engineering, Bandung, Indonesia Real-time online attendance method is helpful for workers who do a lot of activities outside the office or workers with multi-schedule. The attendance system using online biometric fingerprint system will reduce the problems caused by manual system usage such as lags in data management. The purpose of the research is to constructs an online presence system that combines fingerprint modules and GPS. The ZFM-20 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. GPS module is applied to determine the user's location and sends it to the smartphone. Arduino module in the system will send a text message to the parties concerned about the user's location data automatically. Each module works well and testing the entire system showed the system work reliable according to the initial scenario. The User can access the report using SMS, website, and application on the Android smartphone. The fingerprint sensor can determine the fingerprint stored in the database with an average response time of 1.39 seconds, and GPS can

determine latitude and longitude with an average error of 0.007352% and 0.0003% respectively. 19 A. Jain, L. Hong , S. Pankanti, and R. Bolle , “An Identity Authentication System Using Fingerprints”, Proceedings of the IEEE, Vol. 85, Issue 9, 1997, pp. 1365-1388. Abstract- 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. Piyush Devikar, Ajit Krishnamoorthy, Aditya Bhanage, Mohit Singh Chauhan Department of Electronics and Telecommunication Engineering, Vivekanand Education Society’s Institute of Technology, Mumbai University, Mumbai, India. Abstract: 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 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. Dipak Gadekar(1), Sanyukta Ghorpade(2), Vishakha Shelar(3), Ajay Paithane(4).”(1,2,3) Students, (4) Professor, Department of Electronics And Telecommunication Engineering, JSPMs Rajarshi Shahu College of Engineering,

Tathwade, Pune, Maharashtra, India. 20 Abstract - Authentication is one of the vital concern in this era of information system. . Inclusive of the other techniques, Human Face Recognition (HFR)is one ofthe techniques which is used for user authentication . HFR has been extensively used in many appliances asin , video conferencing,military services and attendance systems . Maintaining attendance is difficult process if it is done manually. The automated attendance system for administrating the attendance can be put into effect using the several ways of biometrics. Usage of this system can resolve the issue of fake attendance and proxies. Instead of recording the attendance in writing, taking attendance through fingerprint and face recognition will make it a hassle free process. Niharika Yadu(1) , K Uma(2) Research Scholar, Dept. of Electronics Engineering, BIT, Durg, C.G., India(1) Associate Professor, Dept. of Electronics Engineering, BIT, Durg, C.G., India(2) “A Review on Real Time IOT Based advanced E-attendance System.” Abstract: If we talk about the current scenario of our education system then we found that we have a lot of technologies to use but still we are following the traditional system. 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 and RFID 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. ”Biometric Student Attendance System using IoT”, Sameer Kanse , Monish Shaikh, Siddesh Gadhari , Pravin Labde , Prof. Anuprita Gawande. Department of Electronics and Telecommunication Engineering, Shivajirao S. Jondhle College of Engineering & Technology, Asangaon, Mumbai University, India. ABSTRACT.In the World of Technology, Biometrics plays an effective role in identifying Human beings. Through this paper, we will develop a unique system that can identify students for attendance purpose using their fingerprints. We will need an Arduino Uno board for interfacing microcontroller with the Finger Print Scanner R305. So, with the help of Finger Print Scanner R305, we will store the finger prints of all the students and

once they are stored, the Finger Print Scanner will compare the present finger print on the scanner and previously stored 21 finger prints. If any finger print is matched, the microcontroller will print the concern data stored for the particular finger print on the LCD Display. In addition to this, we can add Wi-Fi module, to upload the data into remote IP address, to access it from anywhere in the world

Working of Project

Implementation

1.The Finger Print Sensor is interfaced with the Arduino board.

2. At the beginning, your finger will be scanned by placing your finger on the scanner

3. Once your finger is scanned, the scanner will generate template by Image Processing method which will be stored for comparing

4. Like this we will store all the templates of different people

5. So when we place our finger, the scanner will scan the finger and it will generate template and this template will be compared with previously stored templates

6. If both templates are matched, then certain people data stored will be shown on the LCD

The data regarding how many students were present on day to day basis can be updated in remote cloud rather maintain ledgers and record books and can be retrieved whenever we want. The working of the Fingerprint Sensor Based Biometric Attendance System. In this project, we have used a RTC Module for time & date display. We used 1 LED for power indication, 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.

Enrolling New Fingerprint

To enroll New Fingerprint Click on the Enroll button. Then select the memory location where you want to store your fingerprint using the UP/DOWN button. Then click on OK. Put your finger and remove your finger as the LCD instructs. Put your finger again. So finally your fingerprint gets stored.

Deleting Stored Fingerprint

To delete the fingerprint which is already clicked on DEL Button. Then select the memory location where your fingerprint was stored earlier using the UP/DOWN button. Then click on OK. So finally your fingerprint is deleted.

Downloading Data:

Simply click on Register/Back Button and reset the button together. At this movement, the serial

monitor should be opened.

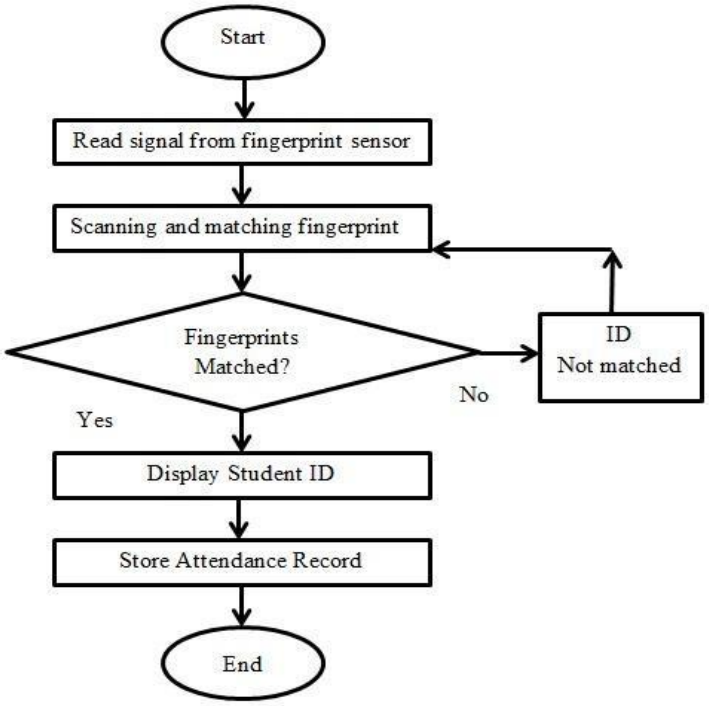


Figure: Working of Hardware

SOFTWARE CODING:

```
#include<EEPROM.h>
#include<LiquidCrystal.h>
LiquidCrystal lcd(13,12,11,10,9,8);
#include <SoftwareSerial.h>
SoftwareSerial fingerPrint(2, 3);

#include <Wire.h>
#include "RTClib.h"
RTC_DS1307 rtc;

#include "Adafruit_Fingerprint.h"
uint8_t id;
Adafruit_Fingerprint finger = Adafruit_Fingerprint(&fingerPrint);

#define enroll 14
#define del 15
#define up 16
#define down 17
#define match 5
#define indFinger 7
#define buzzer 5

#define records 4  // 5 for 5 user

int user1,user2,user3,user4,user5;

DateTime now;
```

```
void setup()
{
   delay(1000);
   lcd.begin(16,2);
   Serial.begin(9600);
   pinMode(enroll, INPUT_PULLUP);
   pinMode(up, INPUT_PULLUP);
   pinMode(down, INPUT_PULLUP);
   pinMode(del, INPUT_PULLUP);
   pinMode(match, INPUT_PULLUP);
   pinMode(buzzer, OUTPUT);
   pinMode(indFinger, OUTPUT);
   digitalWrite(buzzer, LOW);
   if(digitalRead(enroll) == 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;   // 5th user
//  download(eepIndex);
 Serial.println();
}
```

```
}
if(digitalRead(del) == 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("   Attendance   ");
lcd.setCursor(0,1);
lcd.print("    System    ");
delay(2000);
lcd.clear();
lcd.print("Circuit Digest");
lcd.setCursor(0,1);
lcd.print("Saddam Khan");
delay(2000);
     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(1000);
  if (finger.verifyPassword())
  {
    Serial.println("Found fingerprint sensor!");
    lcd.clear();
    lcd.print("Found Module ");
    delay(1000);
  }
  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.isrunning())
   {
   Serial.println("RTC is NOT running!");
   // following line sets the RTC to the date & time this sketch was compiled
      rtc.adjust(DateTime(F(_DATE), F(TIME_)));
   // This line sets the RTC with an explicit date & time, for example to set
   // January 21, 2014 at 3am you would call:
   // rtc.adjust(DateTime(2014, 1, 21, 3, 0, 0));
   }
lcd.setCursor(0,0);
 lcd.print("Press Match to ");
 lcd.setCursor(0,1);
 lcd.print("Start System");
 delay(2000);

 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("Registed");
            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)   // 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(enroll) == 0)
  {
   lcd.clear();
   lcd.print("Please Wait");
   delay(1000);
   while(digitalRead(enroll) == 0);
   Enroll();
  }

  else if(digitalRead(del) == 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(up) == 0)
    {
      count++;
      if(count>records)
      count=1;
      delay(500);
    }

    else if(digitalRead(down) == 0)
    {
      count--;
      if(count<1)
      count=records;
      delay(500);
    }
    else if(digitalRead(del) == 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(enroll) == 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(up) == 0)
```

```
{
  count++;
  if(count>records)
  count=1;
  delay(500);
}

else if(digitalRead(down) == 0)
{
  count--;
  if(count<1)
  count=records;
  delay(500);
}
else if(digitalRead(del) == 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(enroll) == 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");
    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("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("Figer 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("        ");
}
```

CONCLUSION

Here we have developed a Biometric fingerprint based attendance system using Arduino.
In this project we have used R307 fingerprint sensor which reads the Fingerprint and stores in the form of digital data. A buzzer is activated and LED blinks then LCD panel shows that data is stored along with username, date and time. Working of this fingerprint attendance system project is fairly simple. First of all, the user needs to enroll 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. 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. The system can be improved by encasing it in a plastic covering. This would make it more compact and easy to use in a classroom setting. The system can be configured to enable lecture-wise attendance taking. It can further be improved to automatically calculate attendance percentages of students and intimate the teachers if a student's attendance is below a certain percentage. It can also be modified to fit the corporate environment.

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.

RESULTS

The experimental model was made following the circuit diagram and the desired results were obtained. Every time someone places his finger on the sensor the sensor reads the data and stores it in the cloud. Next time someone wants to check the fingerprint he/she places the finger on the sensor. The sensor reads the data and searches and cross-checks the data with stored fingerprints. If it matches with any of them then it displays the username, date and time. If not then says fingerprint doesn't match . That's how the whole system works.

FUTURE WORK

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.

References

https://ijarcce.com/wp-content/uploads/2016/11/IJARCCE-ICRITCSA-3.pdf

www.irjet.net/uploads/2016/05/ IRJET-V3I5237.pdf

https://www.skyfilabs.com/project-ideas/biometric-attendance-system-with-iot

https://www.how2electronics.com/fingerprint-sensor-based-biometric-attendance-system/

https://circuitdigest.com/microcontroller-projects/fingerprint-attendance-system-using-arduino-uno

Wang and Jingli , “The Design of Teaching Management System in Universities Based on Biometrics Identification and the Internet of Things Technology”, IEEE 10th International Conference on Computer Science & Education (ICCSE), Cambridge University, UK July 22-24, 2015, pp. 979-982.