DEVELOPMENT OF FINGERPRINT-BASED ATTENDANCE MANAGEMENT SYSTEM FOR SMART CLASS APPLICATIONS

www.akwapolyjournal.org
P-ISSN: 2536-6733 | E-ISSN: 2616-0773

Supported by:
tetfund
TERTIARY EDUCATION TRUST FUND
TETFund/DESS / POLY / IKOT-OSURUA/ ARJ /1-2)

Essien, Anietie Etim[1]
Department of Electrical and Electronic Engineering,
Akwa Ibom State University Mkpat Enin, Akwa Ibom State

Enyenihi Henry Johnson[2]
Department of Electrical and Electronic Engineering,
Akwa Ibom State University Mkpat Enin, Akwa Ibom State
enyenihijohnson@aksu.edu.ng

Imo E. Nkan[3]
. Department of Electrical and Electronic Engineering,
Akwa Ibom State University Mkpat Enin, Akwa Ibom State

Abstract
This paper describes the development of an innovative attendance management system called Fingerprint-Based Attendance Management (FPAM), which aims to simplify the attendance management process for educational institutions. The system utilizes cutting-edge fingerprint recognition technology, including a portable hardware device with a fingerprint reader, Arduino microcontroller, NodeMCU module, organic light-emitting diode (OLED) module, and keypad. It also consists of a web application and mobile app that provide easy access to attendance data captured by the device, making it convenient for both students and faculty members. The system's codes are written using the Arduino IDE and PHP, ensuring they are efficient and reliable. The system has been subjected to rigorous testing, and the results have been overwhelmingly positive. It has effectively managed attendance for different courses and student populations, making it an ideal solution for educational institutions looking to streamline their attendance management processes. The server is hosted on the "litehost" platform with a domain name of anietieattendancessystem.com.ng, ensuring the system is always accessible and reliable. Overall, the FPAM system is a valuable addition to the educational industry, providing a seamless and efficient attendance management solution.

Keyword: *Fingerprint, Arduino UNO Board, OLED Display, Fingerprint Sensor, , Smart Attendance Management System*

Introduction

Nowadays, the class attendance record is one of the valuable tools used for assessing students' commitment to their course work and to evaluate the factors that influence students' performance in their course [1,2,3,4,5,6,7,8]. The approach to taking class attendance has evolved over the years, from manual counting to hardcopy book recording [9,10,11,12,13,14] and eventually to different forms of electronic data recording and storage, as well as various forms of integrated electronic data capture recording and storage options [15,16, 17,18, 19,20,21,22,23,24].

The RFID-based approach was popularly used in the early days of electronic class attendance [25,26,27,28,29,30]. In the RFID approach, the students are expected to carry their identification card with an embedded RFID tag, which the RFID reader can automatically read as the student enters the classroom [28,29,30,31,32,33, 34,35,36,37,38,39]. However, due to the problem of loss of such cards, impersonation and other fraudulent practices that limited the effectiveness of such method, the fingerprint-based attendance system was advocated [40,41,42,43,44,45].

Accordingly, this paper presents the development of a finger print-based attendance management system for intelligent class application [46,47,48,49,50,51,52]. The system relies on a high-precision fingerprint sensor module with an embedded high-powered digital signal processor (DSP) chip for processing the fingerprint images, after which the photos can be used in the fingerprint enrolment and fingerprint matching process. The fingerprint-based attendance management (FPAM) system presented in this paper also consists of a portable hardware device, web application and mobile app. The portable hardware device consists of R305 fingerprint sensor and scanner [53,54,55,56], the Arduino microcontroller [57,58,59,60,61,62,63], the ESP8266 NodeMCU module [64,65,66,67,68,69], the organic light-emitting diode (OLED) module [70,71,72,73,74] and the 4X4 matrix membrane keypad [75,76,76,78,79,80,81,82,83]. The portable hardware device is used to capture the fingerprint images. Then, it uploads them to the online database from when the web application and Android app are used to access the data and extends their use for diverse applications. This paper builds and tests a simple prototype of the attendance system with some courses and students in a class. In all, the ideas and results presented in this paper showed that the fingerprint-based attendance system can be effectively used to conduct attendance management for different courses and student populations.

Methodology

Description of the fingerprint-based attendance management system and its component parts

In this paper, the main objective is development of fingerprint-based attendance management (FPAM) system for smart class application. The system has a fingerprint reader which it uses to take class attendance of students for the various courses they registered in each semester. In addition, it uses web application to further process the attendance information and then generate requisite attendance reports for each student and course.

The FPAM system (as shown in Figure 1) consistsof portable hardware device, web application and mobile app. The core functions of the portable hardware device are one, to enroll the users, namely, the students and lecturers (faculties) into the system by capturing and storing their fingerprint templates and two, to capture and verify users fingerprint for attendance record update. The portable hardware device consist of R305 fingerprint reader, Arduino microcontroller, ESP8266 NodeMCU module, organic light-emitting diode (OLED) module and keypad.

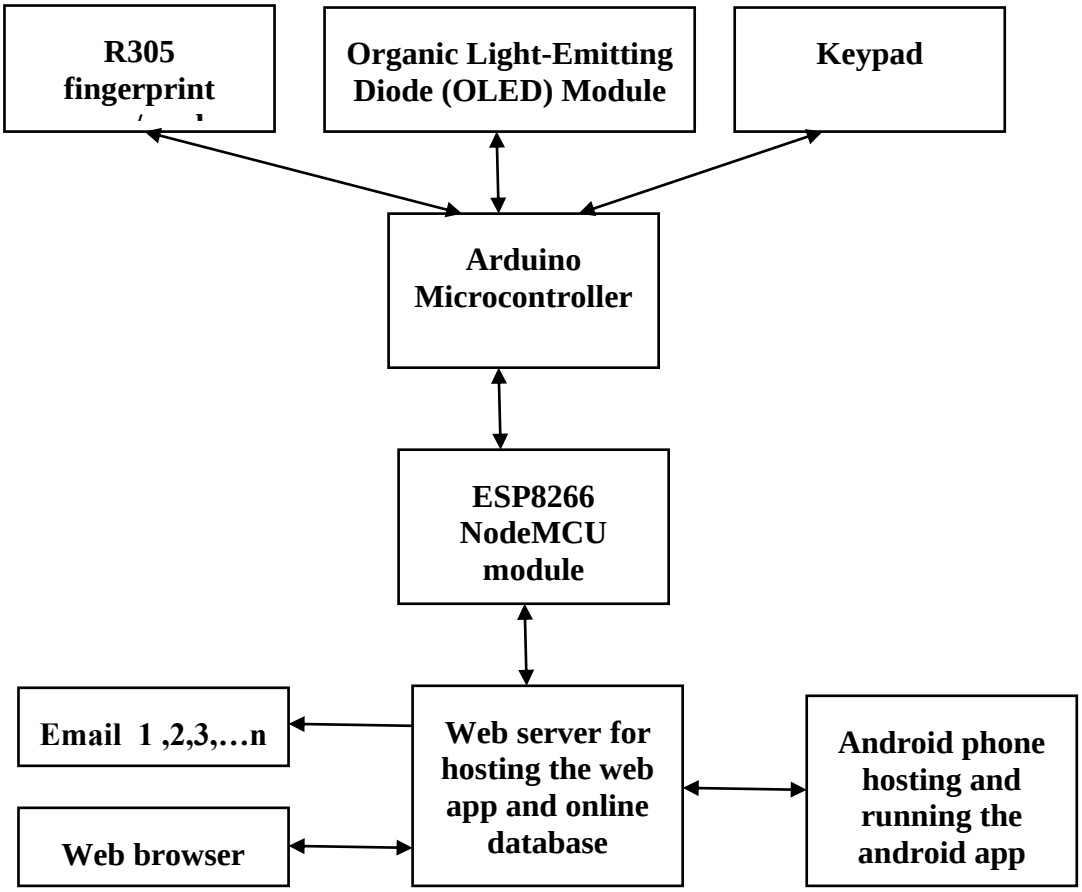


Figure 1 The schematic diagram of the fingerprint-based attendance management (FPAM) system

Apart from the portable hardware device, the FPAM system also has web application that requires web server for hosting both the web application and the online database. Also, the FPAM system has mobile app that is hosted and run on android phones. The web app enables expanded user information to be captured and integrated with the limited user information that is captured with the portable hardware device. The web app interfaces with the online database and permits interface with the mobile app and third party email applications. Consequently, the web app extends the applicability of the attendance data captured with the portable hardware device by providing application program interfaces that can be exploited for smart class applications. Specifically, in this paper, the web app enables automatic emailing of attendance information to user's email box and also permits remote viewing of the attendance information via the mobile app that is hosted and run on android phones.

The ESP8266 NODEMCU

ESP8266 NODEMCU is a popular open-source development board with Lua language-based firmware implemented by the ESP8266 Wi-Fi System on Chip (SoC), which is a product developed by Espress if Systems. The ESP8266 WiFi module acts as the modem that enable the Arduino microcontroller to effectively connect to the Internet. Basically, the ESP8266 WiFi module is equipped with complete functional WiFi Stack and the requisite TCP/IP Stack that can enable various microcontrollers to connect to WiFi networks.

By default, the ESP8266 NODEMCU module in the ESP8266 NODEMCU Development Board comes preloaded with the NodeMCU proprietary firmware that operates on the basis of standard AT commands. Fortunately, the NodeMCU Development Board is developed with Arduino-like input/output hardware. As such, the NodeMCU Development Board can easily be programmed using the Arduino Integrated Development Environment (Arduino IDE). In that case, if the NodeMCU module is interfaced with Arduino and Arduino software is downloaded using the Arduino IDE, the proprietary firmware Lua language-based firmware will be erased and the module can then be programmed and operated using the conventional Ardiuno Arduino software. Also, by the original design, the ESP8266 NODEMCU module can be programmed using Lua interpreted language. However, with the Arduino IDE, the ESP8266 NODEMCU module can be programmed using C language.

In this work, the ESP8266 NODEMCU Development Board is used and the original proprietary Lua language-based firmware is flushed out by using the Arduino IDE to download and installing the Arduino software. Afterwards, the Arduino-based programming option is used to operate and control the NodeMCU ESP8266-12 module and the other system components interfaced to it. Importantly, the ESP8266 NODEMCU Development Board also provides some of the most keys features of Arduino microcontrollers such as GPIO, PWM, and ADC. As such, apart from providing the internet connectivity for the system using its ESP8266 WiFi module, the ESP8266 NODEMCU Development Board also renders the Arduino microcontroller functionalities for the system.

The OLED display and its interface with the microcontroller

The SSD1306 model OLED is used and it is a 0.96-inch monochrome display OLED with 128×64 pixels resolution. The OLED display was selected for the work because, when compared with other similar display technologies, it is a low power display with good contrast in dark environments. The SSD1306 model OLED has four pins for connection to the Arduino microcontroller, they

are: Vin, GND, SCL and SDA pins (as shown in Figure 2). The pin description and connection between the SSD1306 model OLED display and the Arduino microcontroller is shown in Table 1.

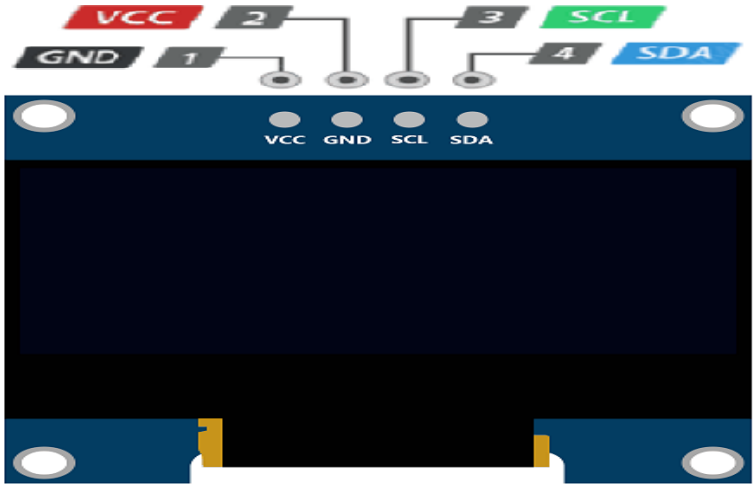


Figure 2 The four pins used by the SSD1306 model OLED to connect to the Arduino microcontroller (Source:

https://randomnerdtutorials.com/guide-for-oled-display-with-arduino/
After the connection of the OLED display board to the Arduinomicrocontroller, the adafruit_SSD1306.h library, as well as the adafruit_GFX.h library are install in the Arduino IDE using the Arduino IDE. The two libraries enable the OLED display to be controlled by the Arduino microcontroller. After the installation of the adafruit_SSD1306.h and adafruit_GFX.h libraries, the Arduino IDE is restarted and the effectiveness of the connection and interfacing is tested.

Table 1 The pin connection between the SSD1306 model OLED display and the Arduino microcontroller

S/N	SSD1306 model OLED display Pin name and description	Arduino Uno microcontroller pin
1	GND (ground pi)	GND
2	Vin (power pin which can take 3.3V or 5V)	5V
3	SCL (serial clock pin)	A5

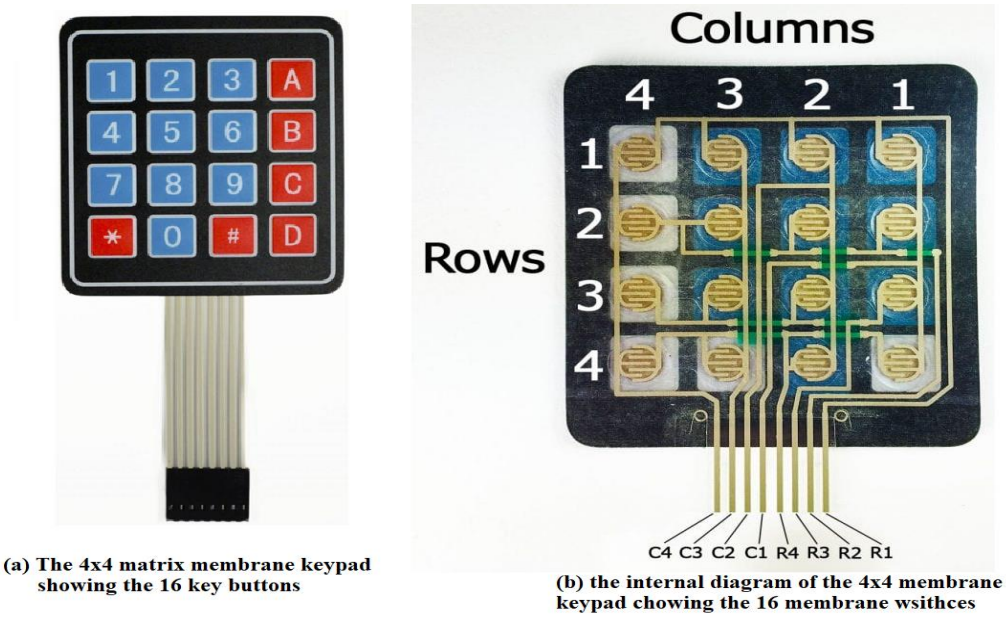
4	SDA (serial data pin)	A4

2.1.3 The keypad module and its interface with the microcontroller

In this paper, a 4X4 matrix membrane keypad is used. The images of the external and internal diagram of the 4x4 Membrane keypad are presented in Figure 3. The 4X4 matrix membrane keypad has 16 keys that are arranged in rows and columns, where each key has a button with a membrane switch beneath it. The membrane switches in each of the four rows and each of the four columns are interconnected and each row and column gives rise to a single pin on the keypad module, for a total of 8 pins as shown in Figure 3. The Arduino microcontroller detects a particular pin pressed by detecting the specific row and column pin that is connected to the button.

The 4×4 keypad module has 8 pins (as shown in Figure 3) which are connected to the Arduino microcontroller via the digital I/O pin 2 to pin 9 as shown the below Table 2 and Figure 4. After the connection of the 4X4 matrix membrane keypad module to the Arduino microcontroller, the keypad library (specifically, keypad library for Arduino authored by Mark Stanley and Alexander Brevig) is install in the Arduino IDE using the Arduino IDE. The keypad library enables

the 4X4 matrix membrane keypad to be controlled by the Arduino microcontroller. After the installation of the keypad library, the Arduino IDE is restarted and the effectiveness of the connection and interfacing is tested.



(a) The 4x4 matrix membrane keypad showing the 16 key buttons

(b) the internal diagram of the 4x4 membrane keypad chowing the 16 membrane wsithces

Figure 3 The images of the external and internal diagram of the 4x4 Membrane keypad (Source: https://www.circuitbasics.com/how-to-set-up-a-keypad-on-an-arduino/).

Table 2 The pin names, pin descriptions and connections between the 4x4 membrane keypad and the Arduino microcontroller

The 4×4 Keypad module pin name	The 4×4 Keypad module pin description	The Arduino microcontroller (digital I/O) Pin
R1	Pin from the 1st row of the membrane switches	2
R2	Pin from the 2nd row of the membrane switches	3
R3	Pin from the 3rd row of the membrane switches	4
R4	Pin from the 4th row of the membrane switches	5
C1	Pin from the 1st column of the membrane switches	6

C2	Pin from the 2nd column of the membrane switches	7
C3	Pin from the 3rd column of the membrane switches	8
C4	Pin from the 4th column of the membrane switches	9

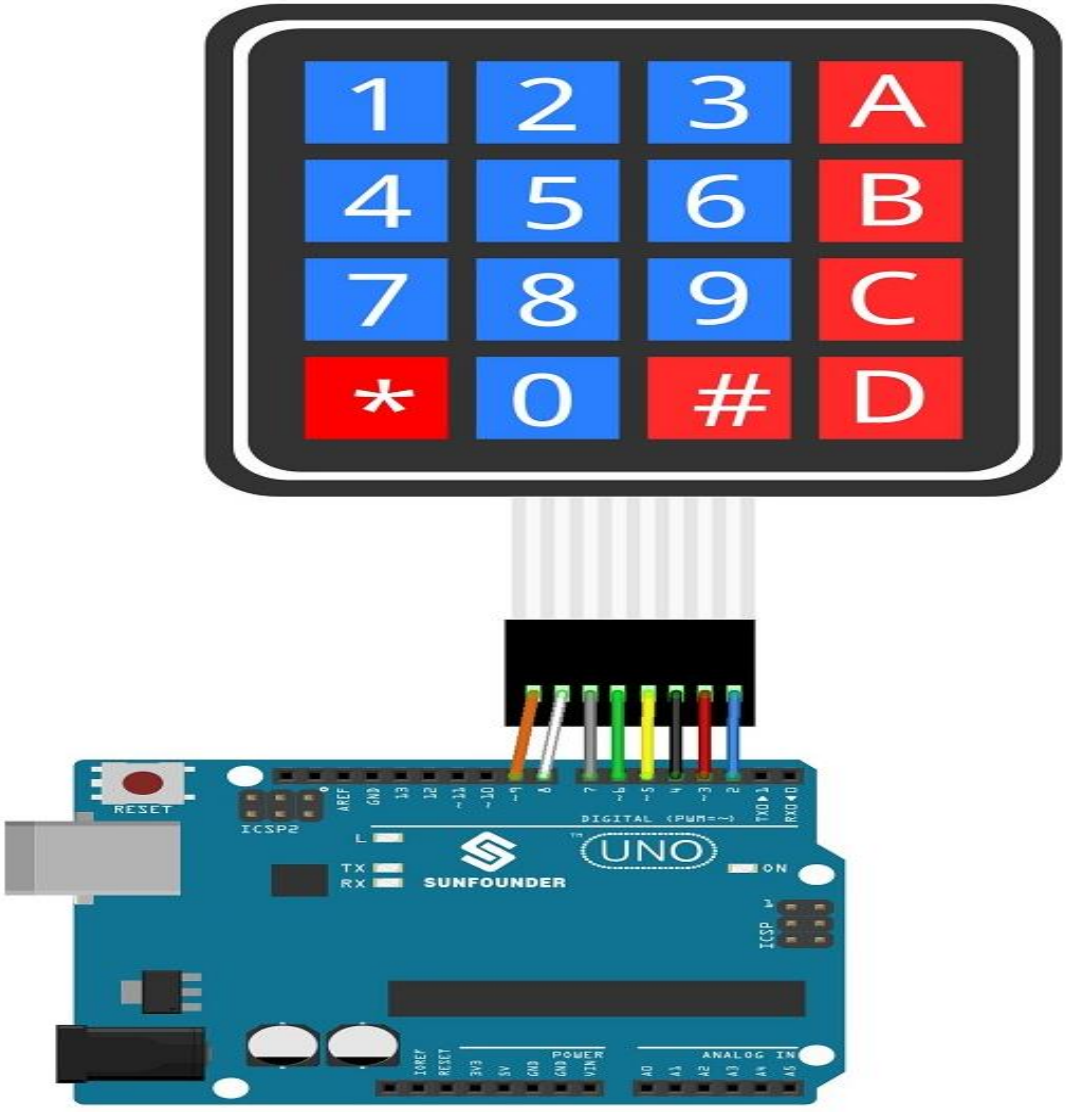


Figure 4 The image showing the pin connections between the 4x4 Membrane keypad and the Arduino microcontroller (source: http://wiki.sunfounder.cc/index.php?title=4x4_Membrane_Switch_Keypad)

R305 Optical Fingerprint sensor and scanner and its interface with the microcontroller

In this work, R305 optical fingerprint module is used. The R305 fingerprint sensor and scanner module has optical fingerprint sensor, high-performance fingerprint matching algorithm and large-capacity on-chip flash memory that can store about 980 fingerprint images. It can connect to computer and microcontrollers via serial communication method and it can also connect to computers using USB cable. The R305 fingerprint module pins and their corresponding a Arduino microcontroller interfacing pins are shown in Table 3.

The R305 fingerprint sensor and scanner can perform fingerprint enrolment and fingerprint matching operations. During a fingerprint scan, the R305 fingerprint sensor and scanner uses an array of LEDs to provide light for the camera in its optical sensing area to take pictures of the fingerprint placed on its sensor. The pattern of the fingerprint image captured is further processed by the high powered digital signal processor (DSP) chip in the R305 fingerprint sensor and scanner, after which the images can be used in the fingerprint enrolment or fingerprint matching process.

During fingerprint enrolment, the fingerprint image is captured twice, processed and store as reference fingerprint template for use during fingerprint matching session. During fingerprint matching session, the current live scanned fingerprint is matched with the fingerprint templates that are stored earlier in the flash memory of the R305 fingerprint sensor and scanner. The fingerprint searching can be either 1 to 1 or 1 to N approach.

Table 3 The R305 Fingerprint Module pins and their corresponding a Arduino microcontroller interfacing pin

R305 Fingerprint Module Pin Name	**R305 Optical Fingerprint Module Pin Description**	**Arduino Microcontroller Pin Name**
VCC	Power supply pin (typically 5 v)	5V (It can also operate with 3.3V)
TX	Pin that is used for receiving data from the serial communication (TTL logical level)	RX (digital I/O pin 2 which is software serial)

RX	Pin that is used for sending data from the serial communication (TTL logical level)	TX (digital I/O pin 3 which is software serial)
GND	Signal ground pin	GND

Development of the fingerprint-based attendance management system

The development of the fingerprint-based attendance management system is divided into some sections, namely;

i. Development of the portable hardware device:
ii. Development of the Web App:
iii. Development of the Android mobile app
iv. The user fingerprint enrolment and fingerprint verification
v. The online database scripts

(i) The portable hardware device: The development of the FPAM system began with the portable hardware device. First, the portable hardware device is assembled by interfacing the R305 fingerprint reader, ESP8266 NodeMCU module, OLED module and keypad with the Arduino microcontroller which coordinates the operations of the various hardware components of the system. Along with the interconnection of the hardware components, requisite software and hardware configurations are also effected. The system uses the Wi-Fi–based network connectivity available in the ESP8266 Wi-Fi module to enable wireless internet connectivity of the portable hardware device to the web server (as shown in Figure 1).

(ii) The Web App: Secondly, after setting up the portable hardware, the web app is developed and hosted along with the online database system. The web app is developed using PHP scripting language and hosted on XAMP while the database is built using MySQL database management system. The web app has three scripts submodules; one module thatis used to post the attendance record to the online database table, a second module that is used to sendemail to users on their attendance record update and statusand a third module that is usedto register, login and verify user Login details.

(iii) The Android mobile app: Thirdly, an android mobile app is also developed and installed on an android phone for use in test-running the system. The Android application is designed for the student and faculty

(lecturer) to view their attendance status based on the information that are extracted from the online database. The mobile app has login module and also another module for viewing the attendance details of the student and faculty (lecturer).

(iv) User fingerprint enrolment and fingerprint verification: After the portable hardware, the web app, the online database and android mobile app are developed, interfaced and tested, the R305 fingerprint reader setup is then used for enrollment of users and subsequently, for verification of user's fingerprint for attendance record update. Particularly, the R305 fingerprint reader has sensor for reading user's fingerprint, flash memory for storing the fingerprint as reference fingerprint templates for fingerprint verification process, and it also has ability to delete stored fingerprint templates or images stored in its memory. Basically, in the FPAM system, the R305 fingerprint reader is used to read and store fingerprints from different users during fingerprint enrolment phase and to read and search for matching fingerprint during the fingerprint verification or identification phase. Each fingerprint captured and stored in the R305 fingerprint reader memory is assigned a unique identification number, denoted as finger_id. In addition, the finger_id is sent to the web app for storage on the online database. At this point, the web application component of the FPAM system associates additional information of the user to the finger_id, namely, user identification number, denoted as s_id, user name, user email (denoted as emailed) and user login details that include username and password.

Essentially, once a user, for instance, a student's fingerprint is successfully captured and stored in the R305 fingerprint reader memory (hence enrolled in the system) the finger_id is uploaded to the web server along with the additional listed user details associated with the user. During the fingerprint verification phase, a user fingerprint will be captured and compared with the different fingerprint templates that are stored in the database until either a match is found and the finger_id is read out or no match is found and an error message is indicated. When a match is found and the finger_id is acquired, the finger_id is used to identify the rest of the user's records in the database and also to mark attendance for that particular student. When no match is found, error message is displayed and recapturing of the fingerprint can be done or other measures can be taken to address the issue of non-matching fingerprint.

In any case, the enrolment phase and verification phase results are displayed on the OLED display. In addition, since the FPAM system uses the web app and online

database to accommodate and communicate the extended user information, the finger_id data generated during the enrolment phase and during the verification phase are also sent to the web app and online database where they are combined with other user information to generate the composite attendance report details. The schematic diagram for data transfer from the fingerprint enrolment and verification phases to the web app and to the OLED display is shown in Figure 5.

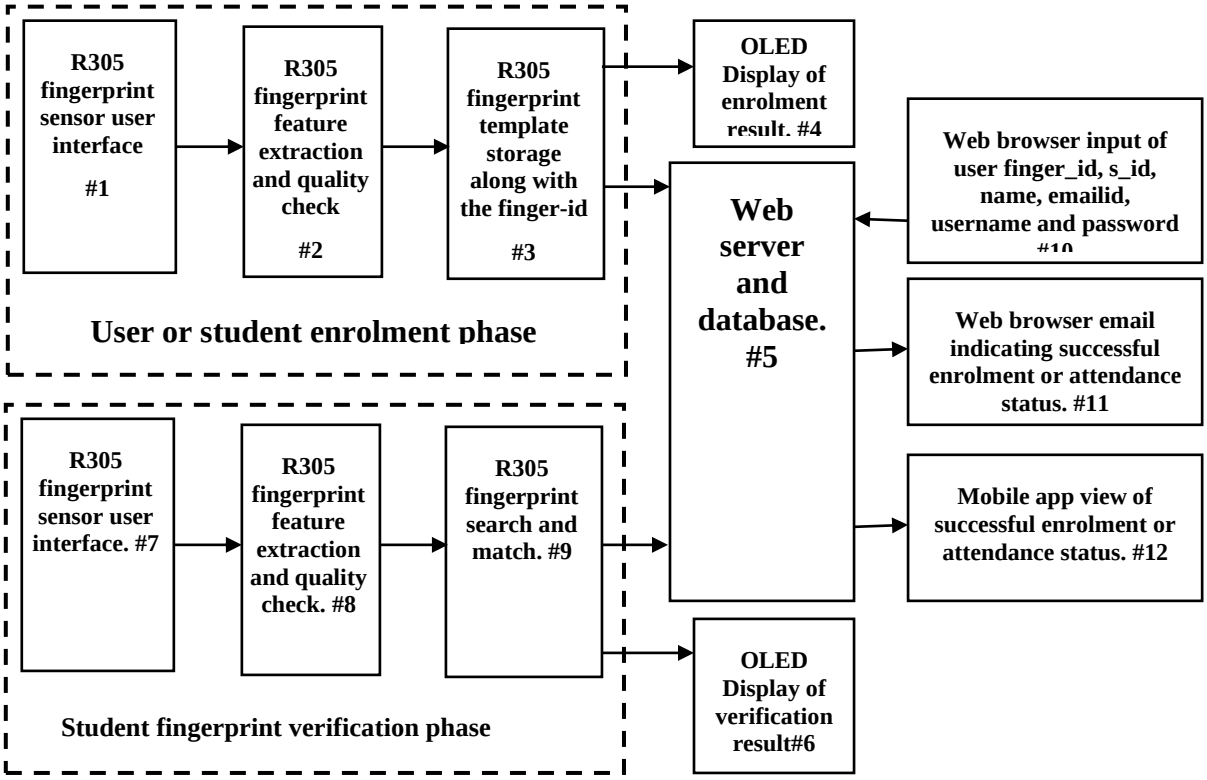


Figure 5 The schematic diagram of the extended finger enrolment and verification phases in the fingerprint-based attendance management (FPAM) system

(v) The online database scripts : The PHP scripts component of online database scripts enable both the hardware device and the mobile App to connect to the online database and hence access the data stored in the database. All the database scripts are hosted on the web server and they respond to request made by a user either by the user login request or when the user post attendance via the ESP8266NodeMCU module. There are at least six different scripts that perform different functions. So, as user makes a request, the specific script

that is designed for the specific user request is executed on the web server in response to the user request.

Notably, conn.php script is used to connect the Android application to the database. The alogin.php script used to verify the student login details (namely, the username and password). The flogin.php script is used to verify the faculty (lecturer) login details (namely, the username and password). The fpage.php script is used to generate the faculty page which shows the courses a particular lecturer teaches and the total attendance for his lectures for different dates. The spage.php script is used to generate the student data which include the student's registered courses, the number of lectures the student attended and the total number of lecture classes conducted by the lecturer. Finally, the postdemo.php script runs when the ESP8266NodeMCU module sends a request and its main purpose is to add a record to the database table where the record components are finger_id, course_id, and attendance received from arduino using the esp_wifi module.

The operating procedure of the fingerprint-based attendance management (FPAM) system

The fingerprint-based attendance management (FPAM) system presented in this paper is design to be used for taking attendance for multiple courses and it can also be used by different lecturers. Hence, it has provision for capturing and identifying different lecturers and course associated with each of the lecturers in the system. The FPAM system operating procedure is as follows:

Step 1: At the commencement of each semester, each student registers the courses he or she has for the semester. During the course registration process, the student is also required to enroll his/her fingerprint details which will be stored in the online database which is accessible to the web application of the FPAM system. Also, courses are allocated to lecturers and the information is uploaded to the same database. After the course registration by the students and course allocation to the lecturers, the database contains the course identification number, the list of students and their details for each course along with their fingerprint templates (images), the lecturers detail for each course and fingerprint.

Step 2: During a class for each course, the Lecturer comes to class with the portable fingerprint-based attendance hardware (PFAH) device. The lecturer select the particular course he wishes to take the attendance. Then, he scans the fingerprint of each student present with the PFAH device. For each student, the R305 fingerprint reader in the PFAH device will match the live scan student fingerprint with the fingerprint templates acquired from the database. If a match is found, the finger_id is displayed on the OLED display and the finger_id is

also sent to the web app which further marks the student as 'present' and also forwards email of the attendance report for that particular student to the student's email. If no matching fingerprint is found, then an error code is displayed.

Step 3: The FAH device has the list of all registered courses for that session. The course Lecturer is expected to press Key "A" on the keypad to select his own course ID. Once a course ID is selected, the attendance system is now ready for scanning the student's fingerprint to take the student's attendance for the course. Each student present in the class will place the registered finger (s) on the biometric scanner, the Arduino processes it and send it to the ESP8266 which in turn sends it to the online database and web app for comparison and recognition.

Step 4: Once the fingerprint match is found, the authentication is completed and the student gets a response displayed on the OLED screen as "Attendance marked successfully". The Lecturer can close attendance by pressing key "D' on the Keypad.

Step 5: The Attendance is now logged into the database for the student. Both Lecturer and Student can view this information on the Android mobile App when logged in as "FACULTY" or as a "STUDENT" respectively. Only the super Admin can access the PHP database. The student who are absent for that course will be alerted via email.

Results and discussion

Results on fingerprint enrolment hardware device and the attendance data capture and processing

The picture of the hardware setup for enrollment of the fingerprints of the students is shown in Figure 5. The setup required includes the R305 fingerprint sensor, the OLED display and the Arduino microcontroller, as shown in Figure 5

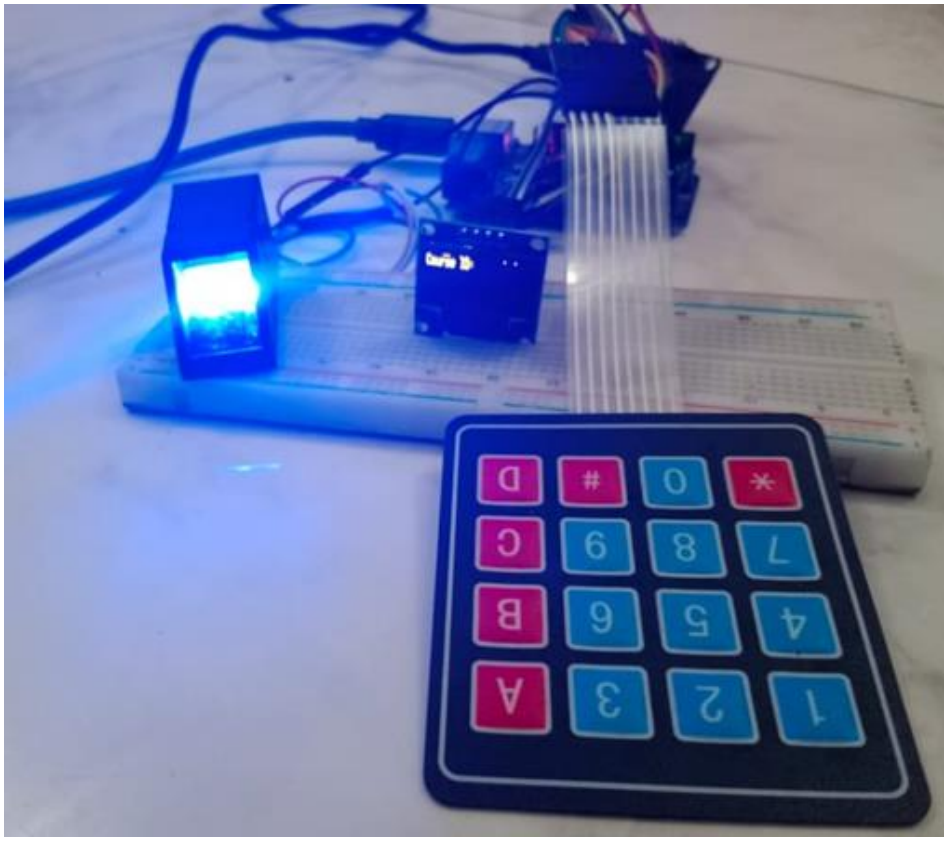


Figure 5 The hardware setup for fingerprint enrollment

Each of the students place their fingers on the R305 fingerprint sensor and the admin will enroll them by adding the finger_id, the corresponding student's ID and other details for each student enrolled in the attendance system. The process is usually done once, during the beginning of a new academic semester. At the end of the fingerprint enrollment, the system now have several fingerprint templates saved in the database, each with their corresponding finger_ids, student ID and the other details of the student that owns the fingerprint template.

In order to take attendance for a class, the course Lecturer must use the hardware device. The functioning of the attendance system starts when the person (Lecturer) taking the attendance presses the key 'A' on keypad (key 'A' as shown in Figure 5). The instruction that follows this is that the person (Lecturer) is prompted to enter the Course ID which attendance needs to be recorded. This prompt is displayed on the OLED display. Now, the person (Lecturer) enters the course id using the keypad, as shown in the picture in Figure 6 , where the course ID is 601.

Figure 6 The picture showing the keying in of the course ID using the keypad

After entering the course ID, the Lecturer is required to press the '#' key on keypad at which point the attendance taking process starts, as shown in Figure7, showing a prompt, 'Start Attendance' displayed on the OLED display. Also, at this point, the fingerprint scanner module is now activated and ready to start taking attendance of the students.

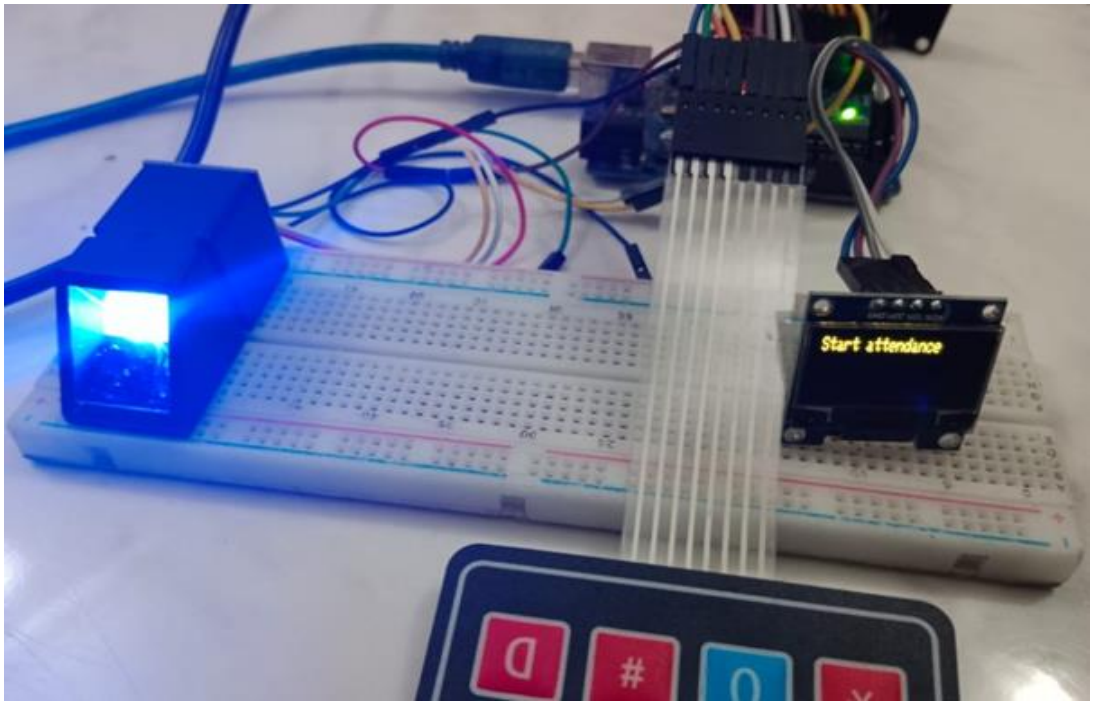


Figure 7 The picture showing the 'Start Attendance' prompt

The student's fingerprint is taken by scanning the finger using the fingerprint hardware device which has the R305 fingerprint sensor, as shown in Figure 8. If the student had been previously enrolled in the fingerprint module memory, a message is displayed on the OLED display saying 'finger_id marked present' (the finger id is the unique fingerprint_id stored in the fingerprint module memory).

When the fingerprint image is captured, the serial communication connection between the arduino and R305 fingerprint sensor is terminated and the serial communication between the ESP8266NodeMCU and Arduino is activated. The data record consisting of the scanned finger_id, the course_id which attendance is being taken and the attendance marked as "1" is sent to ESP8266NodeMCU module via the serial communication. The ESP8266NodeMCU module receives the data record in its buffer denoted as bfr and it then initiates an http request to the postdemo.php script that is hosted on the web server with URL "http://anietieattendancesystem.com.ng/postdemo.php". The bfr data is accepted using the postdemo.php script arguments. Eventually, the other php script is used to insert the bfr data along with the timestamp into the course_attendance table. At this juncture, the http link with the ESP8266NodeMCU module is closed using the **http.end()** and also the serial communication between the ESP8266NodeMCU and Arduino is ends.

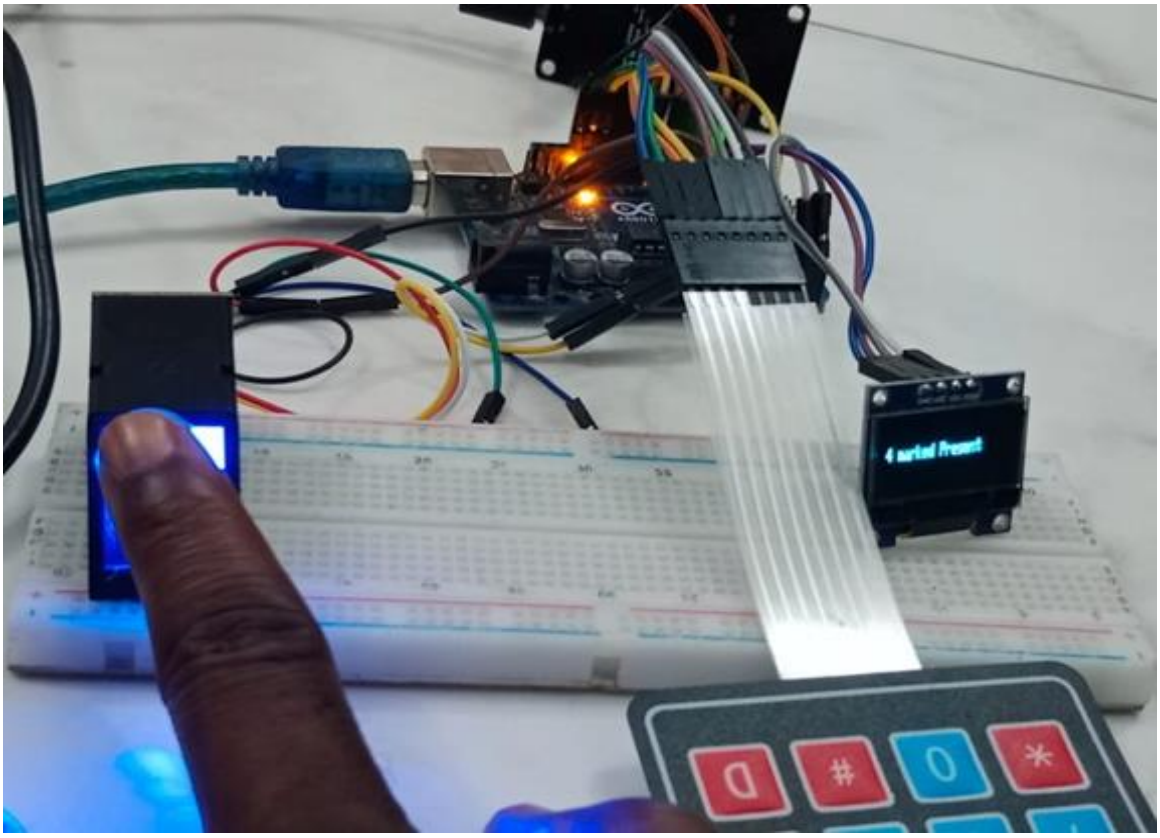


Figure 8 The picture showing the live scan of finger for taking student's attendance

When the serial communication between the ESP8266NodeMCU and Arduino ends, the serial communication between the arduino and the arduino and R305 fingerprint sensor is activated. The same procedure of taking attendance by scanning the student's finger using the R305 fingerprint sensor and then sending the captured attendance data to the online database continues till the person (lecturer) taking the attendance presses the key 'D' on the keypad which denotes the end of attendance taking session.

When the key 'D' on the keypad is pressed, a message will appear on the OLED display indicating, 'Attendance over'. Again, this will automatically close the serial communication connection between the arduino and R305 fingerprint sensor while the serial communication between the ESP8266NodeMCU and Arduino is activated.

The message string consisting of 'D' (denoting attendance is over) and course_id is passed to the ESP8266NodeMCU module which is eventually sent to the online database. Once the message gets to the online database, it signals the end of attendance taking process to the web application and the need for the

web application **Akwapoly Journal of Communication and Scientific Research** to start sending emails to all those students who were absent when the attendance was taken and the lecturer of this particular course for
which attendance was taken. The course_id is extracted and stored in a string denoted as 'S' while at the same time an http request for sending email is initiated to the postdemo.php script on wed server with URL of "http://anietieattendancesystem.com.ng/testmail.php". This string 'S' is accepted using the argument of the php script and then the php script query is used extract the details of the students who were marked absent for this course for which attendance was just taken and email is sent to all those students. Specifically, the email states that the absence of the student has been noted in the attendance list.

The result on for the web app and online database

The online database created with PhpMyAdmin software is hosted on a web server with website name as anietieattendancesystem.com.ng, database name as anietiea_smartattendance and control panel URL as http://172.93.123.7:2082/. The admin can log into the control panel (cPanel) of the web app hosting platform and access the online database for the attendance system. Once the admin logs into the cPanel, he can have access to the details of every user (the students and the lecturers). Also, in the cPanel, the admin can navigate into phpMyAdmin database to locate the database schema of the attendance system which is saved with the name "anietiea_smartattendance". The database schema contains four basic tables used in the attendance web app and the four tables names, as shown in Figure 9 includes the course_attendance, attendance_taken, faculty and student.

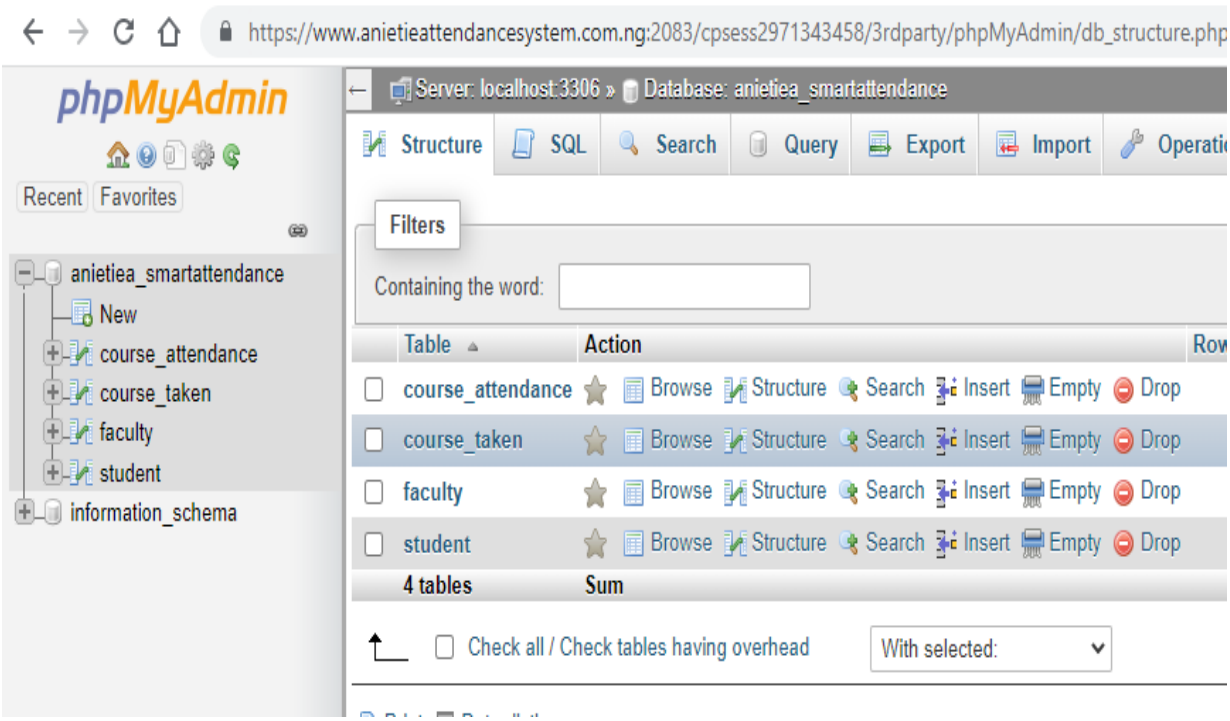


Figure 9 The screenshot showing the summarized database tables and the information schema for the attendance web application

The table titled 'faculty' (as shown in Figure 10) contains the details of lecturer and the course he takes and the record include the course_id, course_name, number of classes conducted which is denoted as 'lecs', the number of students that are registered for the course denoted as 'students', the course lecturer name, the faculty (lecuterer) identification number denoted as 'f_id' and the lecturer's login username and password. The lecturer username and password enable the course lecturer to access the student attendance report via the mobile App. Figure 10 shows the record for the course lecturer in charge of IOT course (with course_id as 605) , the lecturer name is Okon Effiong and he can login to view the student attendance summary using the faculty button on the mobile App.

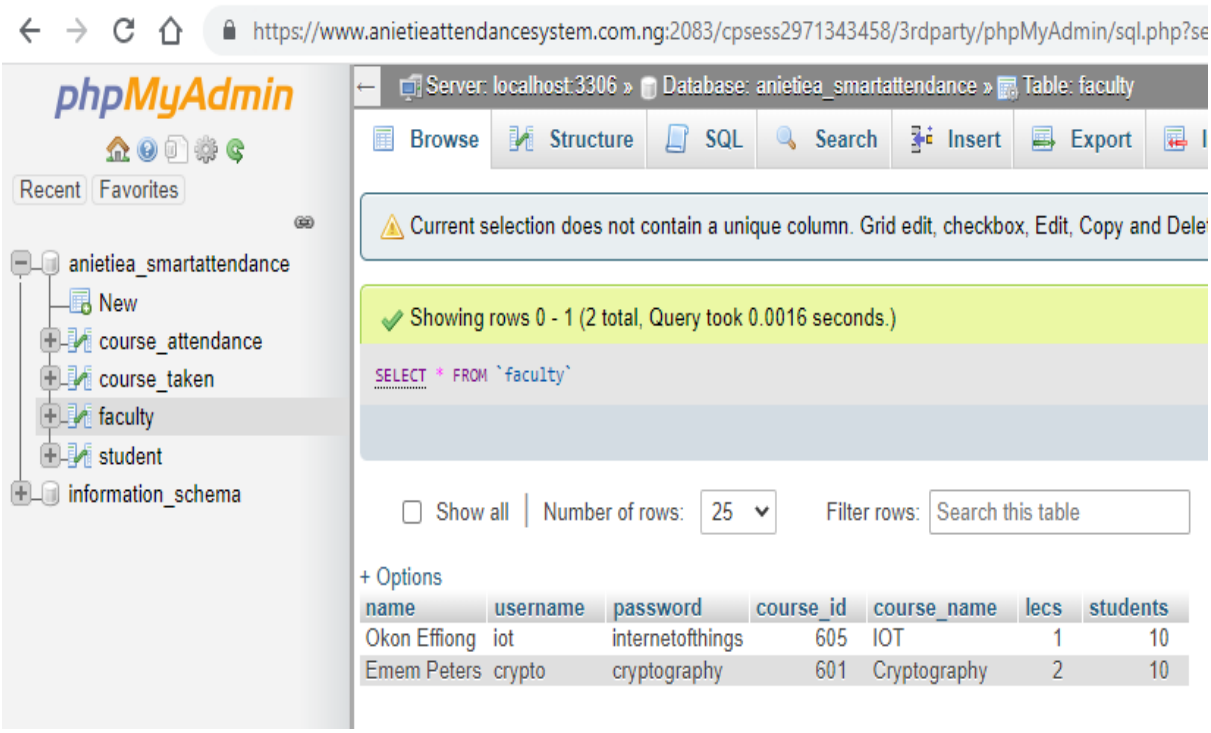


Figure 10 The screenshot of the database showing the 'faculty' table

The table titled 'student' (as shown in Table 4) contains the details of student which include the student finger_id, student_id, student name, student email and the student login username and password. When attendance is taken, the finger_id is matched with the one stored on the database table. If a match is found, the course attendance table on the database is updated with the attendance data marked "1". As shown in Table 4, the student with assigned finger_id of "9" takes attendance in class, the finger_id is matched with the one on the online database table and the attendance recorded on the course_attendance table for Emem with ginger_id of "9" is updated.

Table 4 The student table in the online database

finger_id	s_id	name	username	password	emailid
6	202101009	Vincent	202101009	vincolin	vincolin@gmail.com
3	202101042	Abigail	202101042	abisweet	abisweet@gmail.com
8	202101104	Ekomobong	202101104	bigekom	Bigekom@gmail.com
1	2021011	Brown	2021011	bfineforal	bfineforall@gmail.co

	54		54	l	m
9	2021011 57	Emem	2021011 57	newpeace	newpeace@gmail.co m
10	2021011 58	Nseobong	2021011 58	goldenhea rt	glodenheart@gmail.c om
2	2021012 05	Edidiong	2021012 05	iamblesse d	iamblessed@gmail.c om
7	2021014 19	Emediong	2021014 19	blessing	blessing001@gmail.c om
4	2021014 20	Rita	2021014 20	winingrita	winingrita@gmail.co m
5	2021014 54	Oscar	2021014 54	oscarbigie	oscarbigie@gmail.co m

The table titled ‘course_atendance’ is used to store all the finger_id that has been captured and matched with the one on the online database during the attendance taking process. The course_atendance table (as shown in Figure 11) contains the finger_id, course id for which attendance is being taken. The column marked “1” signifies the attendance marked present. The time at which the attendance is taken is also recorded. Figure below shows the student table as it is on the database.

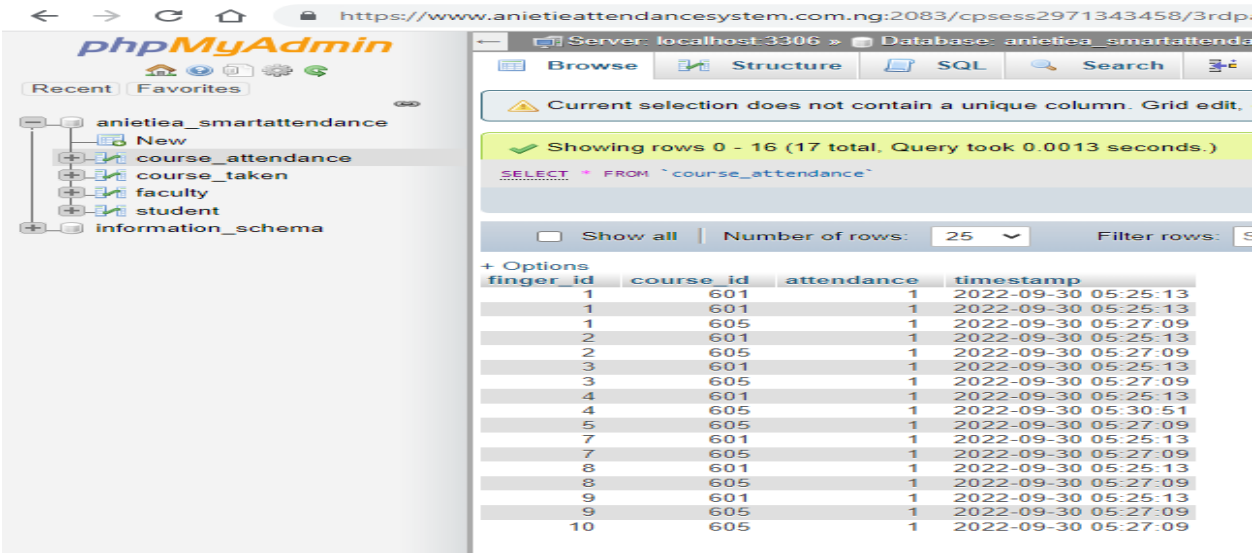


Figure 11 The screenshot of the database showing the course_attendance table

Furthermore, the screenshot of the database showing the course_taken table is shown in Figure 12 . The course_taken table contains the list of finger_id and the courses they attended. So a course will be listed for each student with

finger_id that attended the class for the course. When the "conn.php" scripts is called, the information stored in the course_taken table is retrieved and summarized on the student dashboard on the mobile App.

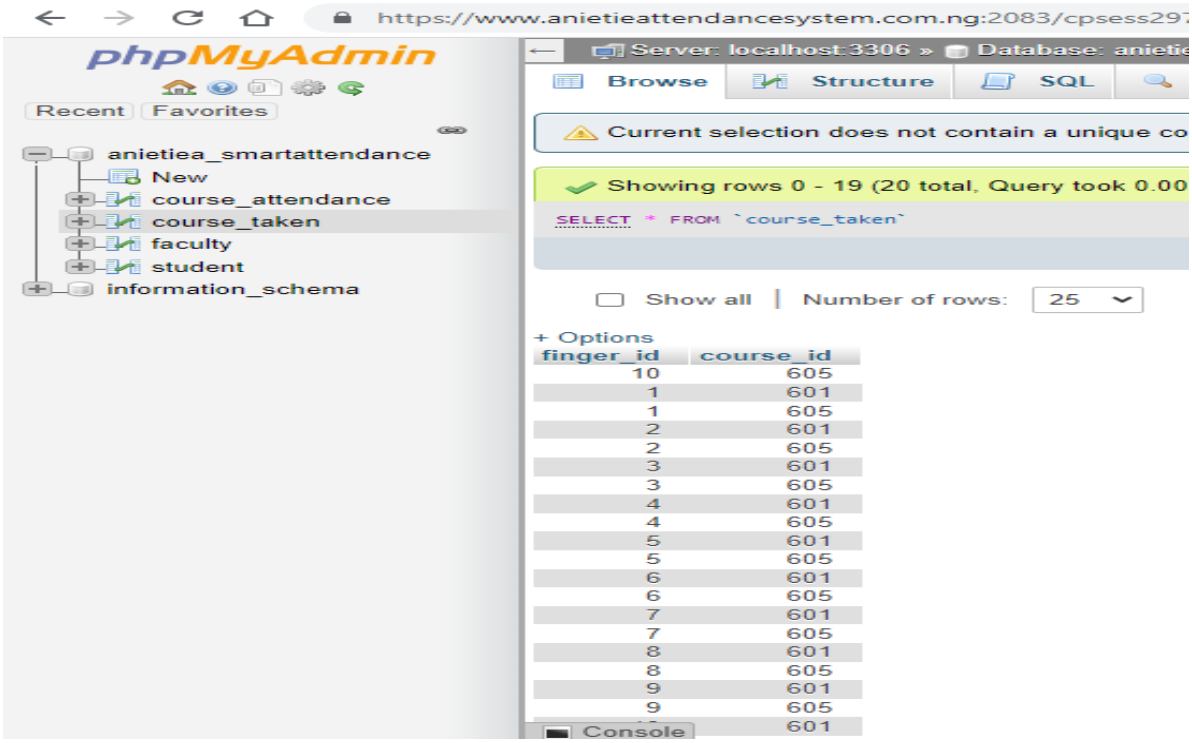


Figure 12 The screenshot of the database showing the course_taken table

3.3 The results for the Android mobile App

The Android application is built for the purpose of viewing the status of the attendance for the student as well as for the faculty (lecturer). To view the corresponding logs, the users need to log in using their respective credentials and depending on who the user is, the system functionalities are delivered. The screenshots of the android app showing the user login page and a sample login are shown in Figure 13.

Again, the Android App functionalities for the student is used to check the student's attendance for the courses that he has registered. The Android App functionalities for the Lecturer get the attendance statistics for each of the courses he teaches which
is available on the online database. The screenshots of the android app showing the student view page for his attendance and the lecturer's view page for the attendance in his course are shown in Figure 14.

A lecturer can log in using the "LOGIN AS FACULTY" button on the App. Upon successful login, a lecturer can see details of the Course he or she has taken. As shown in Figure 14, when the Lecturer taking the 'IOT' course logs in, he can see date and number of students in attendance for a particular date he lectured. He can also see the total number of students expected to be in his class. This is named "Strength" on the mobile app. Also, a student can also log

in to see the total number of attendances he or she has for each of the courses attended, as shown in Figure 14.

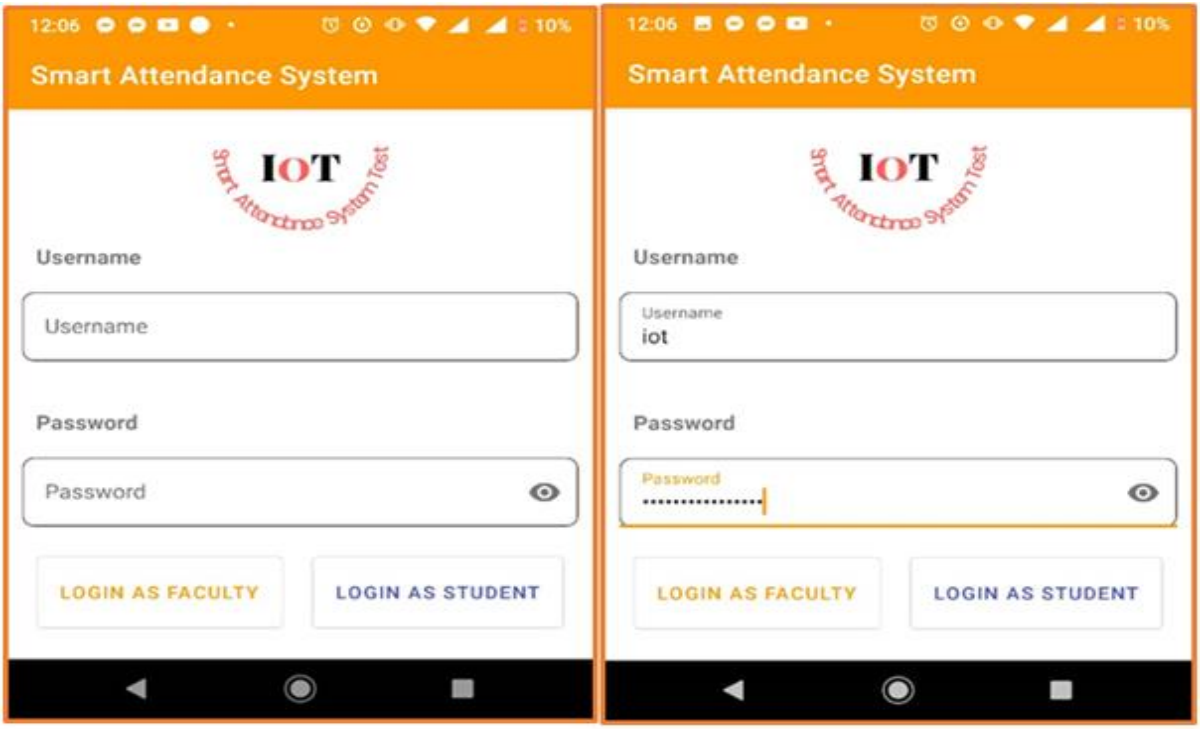


Figure 13 The screenshots of the android app showing the user login page and a sample login entry

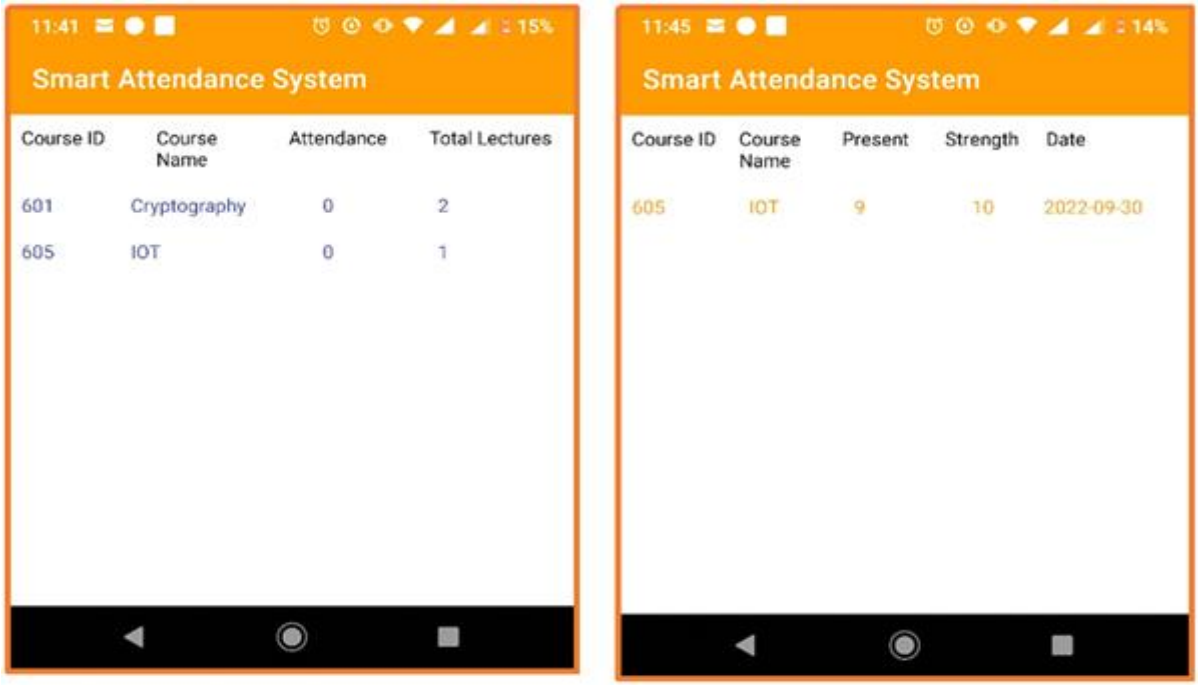


Figure 14 The screenshots of the android app showing the student view page for his attendance and the lecturer's view page for the attendance in his course

Conclusion

The development of a fingerprint-based class attendance management system is presented. The attendance system comprises of a fingerprint sensor, an OLED, an Arduino UNO board, and a NodeMCU ESP8266 module. The fingerprint sensor is interfaced with the Arduino UNO microcontroller which processes the fingerprint information and sends it via the NodeMCU to the online database. This information stored in the database are retrieved and viewed by the students and the faculty or lecturer via a mobile App. The Arduino and NodeMCU codes are written using the Arduino IDE. The server-side scripts are written in PHP.

A simple prototype of the attendance system is built and tested with two courses and 10 students in a class. The server is web server with domain name: anietieattendancessystem.com.ng and it is hosted on "litehost" platform. The results of the test showed that the system can be effectively used to conduct attendance management for different courses and student populations.

References

Adeniran, T. C. et.al (2019). Design and implementation of an automated attendance monitoring system for a Nigerian university using RFID. African J. Comput. ICT, 12(2), 72-89.

Akbar, M. et.al (2018, August). Face recognition and RFID verified attendance system. In 2018 International Conference on Computing, Electronics & Communications Engineering (iCCECE) (pp. 168-172). IEEE.

Akhtar, S., Warburton, S., & Xu, W. (2017). The use of an online learning and teaching system for monitoring computer aided design student participation and predicting student success. International Journal of Technology and Design Education, 27(2), 251-270.

Akinola, O. A., Olopade, S. O., & Afolabi, A. S. (2021). Development of mobile and desktop applications for a fingerprint-based attendance management system. Indones. J. Electr. Eng. Comput. Sci, 24(1), 570-580.

Anjum, R., & Kamble, V. (2017). Student Tracking and Attendance Monitoring System Using RFID. International Journal of Scientific Research in Computer Science, Engineering and Information Technology, 2(2).

Aravindhan, K.et.al (2021, March). Design of Attendance Monitoring System Using RFID. In 2021 7th International Conference on Advanced Computing and Communication Systems (ICACCS) (Vol. 1, pp. 1628-1631). IEEE.

Ball, S. J., & Grimaldi, E. (2022). Neoliberal education and the neoliberal digital classroom. Learning, Media and Technology, 47(2), 288-302.

Bhatti, K., Mughal, L., Khuhawar, F., & Memon, S. (2018). Smart attendance management system using face recognition. EAI Endorsed Transactions on Creative Technologies, 5(17).

Bore, N.et.al (2017, November). Towards blockchain-enabled school information hub. In Proceedings of the Ninth International Conference on Information and Communication Technologies and Development (pp. 1-4).

Chandramohan, J. et.al(2017). Attendance monitoring system of students based on biometric and gps tracking

Chipchase, L. et.al (2017). Conceptualising and measuring student disengagement in higher education: A synthesis of the literature. International Journal of Higher Education, 6(2), 31-42.

Doggrell, S. A. (2020). No apparent association between lecture attendance or accessing lecture recordings and academic outcomes in a medical laboratory science course. BMC medical education, 20(1), 1-12.

Ehikhamenle, M., & Okeke, R. O. (2017). Design And Construction of an RFID Based E-Attendance Register. International Journal of Engineering Research and General Science, 5(1), 161-172.

Fatah, A. F. A., Mohamad, R., & Rahman, F. Y. A. (2021, April). Student attendance system using an android based mobile application. In 2021 IEEE 11th IEEE Symposium on Computer Applications & Industrial Electronics (ISCAIE) (pp. 224-227). IEEE.

Gomes, C., Chanchal, S., Desai, T., & Jadhav, D. (2020). Class attendance management system using facial recognition. In ITM Web of Conferences (Vol. 32, p. 02001). EDP Sciences.

Gunawan, H., & Kadir, E. A. (2017, September). Integration protocol student academic information to campus RFID gate pass system. In 2017 4th International Conference on Electrical Engineering, Computer Science and Informatics (EECSI) (pp. 1-6). IEEE.

Islam, M. M., Hasan, M. K., Billah, M. M., & Uddin, M. M. (2017, December). Development of smartphone-based student attendance system. In 2017 IEEE Region 10 Humanitarian Technology Conference (R10-HTC) (pp. 230-233). IEEE.

Koppikar, U. et.al (2019, July). IoT based smart attendance monitoring system using RFID. In 2019 1st International Conference on advances in information technology (ICAIT) (pp. 193-197). IEEE.

Kotevski, Z., et.al (2018). On the technologies and systems for student attendance tracking. International Journal for Information Technology and Computer Science, 10(10), 44-52.

Lin, Z. H., & Li, Y. Z. (2019, January). Design and implementation of classroom attendance system based on video face recognition. In 2019 International Conference on Intelligent Transportation, Big Data & Smart City (ICITBS) (pp. 385-388). IEEE.

Maramis, G. D., & Rompas, P. T. D. (2018, February). Radio frequency identification (RFID) based employee attendance management system.

In IOP Conference Series: Materials Science and Engineering (Vol. 306, No. 1, p. 012045). IOP Publishing.

Mohandes, M. A. (2017). Class attendance management system using NFC mobile devices. Intelligent Automation & Soft Computing, 23(2), 251-259.

Ogata, H. et.al (2017). Learning analytics for e-book-based educational big data in higher education. In Smart sensors at the IoT frontier (pp. 327-350). Springer, Cham.

Okokpujie, K. et. al (2017,). A face recognition attendance system with GSM notification. In 2017 IEEE 3rd international conference on electro-technology for national development (NIGERCON) (pp. 239-244). IEEE.

Pavithra, M. P., Sarika, A. S., & Rajendran, A. (2021). Attendance Management System Using RFID Technology (No. 6175). EasyChair.

Payne, L. (2019). Student engagement: Three models for its investigation. Journal of Further and Higher Education, 43(5), 641-657.

Pei, Z. et.al (2019). Face recognition via deep learning using data augmentation based on orthogonal experiments. Electronics, 8(10), 1088.

Pereira, H., Carreira, R., Pinto, P., & Lopes, S. I. (2020, June). Hacking the RFID-based Authentication System of a University Campus on a Budget. In 2020 15th Iberian Conference on Information Systems and Technologies (CISTI) (pp. 1-5). IEEE.

Qureshi, K. et.al (2021). Internet of Things for education: A smart and secure system for schools monitoring and alerting. Computers & Electrical Engineering, 93, 107275.

Rabu, S. N. A. (2019). The design and implementation of student attendance tracking system using QR code card. In The International Scientific Conference eLearning and Software for Education (Vol. 3, pp. 154-161). " Carol I" National Defence University.

.Rahmatulloh, A., Gunawan, R., & Darmawan, I. (2019, March). Web Services to Overcome Interope.rability in Fingerprint-based Attendance System. In 2018 International Conference on Industrial Enterprise and System Engineering (IcoIESE 2018) (pp. 277-282). Atlantis Press.

Raj, J. T., & Sankar, J. (2017, December). IoT based smart school bus monitoring and notification system. In 2017 IEEE Region 10 Humanitarian Technology Conference (R10-HTC) (pp. 89-92). IEEE.

Rjeib, H. et.al (2018). Attendance and information system using RFID and web-based application for academic sector. International Journal of Advanced Computer Science and Applications, 9(1).

Rjeib, H. et.al (2018). Attendance and information system using RFID and web-based application for academic sector. International Journal of Advanced Computer Science and Applications, 9(1).

Rusdi, J. et. al (2020,). Student attendance using face recognition technology. In 2020 2nd International Conference on Cybernetics and Intelligent System (ICORIS) (pp. 1-4). IEEE.

Salim, O. et al. (2018, September). Class attendance management system using face recognition. In 2018 7th International conference on computer and communication engineering (ICCCE) (pp. 93-98). IEEE.

Shah, M., & Cheng, M. (2019). Exploring factors impacting student engagement in open access courses. Open Learning: The Journal of Open, Distance and e-Learning, 34(2), 187-202.

Singh, T. G. N. (2017). Automatic Gate Authentication System for RFID Tags and Thai ID Cards. Asian Institute of Technology, School of Engineering and Technology Thailand.

Sulistiyani, E., Ali, A. H. N., & Astuti, H. M. (2020). Change management strategies to implement a fingerprint based attendance system in information systems department using ADKAR model. Applied Technology and Computing Science Journal, 3(1), 22-29.

Surekha, B. et.al (2017). Attendance recording system using partial face recognition algorithm. In Intelligent techniques in signal processing for multimedia security (pp. 293-319). Springer, Cham.

Viegas, C.et.al (2018). Impact of a remote lab on teaching practices and student learning. Computers & Education, 126, 201-216.

Yadav, V., & Bhole, G. P. (2019, February). Cloud based smart attendance system for educational institutions. In 2019 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COMITCon) (pp. 97-102). IEEE.

Zaman, H. el.al (2017, July). RFID based attendance system. In 2017 8th International Conference on Computing, Communication and Networking Technologies (ICCCNT) (pp. 1-5). IEEE.

Zhao, M., Zhao, G., & Qu, M. (2022). College smart classroom attendance management system based on internet of things. Computational Intelligence and Neuroscience, 2022.