International Journal of Computing and Digital Systems
ISSN (2210-142X)
Int. J. Com. Dig. Sys. 15, No.1 (Mar-24)

http://dx.doi.org/10.12785/ijcds/150175

Design and Implementation of a Heterogeneous Distributed Database System for Students Absence Automation Using Portable Devices

Ahmed A. Alsamman[1]

[1]*College of Computer Science and Mathematics, University of Mosul, Mosul, Iraq*

Received 5 Apr. 2023, Revised 26 Jan. 2024, Accepted 7 Feb. 2024, Published 1 Mar. 2024

Abstract: Students' absence recording is a fundamental daily process that occurs within colleges, often relying on paper forms, which is inefficient when the college administration wants to gather absences from multiple remote departments (sites) and calculate the college's students' absence rates. In response to these challenges, a heterogeneous distributed database system was designed to declare and automate student absences in the colleges using smartphones or laptops and send absences in real-time from inside lecture halls or laboratories in departments to the college administration to generate absence reports. The system employed a client-server model to deliver services to the college and its various departments through the local area network (LAN) and the Internet. MSSQL database tables were designed and fragmented across college sites using horizontal fragmentation technology and then encrypted with the AES265 algorithm. The proposed system used private SQLite or MSSQL databases for the lecture's Android or Windows devices. On the other hand, college administration staff used a secure website built with ASP.NET to access the MySQL database server, which contains the absence table populated from all sites. The system was protected against false absence declarations by utilizing the local server's timing factor, allowing the lecturer only to declare absences on specific days and times. The experimental results showed high reliability and accuracy of the system and high satisfaction among the users about achieving the desired goals. The findings demonstrated a strong preference for the system among the lecturers over other systems that relied on student, biometric, and AI-based devices. At the same time, the college administration staff preferred automatic absence collection using the distributed design of the system over manual collection.

Keywords: Distributed Database, Heterogeneous Database, Smartphones, Portable Devices

1. Introduction

Students' absence reporting is a daily routine in educational establishments as it is a part of the student evaluation process. Traditional paper-based systems are still in use worldwide. Still, they are not practical in many respects, as they waste paper, require time to write names on paper, and sometimes the lecturers forget to send the absence sheets to the administration after the lecture, making the calculation of absence reports inaccurate. Moreover, these paper sheets will waste time again to read and transfer onto computers by administrative staff, with possible additional problems in understanding some of the handwritten names [1] [2]. As the demand for dependable, scalable, and easy-access information continues to rise, there is a growing need for client-server applications and distributed databases in the modern business environment. Because their data is spread out across multiple network sites, distributed database systems facilitate communication and data processing improvements. These advantages can be attributed to the decentralized nature of the system. Not only is data access quicker, but there is also a decreased likelihood of a single point of failure, and users have greater local control over the data [3]. The distributed database (DDB), in turn, can be described as a set of logically linked databases with each other and distributed over several sites via computer networks, as depicted in Figure 1 [4]. As a result of the development of corporate information and e-government, the system database of each department constitutes an actual heterogeneous database framework with its independence and autonomy inside the network system of many different functional departments [5]. In this research paper, a distributed system to report student absences will be designed and implemented in a college with multiple departments (sites) using smartphones, laptops, personal computers, and distributed heterogeneous databases to ensure data independence and security. The system is meant to help three groups: teaching staff, site administration staff, and college administration staff. It will also make it easier for teachers to declare absences and make their private reports of absences. Additionally, departmental administration staff will generate

E-mail address: ahmedalialsamman@uomosul.edu.iq

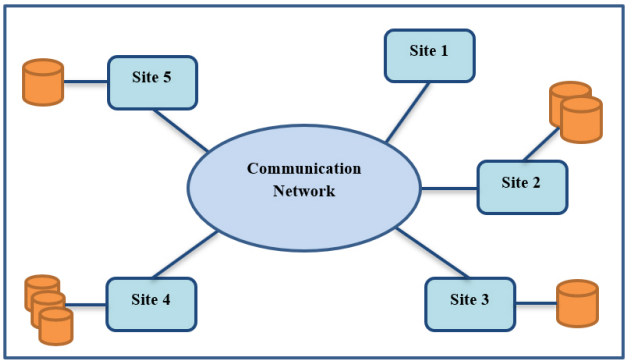

Figure 1. Distributed Database Environment

site absence reports. Finally, college administration staff will use it to obtain absence reports and percentage of absences. This paper will contribute to building a secure and reliable distributed system, working as a single unit that fits a distributed work environment, preserving data dependence in each department, and prioritizing the validity of recorded absences and the accuracy of absence reports.

2. Related Work

A lot of work has been dedicated to the problem of attendance documentation in educational facilities. Some of these works were based on human biometric features, utilizing the hardware capabilities of either smartphones or standalone devices. The hardware includes fingerprint readers, iris scanners, cameras, and microphones. Attendance documentation systems also use non-biometric devices like RFID readers, iBeacon, and barcode scanners. Other works have relied on QR codes, the Internet of Things (IoT), or technologies like Wi-Fi and Bluetooth to communicate between the lecturer and students. A paper developed an automated attendance recording system [6] that stores system data and attendance reports on Android devices and a MySQL web database server. Attendance check-in requires students to install the app on their phones. The system stores data in an Amazon Web Service (AWS) database. PHP was used to create an administration website. The system facilitates student attendance management but requires a smartphone with each student and is prone to cheating. A Bluetooth-connected smartphone and iBeacon attendance notification system were created in [7]. The administration managed student, subject, and staff data on a website. Students might use their cellphones to enter their class schedules through Bluetooth to submit attendance data to the web server, which would alert the lecturer. The system efficiently monitors student attendance but is also prone to cheating and requires iBeacon hardware. lecturers and administration Using RFID card readers, Arduino UNO, and a web-based interface [8], researchers designed an attendance management system for students and staff based on IoT. The system reads student or employee RFID cards, timestamps them, and sends them to a web server with a database server to record attendance and departure following authentication. Although the system was easy to use and synched with a real clock, it can be cheated in addition to the cost associated with Arduino Uno and RFID reader. Researchers in [2] created an east on-tap system that employed Wi-Fi peer-to-peer technology to interact with all students. The instructor's phone recorded and secured attendance. Once the lecturer collects attendance, students access the Wi-Fi and use their app to connect to the instructor's device to register their attendance (registration ID). The instructor's device discovers neighboring devices, collects registration IDs using the peer-to-peer connection, and stores them in their database before sending the attendance record to an administrative server. Although it is handy, it can be cheated and requires many smartphones. Barcodes are still active in attendance systems. A barcode-based student attendance tracking [9] is developed using various software tools, including Zint, Scan-IT to Office Program, and Microsoft Excel. The student's ID barcode is scanned to record attendance in the spreadsheet. The system eliminates data entry errors and improves efficiency but requires a barcode reader and can be cheated. In the same context, another system also relies on barcodes [10] that consist of three parts: an ID cards' barcode scanner using a web camera for attendance, an actions recorder for lecturers, and an administrative activities manager for academic staff. The system uniquely generates ID cards and lecturer accounts and sends notifications to parents via SMS or email. The software uses JavaFX for an adaptive and easy-to-use application. The system requires the installation of a web camera and is also prone to cheating. As QR codes became popular, many QR code-based attendance systems were created [11][12][13]. The system in [11] uses a geolocation feature and relies on a web server and MySQL database to process Android-powered student device requests. GPS and camera permissions are required to install the student's app. After identifying the student's and device numbers, the system verifies the student's location before scanning the QR code and confirming the student's attendance. In [12], a Visual Basic.NET application and QR code scanner are used to record attendance. Student's ID cards' QR codes are scanned upon entering the room, and the system validates the log before saving it to MySQL. Classroom reports can be generated by administrators using the administration module. The system was highly acceptable among students but required a QR code scanner and students' smartphones. The authors of [13] created a Python program that generates a unique 21x21 QR code, which is then distributed to each student via the Internet. Using the PyZBar Python library, they created a QR code scanner interface to scan and decode student attendance tracking codes. The system reduced the student attendance recording but required students to have smartphones. The proposed system in [14] used Google API for speech recognition. It records attendance by taking voice input through an app developed with Kivy and comparing it to voice samples stored in a database with a 95% accuracy rate. Although it is cost-effective and highly accurate, it is not 100 percent perfect. In [15], fingerprint recognition monitors student attendance. Integrated together are the Raspberry Pi, fingerprint sensor, and USB converter. After

scanning the finger, the database is used to identify the fingerprint. Reports will reflect the final attendance. The proposed system stores real-time data and displays it online for parents and other academic purposes. However, it is costly concerning the Arduino Uno, Raspberry Pi, and fingerprint scanner requirements and susceptible to errors. Another fingerprint-based attendance system [16], comprised of an ESP32 microcontroller, a fingerprint scanner, a keypad, and a buzzer, has been developed. The device operates in three modes: fingerprint registration, attendance, and class data input. The easy system achieved good results but was associated with cost. Several articles have designed face recognition-based attendance tracking systems utilizing deep learning and computer vision [17][18][19][20]. In [17], a Raspberry Pi, camera, and SD card with Raspbian were used to create a face recognition-based attendance and door control system. The Raspberry Pi receives images from the camera and passes them to a pretrained Python face recognition algorithm (LBPs) that opens the door and records the student's attendance in a MySQL database connected to a PHP website on a web server. Attendance results were reachable online by any web browser, which is a good feature, but using Raspberry Pi and a camera is costly. The proposed system in [18] includes creating a library of images of students as a first stage, followed by the training stage using the CNN library, and then using the camera to take real-time images of the students. Finally, the images are passed to the Matlab application, which recognizes the images using the CNN and R-CNN detectors. The system achieved good results with little cost but is prone to errors. In [19], a large classroom's attendance system stores student photos in a database to match classroom faces based on cosine similarity. Several facial photos in the database have enhanced the system's ability to differentiate between hazy and distant faces. The proposed system is fast enough for large rooms, with good results, but is not error-free. The author of [20] developed an attendance system using PyCharm for coding, QT Designer for interface design, MySQL for database management, and OpenCV face recognition algorithm. The system follows two stages, face recognition, and attendance recording, to fulfill attendance system functional requirements with 91% accuracy and real-time performance. It includes face detection, recognition, information entry, and attendance data checking, enhancing its structure and function. The system performance was acceptable but is prone to errors. An iris recognition-based hybrid system [21] employed deep learning algorithms and artificial neural networks to record the attendance of masked persons. The system reads the iris through a robotic arm equipped with an iris reader that can automatically and accurately align itself with the iris. It achieved an accuracy of 96.05 percent. This system is less effort and highly accurate but not error-free.

In the literature review, it was found that systems relying on student's ID cards or students' reactions through apps in their smartphones using Bluetooth or Wi-Fi to submit attendance directly or to scan barcodes or QR codes techniques are accurate in the process but practically inaccurate due to the system's susceptibility to cheating as students can masquerade the identity of others using their cards and phones to record illegitimate attendance. Also, these types of systems are costly and require each student to have a smartphone, which is not affordable for some. The systems based on standalone biometric devices like fingerprint or iris scanners can be expensive and not necessarily accurate due to recognition errors. The biometric systems utilizing smartphone hardware, like cameras, microphones, fingerprint scanners, and iris scanners, lack sufficient accuracy and may not even work on entry-level or cheap devices. Moreover, related works have not considered data distribution and security, so many sites (departments) cannot collaborate to serve a higher-level site. Compared to all previous studies, the proposed system in this research paper differs in several respects. First, the system uses distributed databases spread horizontally across several sites and encrypted with the AES256 algorithm, resulting in more security and fewer single-point failures. Second, the system can generate absence reports from a remote location via the Internet in a distributed manner. Finally, the system distinguishes itself by not depending on critical equipment that is either costly or cheatable, such as fingerprint scanners, iris scanners, personal identification cards, and students' phones, to assure the authenticity and correctness of the data. Therefore, the proposed system allows all departments to work together within one system while preserving the security and independence of data in each department and allowing the college to obtain students' absences automatically. On the other hand, it prioritizes the validity of absences and the accuracy of absence reports.

3. System Design Methodology

The proposed system's implementation depended on the correct analysis and design steps to ensure it worked well and kept maintenance costs to a minimum. This is because flaws or mistakes in the analysis process or logical design would show up in the final system with an actual payload and use.

A. Problem Statement

The traditional paper-based systems that colleges use to keep track of absences have some flaws. For example, some lecturers are slow to hand out absence sheets, so absence reports are often wrong. Paper systems are also slower than electronic ones. Furthermore, the college administration is disconnected from its sites and wastes time gathering manual data to create absence reports.

B. Requirements Analysis

Before starting the implementation of the system, some basic requirements must be analyzed, including:

1) Hardware

- A smartphone with at least Android operating system version 4.3.x (API 18).
- A server PC running the Windows operating system.

- A network device (a router) that can reserve a static IP for allocation to the server.
- A hosting website with a MySQL database server.
- A reliable Internet connection.

2) Software

- Microsoft SQL Server.
- Microsoft Visual Studio.
- Android Studio or any Android Integrated Development Environment (IDE).

3) Functional Analysis

The system involved three types of users (actors) associated with certain functionalities, as follows:

- Lecturer: A lecturer at a site who can login, send absences to local administration, and issue a class absence report.
- Head: The head of the department can login, read data, and issue a site absence report.
- Site Employee: A site administrative employee who can login, enter data, issue a site absence report, and send absences to the college administration.
- College Employee: A college administrative employee who can login, and generate college absence reports.
- Dean: The college dean who can login, browse absence totals, and generate college absence reports
- Admin: A college administrative employee who can manage users and perform database backup and restore.

Figure 2 shows the developed system's user case diagram (UCD).

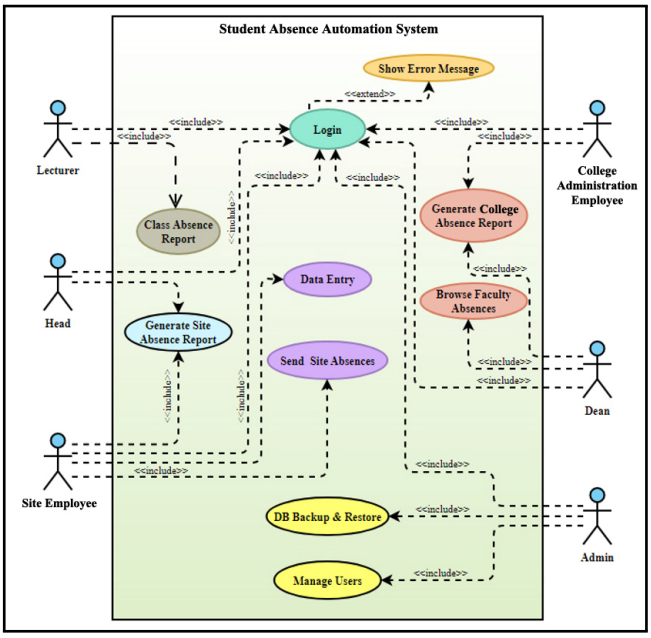

Figure 2. Use Case Diagram of the System

C. System Architecture

The system is based on the two-tier client-server model and employs heterogeneous distributed databases that are: an MSSQL database that is distributed horizontally over all sites; SQLite databases at each site; and a remote MySQL database server used by the college administration, as shown in Figure 3. The system consists of two main parts:

1) College Departments (sites)

It is a local area network (LAN) comprising a router, client devices, and an MSSQL database server. It relies on a two-tier client-server architecture:

- Client: A client can be either a lecturer or a site administration device.
  - Lecturer devices: This can be a smartphone (mobile handset) that works under the Android operating system with a local SQLite database server or a PC running a Windows operating system with a local MSSQL database server. Clients use a dedicated application to access data and a local database to keep track of absence reports related to their subjects. These devices are connected to the site's MSSQL server via Wi-Fi and utilize the server's time to keep synchronized.
  - Site administration devices: a Windows PC running a Windows application connected to a local server.
- Server: a Windows PC equipped with an MSSQL database server. This server's database is a data fragment with the same structure as other data fragments but with different data.

2) College Administration

It also relies on a two-tier client-server architecture and includes:

- Client (administrator): a PC running a website (a web client) to view college absences and issue absence reports, relying on the Internet.
- Server: a remote MySQL database server that enables administrative staff to read the data and issue the required absence reports.

The "ADMIN" privileged users in both departments and college administration interact with the system through a

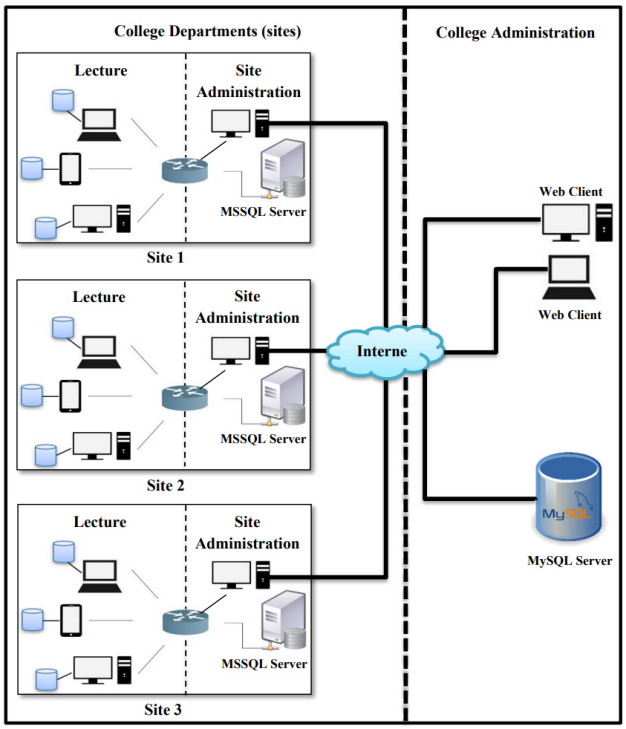

Figure 3. System Architecture

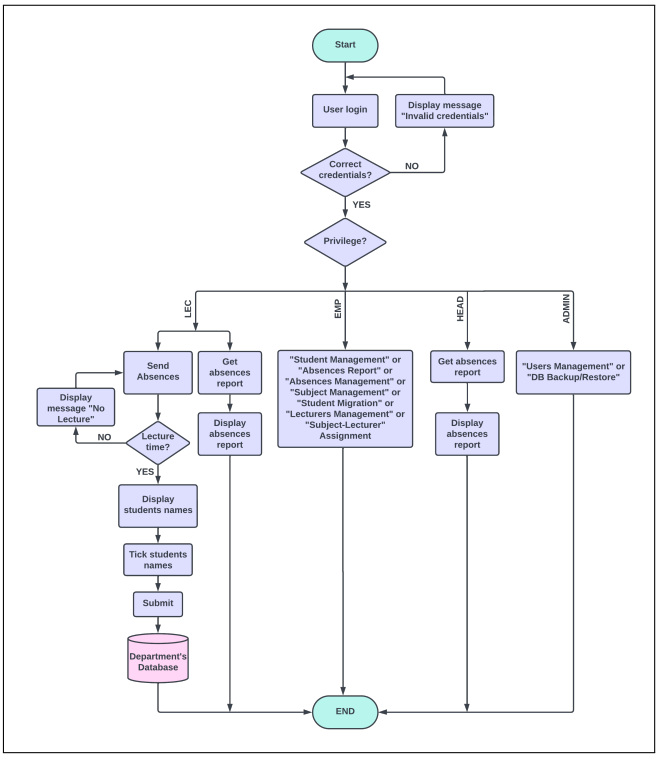

Figure 4. The Process Flow Diagram of Windows-Based Application

GUI to manage users, backup, and restore the database.

D. System's Process Flow Diagram

This section illustrates the process flow at each subsystem: Windows-based client application, Android-based client application, and web-based client application.

1) Windows-based application Process Flow

A multi-user application designed for departments to serve four types of users: the lecturers, employees, the head of the department, and the system administrator. Four privileges, "LEC", "EMP, "HEAD", and "ADMIN" are assigned to the lecturers, employees, the head of the department, and the system administrator, respectively. During the lecture (or lab), the lecturer opens the application from his smartphone, computer, or laptop, connected to a local area network (LAN) within the department. When logging in, users are first authenticated based on credentials (username and password), and secondly, the privileges are checked up to permit the user to perform specific actions. Graphical user interface (GUI) will be customized based on user privilege. "LEC" privileged users automatically get the current class (group of students) and subject's name (course) based on a weekday and time. Besides, when trying to use the system at other times outside of the lecture itself, a message will appear saying that the lecturer does not have a lecture at that time. As the lecturer taps on the current class (classes), a list of student names appears ready to be ticked and sent to the local server. This lecturer can also obtain absence reports for their students anytime and anywhere through their local SQLite database. "EMP" privileged users are responsible for feeding data to the site's server, including the names of students, lecturers, and the subject-lecturer assignments. Besides, they do other jobs like student migration and the generation of absence reports through a dedicated GUI to prepare the system to receive the student absences sent by the teaching staff (lecturers). They also send absences to the remote MySQL server. "HEAD" privileged users can issue department absence reports as needed. Figure 4 shows the system workflow of a Windows-based application.

2) Android-Based Application Process Flow

This application is only used by lecturers, so no privileges are assigned to the users. The user of this application will use their credentials to successfully log into the application and perform the same actions as the lecturers using a Windows-based application. Figure 5 illustrates the process flow of this subsystem.

3) Web-Based Application Process Flow

The users of this application work in the college administration and are assigned three types of privileges: "EMP", "DEAN", and "ADMIN". The user with "EMP" privilege uses the web application to generate college absence reports. "DEAN" privileged users can browse college absences and generate reports from the web application. Figure 6 illustrates the process flow of this subsystem.

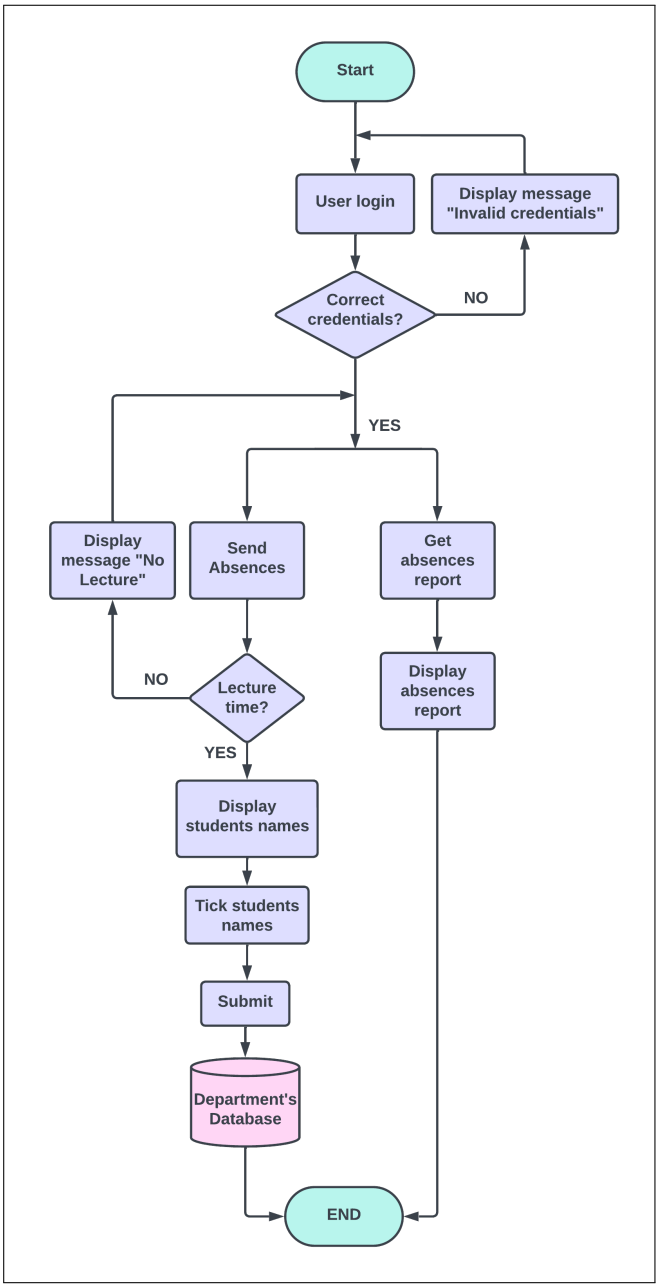

Figure 5. The Process Flow Diagram of Andoid-Based Application

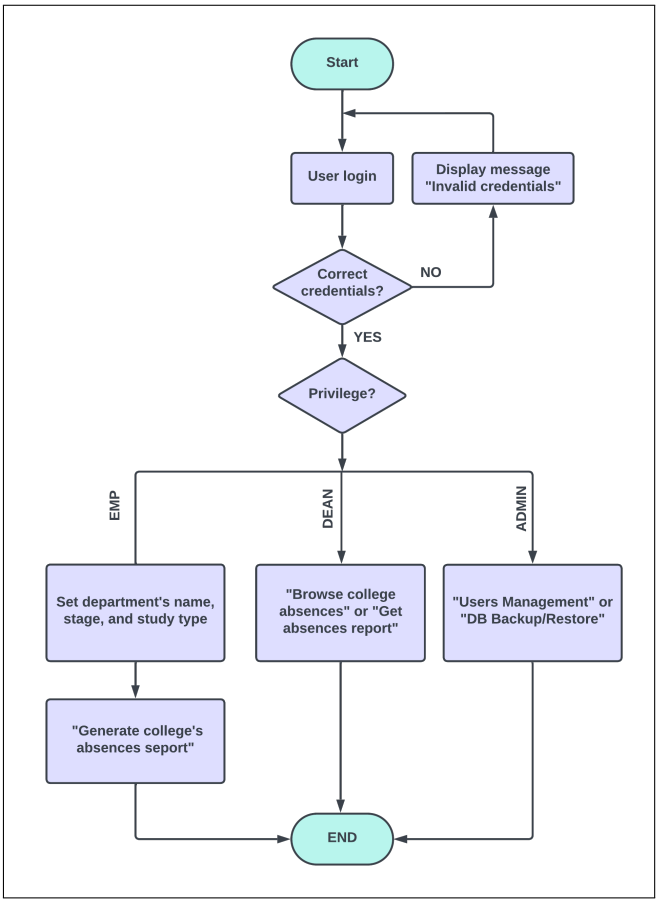

Figure 6. The Process Flow Diagram of Web-Based Application

E. Logical Design of the Database

Data were collected and analyzed using the traditional paper-based system and interviews with the department's administrative employees. After that, the main entities (logical tables) were identified according to their attributes. For the local administration's MSSQL database server, six entities were identified, which are: "Student", "Absence", "Subject", "Lecturer", "User", and "Assignment", as depicted in Fig. 7. Only one entity, "Absence", was designed for the client's local database (see Fig. 8), while three entities were designed for the remote (college administration) MySQL database, namely "Absence", "StageTotal", and "Users" (see Fig. 9). "Absence" relation (table) at the college administration is nothing but tuples that are submitted from each department administration via the Internet, as shown in Fig. 10. The same principle is applied to the "StageTotal" relation.

F. System Implementation

This phase of the design methodology begins with installing database management systems (DBMS), creating an MSSQL database at each site, and creating a MySQL database on the remote server. After this, physical tables corresponding to logical tables were created inside each site's database and lecturers' local SQLite databases, with the administration MySQL database server, using the appropriate SQL language. Finally, the graphic user interfaces (GUIs) for the mobile and PC applications were built and linked to the various databases of the system via database connections, according to the system architecture illustrated in Fig. 3.

- Graphical User Interface (GUI)
  The GUIs have been designed in such a way as to minimize human error using ComboBox and

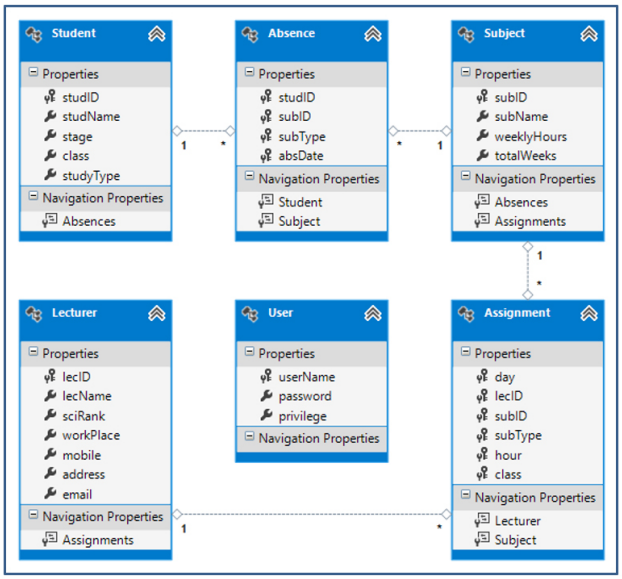

Figure 7. Site Administration Database Entities

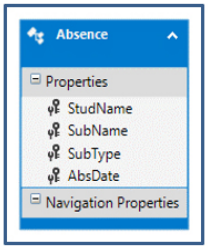

Figure 8. Lecturer's Private Database "Absence" Entity

CheckBox controls, which are the only ways in which lecturers and college administration staff can interface with the system. Local administration staff can also use the same controls and the TextBox control to feed data into the system.

- Android Application's GUI: It has many functional GUI's (see Figure 11) designed using Android Studio) programmed in Java to send absences to the local server and get absences reports anytime.
- Site's Desktop Application's GUI: This GUI was designed and programmed using Visual C# and includes several tabs, namely Students, Lecturers, Subject Assignment, Absences, Users, Subjects, Migration (student migration), Reports, Backup & Restore, About, and Sign Out (see Figure 12). Upon successful login, specific interfaces appear according to user privilege "HEAD", "LEC", and "ADMIN". For instance, the Backup Restore tab will only be visible to the "ADMIN" user.
- College Administration's Web Application GUI: This GUI was designed using ASP.NET and consists of three pages; Default.aspx, Ab-

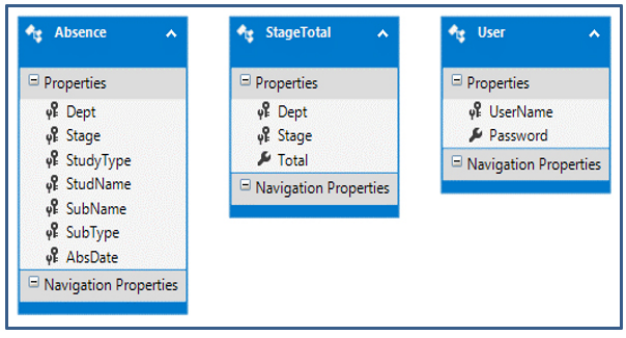

Figure 9. College Administration Entities

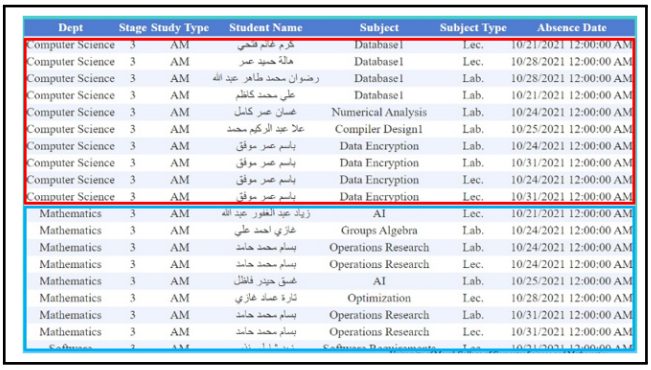

Figure 10. "Absence" Relation at the College Administration

Figure 11. Lecturer's Android Application GUI

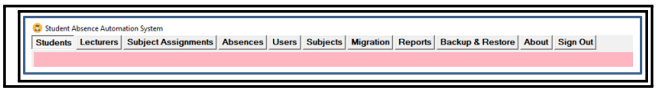

Figure 12. Desktop Application Tabs

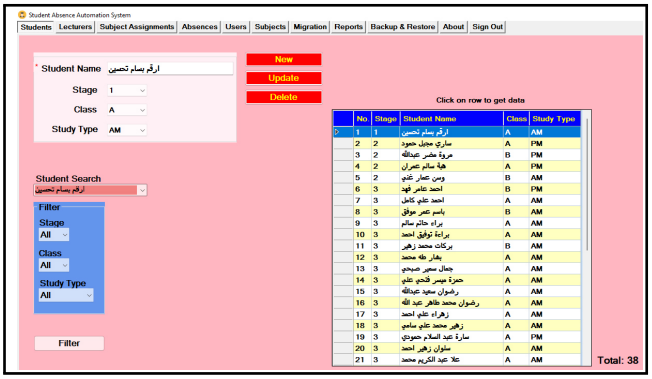

Figure 13. Students Tab

sence.aspx, and Report.aspx. The login page (default.aspx) displays the login screen and can be used by authorized college administration users to log into the website. In contrast, the absence page (Absence.aspx), lists college absences. Finally, the report page (Report.aspx) lists departmental absences and generates absence reports for the college.

The security of the local sites has been tweaked in the architecture and the physical design phase. Each site's server was first secured as residing on a local area network, and secondly, the database was set to use the AES265 encryption algorithm.

G. Monitoring and Tuning

The databases have been fed with test data, and every part of the system has been thoroughly tested and tuned several times in terms of both the GUI and physical design to ensure proper workflow.

4. Results

Three user groups: college administration, department, and teaching staff, have put the system through its paces with an actual payload. A lecturer successfully received student absence percentages; lists of student names and subjects (courses) were automatically obtained, and absences were also successfully sent to the department's management. Some of the lecturer's android GUIs are depicted in Figure 11. The departmental staff was successful in completing many crucial tasks, including managing students (Figure 13), allocating subjects to the lecturers (Figure 14), migrating students (Figure 15), and generating absence reports for the department, as shown in Figure 16. They could do this thanks to their own GUI, which allowed them to access absences sent by lecturers from the lecture halls. The college administration staff members could complete

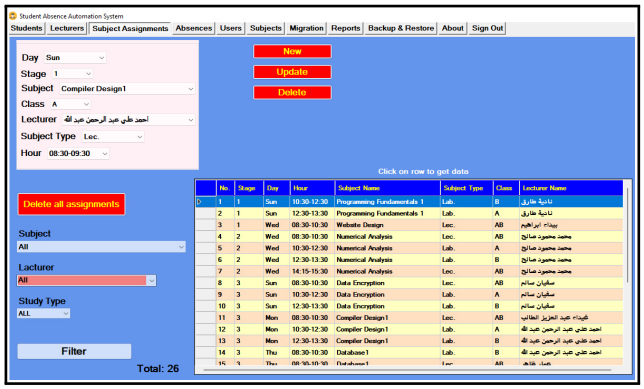

Figure 14. Subject Assignment Tab

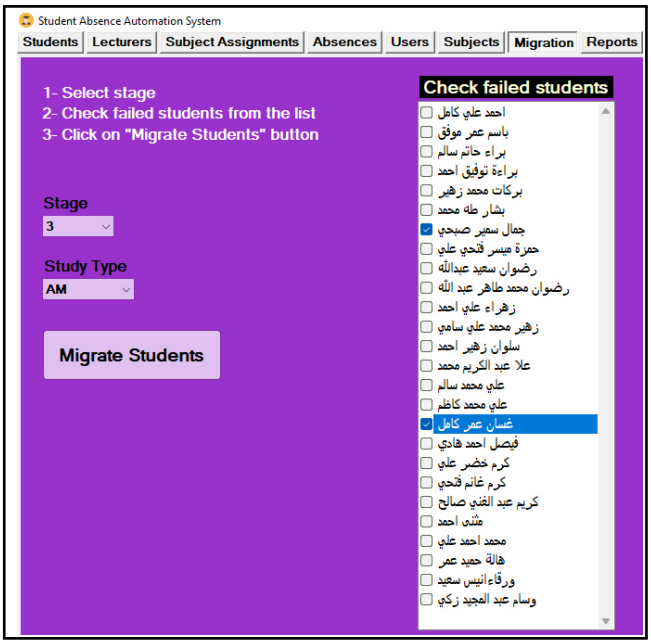

Figure 15. Migration Tab

their tasks using their web application GUI. The information regarding students' absences for all or specific departments is displayed in Figure 17. Also, the distributed report of absence percentages for the college was obtained, as depicted in Figure 18.

To evaluate the reliability of the developed system, three questionnaires, Q1, Q2, and Q3, were created for the three user groups, namely teaching staff, departmental administrative staff, and administrative staff in the college's administration, and emailed to each participant in the group. The questionnaires contained various evaluation statements and utilized a 5-point Likert scale with the following options: strongly disagree, disagree, neutral, agree, and strongly agree, to indicate the degree of agreement with each statement. The responses were statistically analyzed in terms of percentages using the SPSS program.

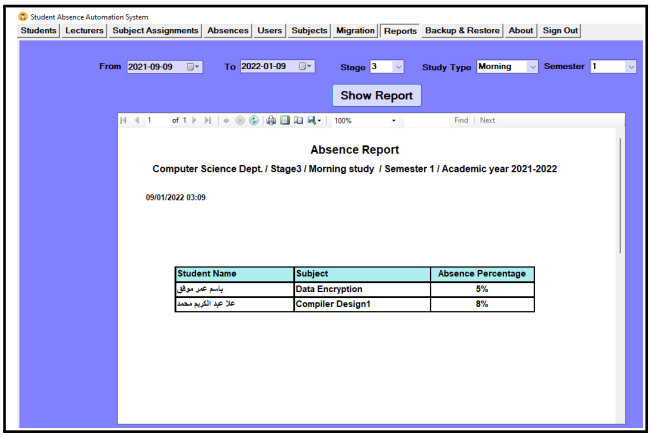

Figure 16. Reports Tab

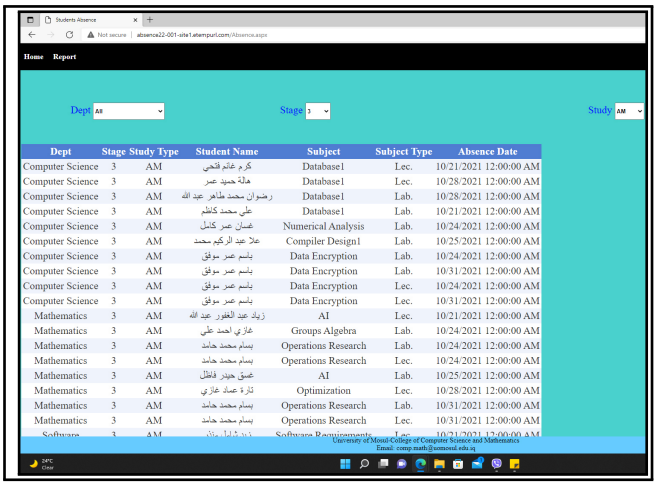

Figure 17. Absence Page

Teaching staff: a sample (N) of thirty lecturers were assigned a questionnaire (Q1) with a set (S) of seven statements, as follows:

S1: The system is crash-free.
S2: Everyone can use this system because it is simple to operate.
S3: The system is more truthful than systems based on student ID cards or phones.
S4: The system is more accurate than biometric-based and AI-based systems.
S5: For large halls, I prefer quick, semi-accurate gathering of absences over accurate, slower systems.
S6: Private students' absence reports generated by this system for my courses are accurate.
S7: The system meets the lecturer's needs regarding student absences.
Thirty responses were received. Table I lists the statistical analysis of Q1, and the visualization of this analysis is depicted in Figure 19.

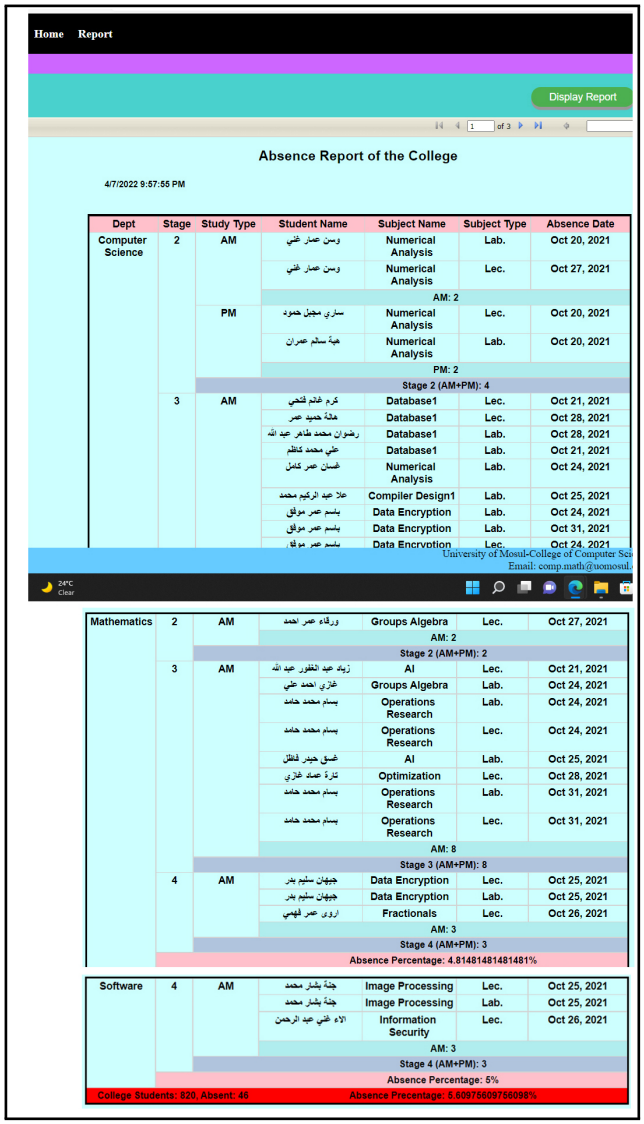

Figure 18. College Absences Report

TABLE I. Statistical Analysis of the Q1

S	Valid N	Percent %				
		SD	D	N	A	SA
S1	30				30.0	70.0
S2	30			6.7	40.0	53.3
S3	30			6.7	30.0	63.3
S4	30		6.7	6.7	53.3	33.3
S5	30	23.3	16.7	10.0	36.7	13.3
S6	30				36.7	63.3
S7	30			3.3	40.3	56.7

S: Statement, N: Sample size, SD: Strongly Disagree, D: Disagree, N: neutral, A: Agree, SA: Strongly Agree

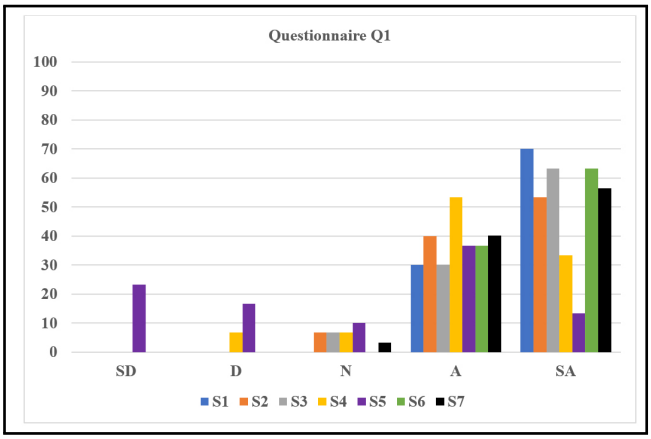

Figure 19. Visualization of Q1 Analysis

Departmental Administrative Staff: A sample (N) of thirty employees were assigned a questionnaire (Q2) with a set (S) of five statements, as follows:

S1: The system is crash-free.
S2: Everyone can use this system because it is simple to operate.
S3: Adopting the system assures high absence delivery rates and quicker response by the lecturers.
S4: The system generates accurate absence percentage reports for the department.
S5: The system meets the department's administration staff's needs regarding student absences.
Twenty-eight responses were received. Table II shows the statistical analysis of Q2 and visualization for this analysis is shown in Figure 20.

TABLE II. Statistical Analysis of the Q2

S	Valid N	Percent %				
		SD	D	N	A	SA
S1	28				28.6	71.4
S2	28		3.6	17.9	46.4	32.1
S3	28			7.1	53.6	39.3
S4	28				35.7	64.3
S5	28				60.7	39.3

S: Statement, N: Sample size, SD: Strongly Disagree, D: Disagree, N: neutral, A: Agree, SA: Strongly Agree

Administrative staff in the college's administration: A sample (N) of thirty employees were assigned a questionnaire (Q3) with a set (S) of four statements, as follows:

S1: The system is crash-free.
S2: Everyone can use this system because it is simple to operate.
S3: The system generates accurate absence percentage

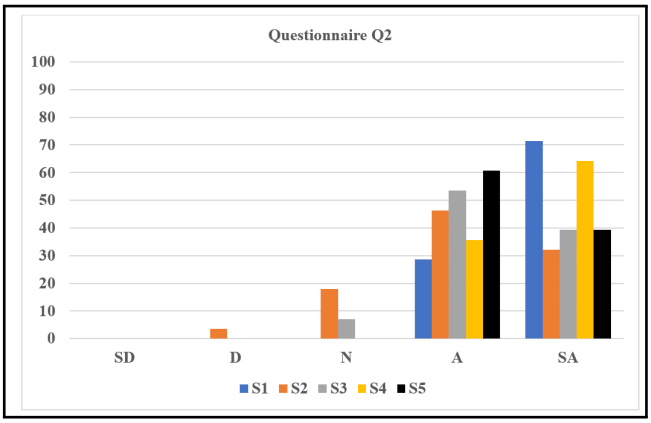

Figure 20. Visualization of Q2 Analysis

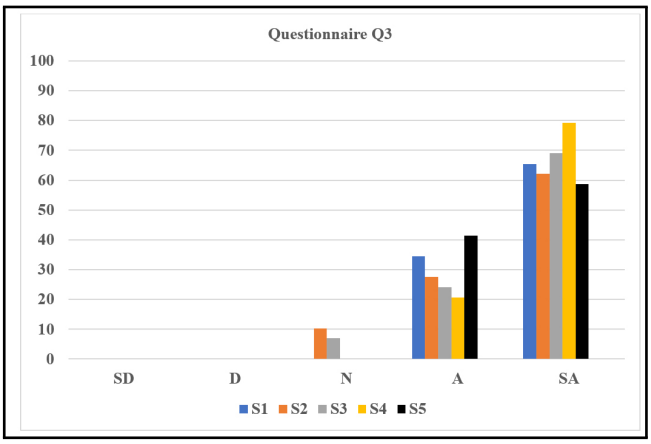

Figure 21. Visualization of Q3 Analysis

reports for the college.
S4: I prefer to have all departments' absences collected automatically by a single system as opposed to manually.
S5: The system meets the college's administrative staff's needs regarding student absences.
Twenty-nine responses were received. Table III depicts the statistical analysis of Q3 and visualization for this analysis is provided in Figure 21.

TABLE III. Statistical Analysis of the Q3

S	Valid N	Percent %				
		SD	D	N	A	SA
S1	29				34.5	65.5
S2	29			10.3	27.6	62.1
S3	29			6.9	24.1	69.0
S4	29				20.7	79.3
S5	29				41.4	58.6

S: Statement, N: Sample size, SD: Strongly Disagree, D: Disagree, N: neutral, A: Agree, SA: Strongly Agree

Each of the three questionnaires (Q1, Q2, and Q3) was

subjected to Cronbach's alpha reliability test, as shown in Table IV, where CA denotes the test value, and the number of items (statements) is denoted by N. The test results demonstrated that the three questionnaires are reliable and internally consistent.

TABLE IV. Reliability Statistics

Q1		Q2		Q3	
CA	N	CA	N	CA	N
0.778	7	0.716	5	0.731	5

CA: Cronbach's Alpha ($0.0 \leq CA \leq 1.0$), N: No. of statements

5. Discussion

The Cronbach's alpha (CA) test results in Table IV showed that Q1, Q2, and Q3 questionnaires had good reliability and internal consistency, scoring 0.778, 0.716, and 0.731, respectively. Generally, CA values greater than 0.5 are acceptable. The reliability of teaching staff (lecturers) responses was marginally higher (more reliable) than those of departmental and administrative staff. The vast majority of users strongly agreed with the system's error-free operation as well as coordinated with the system's usability, report accuracy, and satisfaction with user needs, according to the statistical analysis of the three questionnaires in Tables I&II&III. Although system usability looked very good, it was demonstrated that 94.3% of lecturers agreed with system usability versus 78.5% and 89.1% for the departmental and college administrative staff, respectively, because, from my point of view, 17.9% neutral responses related to unfamiliarity with comprehensive systems. In comparison to systems relied on students' phones via Internet [6], Bluetooth [7], Wi-Fi [2], barcode [9][10], QR code [11][12][13], and RFID cards [8], this one has been endorsed by 93.3 percent of lecturers in Q1 for its truthiness, and 86.6 percent of them think it is more accurate than biometric-based systems [15][16], and artificial intelligence-based systems (deep learning and computer vision) [17][18][19][20]. However, for large lecture halls, only 40% of lecturers -13.3% strongly agreed- preferred speed over accuracy, and 50% -23.3% strongly disagreed- would not forego accuracy. In another situation, 92% of employees in Q2 agreed that the system would ensure that lecturers would respond to absences more quickly and thoroughly. Also, 100 percent of employees in Q2 have certified the accuracy of absence reports and fulfillment of the work tasks. It was evident in Q3 that all employees (100 percent) preferred a single distributed system to automatically gather absence data from remote departments, which is the relevant feature of the proposed system. The proposed system began work after accurate data was fed into the system by the research sample of employees. The flawless design of the system contributed to facilitating the data entry process with a minimum of errors through the use of drop-down menus or selections. Putting the system under experimental use by the lecturers and employees after feeding it with data for about a month was important in effectively testing the system and highlighting its strengths or weaknesses. Obtaining reliable results from the questionnaires Q1, Q2, and Q3 required quite a bit of time, especially since the employees have job responsibilities on the one hand. On the other hand, patience in completing the questionnaire without rushing was emphasized because speed in such matters is disastrous and unreliable. As a consequence, reliable answers were obtained from the questionnaires.

6. Conclusions And Future Work

Recording absences is one of the basics in evaluating the academic careers of university students. Still, most colleges need help collecting these absences from departments that follow separate and independent systems. This paper contributed significantly to present a solution to these issues by building a distributed system suitable for the colleges' structure, consisting of several geographically separated departments. The system's sites operate as a single coherent unit via distributed database technology, saving all parties the time and effort of handing over and calculating absence percentages. The system employs predefined lecture assignments using a combination of weekday and time schedules that automatically allow the lecturer to receive a list of student names during the lecture. It also prevents absences from being declared outside the lecture's time and ensures validity. Testing results by the teaching staff and all employees demonstrate excellent reliability, usability, and accuracy. Most teachers prefer systems with credibility and high accuracy over systems that depend on biometric devices, student response, and artificial intelligence for small and medium classrooms. On the other hand, Not a small percentage of the teachers and staff preferred using these systems for only large classrooms, giving up only part of the accuracy and credibility for speed. The limitations of this study were that it was about a segment of employees who were not accustomed to information systems, which required giving them the basics of working on the system. Some showed reluctance or unwillingness to use the system at first, but things went normally after they were trained to use the system. So, training is crucial for some. In addition, some iOS-based smartphone users have been excluded from testing the Android app because their phones were not supported. According to the limitations and the results, the future direction is to improve the system by incorporating other platforms, notably iOS. Also, the system can be further improved by adding biometric or AI-based techniques that can be activated automatically for large halls and used when time matters.

7. Acknowledgment

I am incredibly grateful to my colleague Dr. Ghassan J. Mohammed for his support, the Scientific Committee in the Department of Computer Science/ College of Computer Science and Mathematics/ University of Mosul, and the Deanship of the College for their support. Finally, I would like to thank the anonymous reviewers for their valuable comments on the earlier version of this paper, which forced me to spend more time and effort enhancing it, and everyone who cooperated with me to complete the research.

References

[1] S. Budi, O. Karnalim, E. D. Handoyo, S. Santoso, H. Toba, H. Nguyen, and V. Malhotra, "Ibats - image based attendance system: A low cost solution to record student attendance in a classroom," in *2018 IEEE International Symposium on Multimedia (ISM)*, (Taichung, Taiwan), pp. 259–266, IEEE, December, 2018.

[2] A. Jaiswal, J. S. Mittapalli, B. R. Dilip, and S. Shridevi, "Android based – one tap attendance system," in *2020 International Conference on Emerging Trends in Information Technology and Engineering (ic-ETITE)*, (Vellore, India), pp. 1–7, IEEE, February, 2020.

[3] B. M. Abdelhafizr, "Distributed database using sharding database architecture," in *Asia-Pacific Conference on Computer Science and Data Engineering (CSDE)*, (Gold Coast, Australia), pp. 1–17, IEEE, December, 2020.

[4] M. T. Özsu and P. Valduriez, *Principles of Distributed Database Systems*. Cham, Switzerland: Springer International Publishing, 4 ed., 2020.

[5] Y. Zhu and Y. Li, "A data sharing and integration technology for heterogeneous databases," *International Journal of Circuits, Systems and Signal Processing*, vol. 16, no. 2, pp. 232–238, 2022.

[6] A. F. A. Fatah, R. Mohamad, F. Y. A. Rahman, and N. I. Shuhaimi, "Student attendance system using an android based mobile application," in *2021 IEEE 11th IEEE Symposium on Computer Applications Industrial Electronics (ISCAIE)*, (Penang, Malaysia), pp. 224–227, IEEE, April, 2021.

[7] M. A. Hidaya and H. M. Simalango, "Students attendance system and notification of college subject schedule based on classroom using ibeacon," in *2018 3rd International Conference on Information Technology, Information System and Electrical Engineering (ICITISEE)*, (Yogyakarta, Indonesia), pp. 253–258, IEEE, November, 2018.

[8] U. Koppikar, S. Hiremath, A. Shiralkar, A. Rajoor, and V. P. Baligar, "Iot based smart attendance monitoring system using rfid," in *2019 1st International Conference on Advances in Information Technology (ICAIT)*, (Farmingdale, NY, USA), pp. 193–197, July, 2019.

[9] A. E. A. A. Salah Elaskari, Muhammad Imran, "Using barcode to track student attendance and assets in higher education institutions," *Procedia Computer Science*, vol. 184, pp. 226–233, 2021.

[10] L. M. J. Ahmed Yaseen Mustafa, "Electronic student attendance management system based on id-card," *Sumer University Journal for Pure Science*, vol. 1, no. 1, pp. 56–66, November, 2022.

[11] T. H. T. Khang Jie Liew, "Qr code-based student attendance system," in *2021 2nd Asia Conference on Computers and Communications (ACCC)*, (Singapore), pp. 20–24, IEEE, September, 2021.

[12] D. M. Acasamoso, E. C. Avila, and S. Vargas, "Development and acceptability of a student daily attendance monitoring system," in *2021 IEEE 13th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment, and Management (HNICEM)*, (Manila, Philippines), pp. 1–4, 2021.

[13] S. V. Sai Harsith Reddy, G. R. Sekhar Raju, N. Jayanth, M. C. Sai, B. Pandey, G. G., and H. Gohel, "Design of qr based smart student attendance system," in *2023 IEEE 2nd International Conference on AI in Cybersecurity (ICAIC)*, (Houston, TX, USA), pp. 1–4, 2023.

[14] N. Sandhya, R. Vijaya Saraswathi, P. Preethi, K. Aarti Chowdary, M. Rishitha, and V. Sai Vaishnavi, "Smart attendance system using speech recognition," in *2022 4th International Conference on Smart Systems and Inventive Technology (ICSSIT)*, (Tirunelveli, India), pp. 144–149, January, 2022.

[15] R. Memane, P. Jadhav, J. Patil, S. Mathapati, and A. Pawar, "Attendance monitoring system using fingerprint authentication," in *2022 6th International Conference On Computing, Communication, Control And Automation (ICCUBEA*, (Pune, India), pp. 1–6, August, 2022.

[16] Maurizfa and T. Adiono, "Smart attendance recording device based on fingerprint identification," in *2021 International Symposium on Electronics and Smart Devices (ISESD)*, (Bandung, Indonesia), pp. 1–4, June, 2021.

[17] O. A. R. Salim, R. F. Olanrewaju, and W. A. Balogun, "Class attendance management system using face recognition," in *2018 7th International Conference on Computer and Communication Engineering (ICCCE)*, (Kuala Lumpur, Malaysia), pp. 93–98, IEEE, September, 2018.

[18] P. Asmitha and T. Sunitha, "Student attendance using face recognition technology," in *2022 International Conference on Intelligent Controller and Computing for Smart Power (ICICCSP)*, (Hyderabad, India), pp. 1–4, July, 2022.

[19] D. Mery, I. Mackenney, and E. Villalobos, "Student attendance system in crowded classrooms using a smartphone camera," in *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, (Waikoloa, HI, USA), pp. 857–866, IEEE, January, 2019.

[20] X. Chen, "Study on student attendance system based on face recognition," *Journal of Physics: Conference Series*, vol. 2492, no. 1, pp. 1–6, May, 2023.

[21] A. Bhardwaj, A. Mehta, A. M. Gaikwad, V. Vyas, and R. Mahajan, "Contactless mask monitoring and biometric attendance system," in *2022 International Conference on Signal and Information Processing (IConSIP)*, (Pune, India), pp. 1–5, August, 2022.

Ahmed A. Alsamman received an MSc in Computer Science from the University of Mosul in 2012. He joined the teaching staff at the computer science department, College of Computer Science and Mathematics at the University of Mosul in 2012 and started his academic career. He is a PhD student in the computer science department, College of Computer Science and Mathematics, University of Mosul. His research interests lie in information systems, IoT, image processing, and data security.