International Journal of Engineering & Technology, 7 (3) (2018) 1593-1598

SPC

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET
doi: 10.14419/ijet.v7i3.13974
Research paper

An Effective Attendance Monitoring System with Fraud Prevention Technique for Educational Institutions

Khandker M Qaiduzzaman[1*], Mohammad Shahjahan[2], Sadman Sobhan[3], Md. Shohel Arman[1], Manan Binth Taj Noor[1] and Mostafijur Rahman[1]

[1] *Department of Software Engineering, Daffodil International University, Dhaka, Bangladesh*
[2] *Oscillosoft Pty Limited, Dhaka, Bangladesh*
[3] *Dohatec New Media, Dhaka, Bangladesh*
[*] *Corresponding author E-mail:nafees.ku11@gmail.com*

Abstract

In this article, an Attendance Monitoring System (AMS) is proposed for educational institutions. The novel aspect of this work is to integrate a password-protected attendance counting module with the system. The module can effectively limit the fraudulent activities in attendance monitoring process. Besides that, some constraints are proposed to increase the security of the AMS. Based on the constraints a new system is designed to maintain the attendance data and the user information. The aim of this research is to design an effective AMS using the Radio Frequency Identification (RFID) technology to prevent fraudulence. Besides, the attention is given to the usability and the efficiency of the system. For this purpose, a usability testing named System Usability Scale (SUS) is used after implementing the system in the real world. The usability test result shows that the proposed system has an average SUS score of 71.92 out of 100. The score indicates that the proposed system has the adjective rating,"good" (range 71.4 – 85.5). A performance testing is conducted by calculating the average response time for the system. The response time is collected by varying the number of RFID tags and the lengths of the serial cable. The average response time of the proposed system is found 1.11 second which is better than the traditional paper-based system.

Keywords: *Attendance monitoring, fraud prevention, RFID, radio frequency identification, student attendance*

1. Introduction

Attendance Monitoring System (AMS) is widely used in most of the educational institutions of developed countries. It is a system that can identify the students and keep the track of their attendance automatically. This system can replace the paper-based attendance monitoring system which lacks reliability, accuracy, consistency, and efficiency [1].

In most of the AMSs, the front end consists of an identification unit. Generally, identifications are done based on various modern technologies such as RFID, biometrics, Bluetooth, face recognition, voice recognition etc. Among these, RFID is one of the most popular technology because of its cost efficiency, flexibility and sustainability [2]. As most of the educational institutions require student ID cards, it is beneficial to use the same ID card for counting the attendance. These type of ID cards are known as smart card tags.

An RFID module consists of a tag and a reader. The RFID tag stores and transmits data to the RFID reader. There are two types of RFID tag, the active tag, and the passive tag. The active tag contains an onboard battery. It does not depend on the reader for power supply and it can be used from a long distance. On the other hand, the passive tag does not carry any battery. It is lighter and smaller than the active tag. There are a microchip and a coil inside it. The chip has an internal EEPROM where a unique ID is stored. The coil works as an antenna.

An RFID reader acts as a transceiver. It sends an encoded radio signal to the tag. The coil inside the tag receives this signal. The signal carries both power and information which is extracted by the chip inside the tag. The tag answers to the signal by sending the tag ID to the reader via radio signal. The reader receives this signal and sends it to the processing unit.

In this paper, we proposed an AMS that uses RFID technology which is easier to interface with the processing unit, yet, effective. The proposed AMS can count attendance of the students. It can generate reports with detailed information. We also designed an application to manage the attendance data, student information, and the teacher information. But the most important part of our proposed system is to make it safe from the fraudulence. For this purpose, we defined some constraints by keeping the security concerns in mind. Our system is bound to follow these constraints to secure the monitoring process. Besides, we added a password protected attendance counting module with our system. The module should be activated and deactivated by using a secret password provided by a course teacher. All the students have to give their attendance within the activation and deactivation timespan of the module. Hence, it reduces the malpractices in attendance counting and also keeps the system easier to use.

The remainder of this paper is organized as follows. Section 2 demonstrates some recent related works on attendance monitoring system. The methodology of our system is presented in Section 3. The constraints and the solutions for our proposed system are discussed in Section 4. Section 5 presents the experimental setup of our system. The experimental results are discussed in Section 6. Finally, Section 7 concludes our work with some important remarks.

2. Related Works

In the recent years, AMS has become one of the extensively studied research with the growth of the educational institutions. In these institutions, RFID-based AMSs are becoming popular day by day. The basic structure of some RFID-based AMSs is described in [3, 4, 5]. Among these, in [3] an AMS is proposed that can track a student using pre-installed RFID readers on the different locations of the campus. In [4] a detailed synopsis of an investigation is done to check the practicality as well as the feasibility of RFID. In [5] an AMS is proposed to increase the efficiency of reporting, discipline and performance evaluation of the employees.

In the recent years, RFID based AMSs are being blended with the Internet of Things (IoT) technology. Some of the IOT based AMSs are proposed in [6, 7, 8, 9]. Among these, in [7] an RFID based AMS is proposed with wireless mesh network platform that can take attendance and track students. In [8] a comparison is shown among the current generation of automated attendance system and previously studied RFID based AMSs. Besides RFID, many researchers put attention on biometrics and face recognition techniques for designing AMSs such as in [10, 11, 12].

However, there is a lack of studies in the fraudulence management of RFID-based AMSs. Use of the Bluetooth, NFC, barcode, or GSM/GPRS technologies still lack security. Though some cutting-edge technologies such as biometric authentication, face detection, etc. can improve the authentication process, increases cost. Thus researchers are still trying to implement such system that uses cost-efficient technology with easier maintenance. Most importantly, the system must have the fraudulence detection mechanism.

3. Methodology

Our proposed AMS is divided into two distinct parts. One is the client module and another one is the server module. The system architecture of our system is shown in Fig. 1.

3.1. The Client Module

The client module is used by both the students and the teachers. There is a keypad integrated with it. When a lecture class is started, the teacher provides a secret password to the module using the keypad. Only after the password activation, the students can give attendance using their ID cards. The teacher can stop attendance counting whenever s/he wants by providing the password again. Therefore, students are restricted to give attendance only on the lecture time in front of the teacher. The password is set or updated by the course teacher only. An admin panel is provided on the server pc for this purpose. The ID card issued for each student works as an RFID tag. Whenever the ID card is brought to the RFID reader's range, the reader reads the unique tag id of the ID card. Then the tag id is sent to the Arduino, which is the micro-controller unit of our system. The Arduino further sends the tag id to the server pc. The server pc has an application that reads the tag id using the serial communication port. The application counts attendance and sends a notification back to the Arduino. The Arduino displays the response on the LCD. In case of successful response, the green LED is blinked as well as the buzzer is beeped.

The whole client module of the proposed system is independently operated by the Arduino board. It is programmed by following the Algorithm 1. Firstly, the Arduino, RFID and the LCD modules are initialized. Then a loop continuously looks for a key-press or the detection of an ID card. Whenever a key-press is detected it means that a user is trying to input a password. The program then reads the whole password and sends it to the server. Then it waits to get a feedback response from the server module. When the server sends a response, the client program shows the feedback in the LCD. On the other hand, if an RFID card is detected the program reads the tag ID from the ID card. The tag ID is a 5-byte number which is sent to the server module. Then the program waits for the response from the server. Finally, the response is displayed in the LCD.

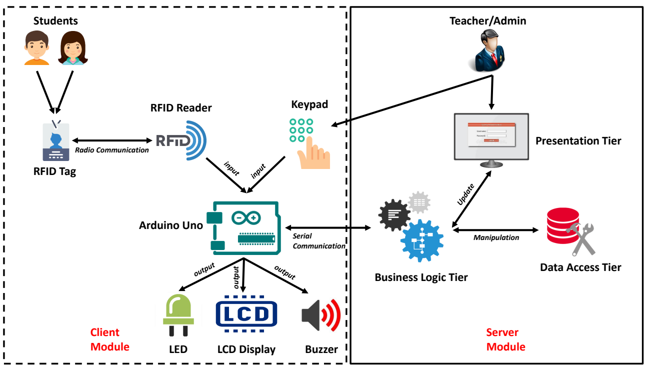

Figure 1: The system architecture of our proposed AMS. The black dashed box indicates the client module and the black solid box indicates the server module of the system.

3.2. The Server Module

The server module of our proposed system consists of an application and a database. The application is designed by following the N-Tier Architecture. The tiers are the data access tier, presentation tier, and the business logic tier. The data access tier contains the database initialization functions. The presentation tier contains the Graphical User Interface (GUI) which is accessed by the teachers only. The business logic tier contains the logical functions and the database queries. It also controls the data flow of the program. In our system, the presentation tier cannot access the data access tier directly. Only the business logic tier has the permission to access both the presentation tier and the data access tier and vice-versa. The directions of the data flow among the presentation tier, business logic tier and the data access tier are shown in Fig. 1.

Algorithm 1: CLIENTMODULE

```
initialize the Arduino
initialize the RFID
initialize the LCD
while true do
    if any key is pressed then
        password[] ← read all key input given by the user
        SerialWrite (password[])
        while response not received or not timed out do
            feedback[] ← SerialRead()
        end
        LCDWrite (feedback)
    else if RFID card is detected then
        tagID[] ← read all bytes of the tag ID
        SerialWrite (tagID[])
        while response not received or not timed out do
            feedback[] ← SerialRead()
        end
        LCDWrite (feedback)
    else
        LCDWrite ('Welcome')
end
```

We have shown how client module sends password and tag ID to the server module using Algorithm 1 in section 3.1. To receive these passwords and tag IDs properly, the server runs a synchronous program. The program is written by following the Algorithm 2. The program always listens the serial port of the server pc and whenever any data is available in the port it receives the data using a DataReceivedHandler function. Now, this data can be a password or a tag ID. If the data is a password it goes through a validation process. If the password is valid it is used as a search key to find the corresponding course in the database. It is better to mention that every course has a specific password, set by the course teachers at the beginning of the semester. However, if there is no lecture class being conducted at that point in time, a new attendance counting session is started for the corresponding course. Otherwise, if the corresponding course is already running, the system stops counting the attendance for it. On the other hand, if the data is a tag ID the system takes attendance for the corresponding student between the starting and stopping interval of the running course.

Algorithm 2: SERVERMODULE

```
initialize the serial port
open serial port
if DataReceivedHandler() is triggered then
    inData[] ← SerialRead()
    if inData[] is a password then
        password[] ← inData[]
        if password[] exists in database then
            courseCode[] ← get corresponding course code for
              the password[]
            if no active course then
                set courseCode[] as active course
                start attendance counting for courseCode[]
                SerialWrite('Attendance counting started')
            else if courseCode[] is the active course then
                stop attendance counting for courseCode[]
                SerialWrite ('Attendance counting ended')
            else
                SerialWrite('Another course is active')
            end
        else
            SerialWrite ('Wrong password')
        end
    end
    if inData[] is a tag ID then
        tagID[] ← inData[]
        if any course active then
            if tagID[] is enrolled in the active course then
                if attendance not counted for tagID[] then
                    count attendance for tagID[]
                    SerialWrite ('Attendance counted')
                end
                else
                    SerialWrite ('Attendance already counted')
                end
            end
            else
                SerialWrite ('Not enrolled')
            end
        end
        else
            SerialWrite ('No active course')
        end
    end
end
```

As the client module and the server module runs simultaneously, we can represent the whole working procedure in a flowchart shown in Figure 2.

4. The Proposed Constraints for AMS

In most of the cases, students find out the flaws of an AMS and bunk the lecture classes using it. As a result, one of the most challenging aspects of an AMS is the fraudulence detection and prevention technique. For this purpose, we proposed some constraints those should be followed by an AMS. If a constraint is denoted by c and the number of constraints is denoted by n, then set of constraints can be defined by $C = \{c1, c2, \ldots\ldots, cn\}$. the proposed constraints for our system are as follows:

- *c1* = Attendance of a course must not be taken by any person except the assigned course teacher
- *c2* = Attendance of a course must not be taken when the lecture class of another course runs
- *c3* = A student must not be able to give attendance out of the lecture time of a course
- *c4* = A student must not be able to give attendance of other days
- *c5* = A student must not be able to give attendance of other students
- *c6* = A student must not be able to give attendance for a course in which s/he is not enrolled
- *c7* = A student must not be able to give duplicate attendance for a course

Our proposed system follows each of the constraints of C. In Table 1 we explained the solutions for each of the constraints.

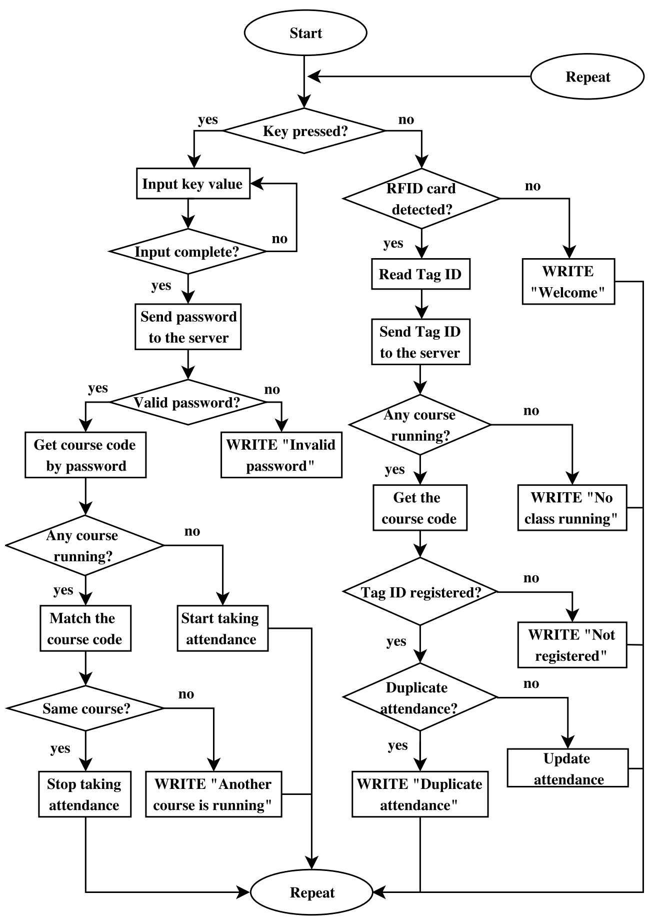

Figure 2: The flowchart of our proposed AMS.

Table 1: Solution by following each of the constraints

Const.	Solution
c1	In our system, every teacher can set a secret password on the server using an admin panel for their courses. Without this password, no attendance counting is started. So, except the course teacher, no one can count the attendance.
c2	In our system, a teacher comes in the classroom and enters the secret password of that course in the client module. The first entry of the password starts counting the attendance. Whenever the teacher ends the class, s/he provides the same password in the module again. The second entry of the password ends attendance counting. If any other password is given in between, the system will not allow it to access or interrupt the attendance of the running course. So, attendance of a course can not be taken when another class runs in the same classroom.
c3	In our system, attendance is allowed in the timespan of the first and the second entry of the password. As passwords are entered at the starting and the ending of a lecture class, no one can give attendance out of the lecture time.
c4	According to the solution of constraint *c3*, students must have to give attendance at lecture time. So there is no option to give attendance of other days. The admin panel of the server is also password protected. So students can not modify the attendance.
c5	We mounted the RFID reader module beside the teacher's desk. As the attendance is given in front of the teacher, no student can use more than one ID cards at the same time. Hence a student can not give attendance of other students.
c6	Each time a student attempts to give attendance of a course, the system checks whether s/he is enrolled in that course by a database query. The students who are not enrolled in the course are not allowed to give the attendance.
c7	Our system can detect duplicate entries. If any student attempts to give attendance more than one time for the same course, the system can detect it using a database query. The module denies to take attendance and gives an error notification.

5. Experimental Setup

The components used for the construction of the client module and the server module of our system is shown in Fig. 1. The client module includes RFID reader, RFID tag, Arduino board, keypad, LCD display, LED and buzzer. The RFID reader is based on MFRC-522 [13]. The RFID tag used for this work can be exerted to 13.56 MHz reader. For signal processing Arduino Uno [14] board is chosen which is programmed by Arduino software 1.8.5 (IDE) [15]. The keypad used in the client module is a 3 x 4 matrix keypad. A 16 x 2 LCD display is also connected to the Arduino board. We preferred RGB LED as an indicator and a digital buzzer module for generating audio alert.

The server module is implemented in a desktop pc. The pc has Intel(R) Core(TM) i5-2430M 2.40 GHz processor and 8.00 GB RAM. The operating system of the server pc is Windows 10 (64 bit). The pc always runs an application and a database server. The GUI of the application is developed by .NET framework. For logical development, C# programming language is used. We followed N-Tier Architecture for adding robustness and security to the application. Microsoft Visual Studio 2015 [16] is used for the total development of the application. The database is designed in Microsoft SQL Server Management Studio 2014 [17]. The experimental setup of our proposed system is shown in Fig. 3.

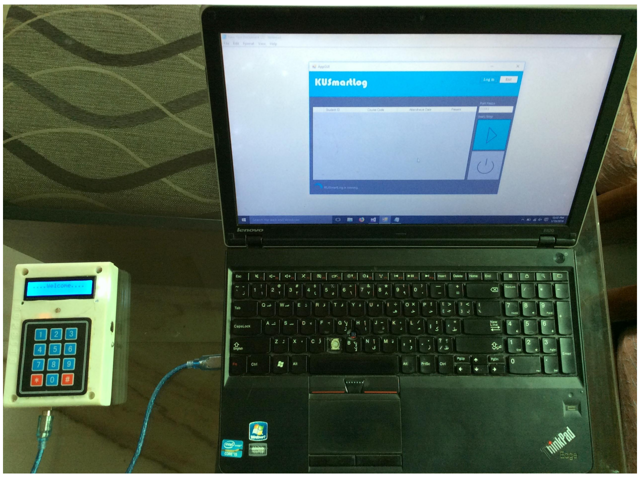

Figure 3: The experimental setup of our proposed AMS.

6. Results and Discussion

We implemented our proposed system in a classroom and took attendance entry from a different number of students. The response time of each entry was recorded for performance testing. To understand the usability of our system we also performed a usability testing.

6.1. Implementation

The system was successfully implemented in a real-world environment. The teachers could give the password to the client module and start or stop the attendance counting. The students could give their attendance using RFID tag and the attendances were counted in the server module, shown in Fig. 4(a). The features of the server application were accessed by the teachers only. They could use their username and password to log in the application. shown in Fig. 4(b). Fig. 4(c) shows the features of the application. The feature description of the application is given below:

6.1.1. Add Student

The admin can add information about new students using this option. The information can be updated from this option. It also prohibits any duplicate entry of student information.

6.1.2. Assign Course

Using this option an admin can assign different courses to a specific student. The admin can see all enrolled courses of a given student ID as well as delete the courses.

6.1.3. Show Students

It is a very useful option to see the information of all students existing in the system database. The student information can be filtered using student ID and the information can be printed from here.

6.1.4. Admin Profile

Using this option, the information of an admin's profile can be seen and updated. Also, new admin can be added using this window.

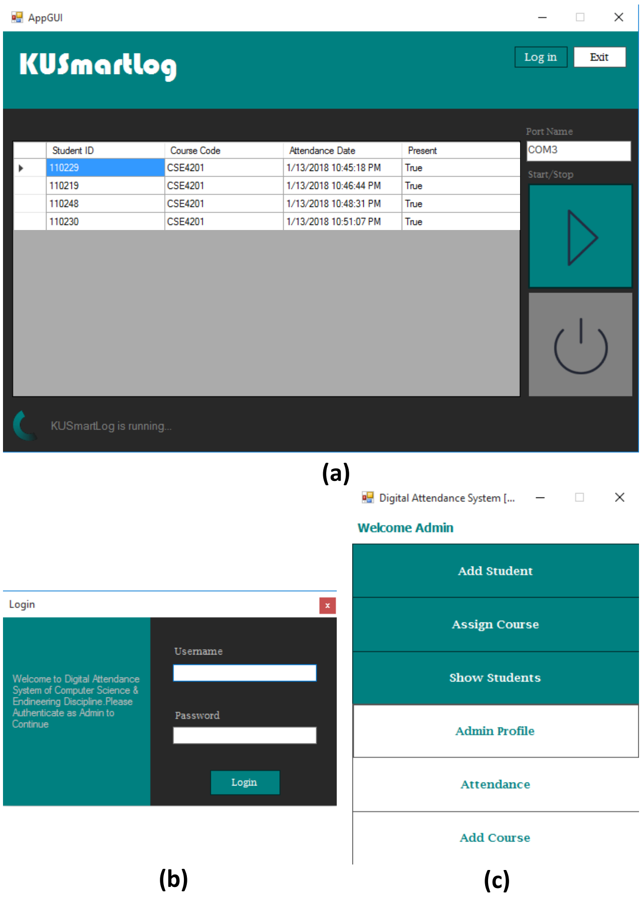

Figure 4: The forms of the server application. The main form of the application is shown in (a) where the attendance of a day is counted and displayed. The form is also used to start and stop the server application. An admin can access the applications by using a login form, shown in (b). After login s/he can use all of the features of the applications shown in (c).

6.1.5. Attendance

The attendance record can be seen from this option for a specific course. From this record, an attendance report can be generated and printed for hardcopy documentation.

6.1.6. Add Course

As courses are variable for a given department, this option is used to add course names and course IDs to the system database. Most importantly, it is used to set or edit course password which is vital for attendance counting. Courses can be dropped from this window too.

6.2. Usability Testing Result

To test the usability of our system we circulated questionnaire among 30 random students. The questionnaire was based on SUS usability testing [18]. Each of the students was asked 10 questions and their responses were recorded on a scale ranging from 1 to 5. The questions and the Score Contributions (S.C) by the students are shown in Fig. 5. Finally, the SUS score is calculated for each response. According to [18], all of the questions of the questionnaire are denoted as $item_1, item_2, ..., item_n$, where $n = 10$. For the odd and the even items, the S.C.s' were summed up using the Equation 1 and Equation 2 respectively. Then the SUS score is calculated using the Equation 3.

$$S.C.\ for\ odd\ items = \sum_{n=1,\ n\ odd}^{9} S.C.\ of\ item_n - 1 \qquad (1)$$

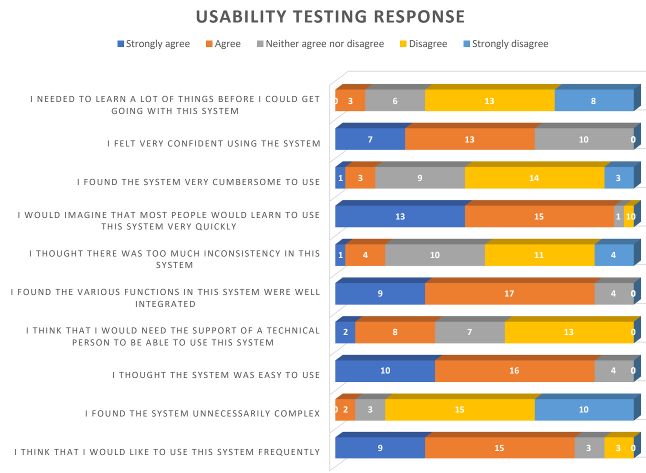

Figure 5: Usability testing response of our proposed system. For each of the questions, students could give a score 1 for strongly agree, 2 for agree, 3 for neither agree nor disagree, 4 for disagree and 5 for strongly disagree. The bars in the figure shows how many students gave what score for each question.

$$S.C.\ for\ even\ items = \sum_{n=2,\ n\ even}^{10} 5 - S.C.\ of\ item_n \qquad (2)$$

$$SUS\ score = (S.C.\ for\ odd\ items + S.C.\ for\ even\ items) * 2.5 \qquad (3)$$

The SUS score has a range of 0 to 100. According to our test results, the best and worst SUS scores found for our system are 82.5 and 60 respectively. The average SUS score for all of the responses for our system is 71.92 and the standard deviation among the SUS scores is 6.21. A survey is performed in [19], where an identical adjective rating scale is defined with respect to the SUS scores shown in Fig. 6. According to [19], any system having SUS score in between 71.4 and 85.5 and having standard deviation below 11.6 is considered as a "good" system. As the average SUS score of our system is 71.92 and the standard deviation among the SUS scores is 6.21, we can say that the adjective rating for our system is "good".

6.2.1. Performance Testing Result

We performed a system response testing for a different number of RFID tags. As the cable length has an impact on serial communication speed we also took the cable length in account for system response time analysis. The response time of our system for these cases are recorded. We used serial communication between the client

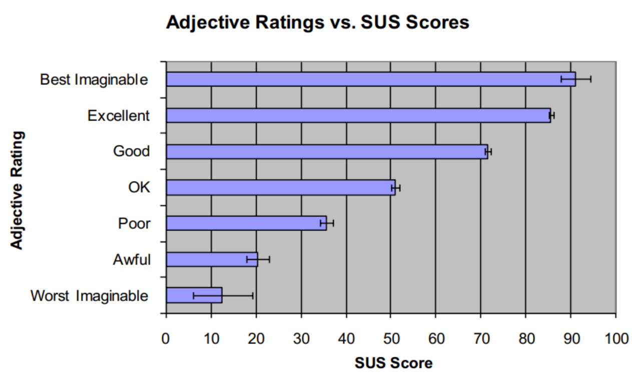

Figure 6: Mean SUS score ratings corresponding to the seven adjective ratings [19].

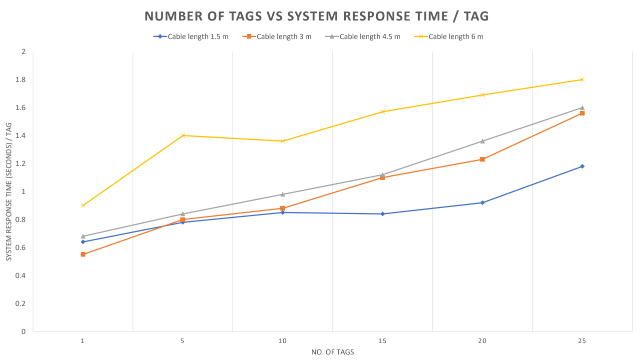

Figure 7: The system response time (seconds) per tag for different lengths of cable with respect to the number of tags

and the server module at standard 115200 bps baud rate. We used 1.5, 3, 4.5 and 6 meters of serial cable. For each of the cables, we recorded the response time for 1, 5, 10, 15, 20 and 25 tag(s) and calculated the response time per tag. The system response time per tag for different lengths of cable with respect to the number of tags is shown in Fig. 7. Finally, we calculated the average response time per tag for our system. The test result is shown in Table 2. The test result shows that our system has average response time of 0.87, 1.02, 1.1, and 1.45 seconds per tag for 1.5, 3, 4.5, and 6 meters of cable length. Hence, the response time of the overall system is calculated 1.11 second by finding the mean of the average response times per tag.

Table 2: Average response time (s) per tag of our proposed system.

Cable length (m)	Number of tags	Resp. time (s) per tag	Avg. resp. time (s) per tag
1.5	1	0.64	0.87
	5	0.78	
	10	0.85	
	15	0.84	
	20	0.92	
	25	1.18	
3	1	0.55	1.02
	5	0.8	
	10	0.88	
	15	1.1	
	20	1.23	
	25	1.56	
4.5	1	0.68	1.1
	5	0.84	
	10	0.98	
	15	1.12	
	20	1.36	
	25	1.6	
6	1	0.9	1.45
	5	1.4	
	10	1.36	
	15	1.57	
	20	1.69	
	25	1.8	

7. Conclusions

In this paper, we explained the basic architecture of our proposed AMS. Generally, the traditional paper-based system takes lots of time for data entry. It is also vulnerable to the human error. But, the major disadvantages of the paper-based system could be solved by our system. The system can take attendance within a second (almost) and generate reports accurately. Moreover, the implementation of the proposed constraints in the system made it more secured. The usability test result of our system reveals that it has "good" adjective rating. It concludes that our system is user-friendly. On the other hand, the performance test result shows that the average response time of our system is 1.11 second which is also satisfactory. Though increasing the cable length increases the response time of the system, still, it is acceptable.

However, there are some scopes of improvement for the proposed system in future. Such as, combining other low-cost but effective technologies with RFID may provide robustness to the system. We also noted that the system could take time to respond due to loose connections and server laggings. In future, we would like to solve these issues and search for alternative solutions.

References

[1] M. Kassim, H. Mazlan, N. Zaini, and M. K. Salleh, "Web-based student attendance system using rfid technology," in *2012 IEEE Control and System Graduate Research Colloquium*, July 2012, pp. 213–218.

[2] S. Nainan, R. Parekh, and T. Shah, "RFID technology based attendance management system," *CoRR*, vol. abs/1306.5381, 2013. [Online]. Available: http://arxiv.org/abs/1306.5381

[3] R. Anjum and V. Kamble, "Student tracking and attendance monitoring system using rfid," 2017.

[4] J. Dwivedi, A. Tyagi, A. Pushkar, D. K. Tiwari, R. Anand, S. Dubey, B. K. Dubey, P. Agnihotri, and H. Kanpur, "Rfid technology based attendance management system," *International Journal of Engineering Science and Computing (IJESC)*, vol. 7, no. 3, 2017.

[5] G. Maramis and P. Rompas, "Radio frequency identification (rfid) based employee attendance management system," in *IOP Conference Series: Materials Science and Engineering*, vol. 306, no. 1. IOP Publishing, 2018, p. 012045.

[6] E. Gupta, N. Singh, M. Saxena, K. Kartikey, and A. Sharma, "Smart attendance monitoring and counting system," 2018.

[7] C. Zulkifli, H. Hassan, A. Zalay, S. Kamis, and N. Hassan, "Integrated rfid technology and wireless mesh network platform for boarding school management," *Journal of Fundamental and Applied Sciences*, vol. 10, no. 6S, pp. 681–691, 2018.

[8] H. D. Rjeib, N. S. Ali, A. Al Farawn, B. Al-Sadawi, and H. Alsharqi, "Attendance and information system using rfid and web-based application for academic sector," *management*, vol. 9, no. 1, 2018.

[9] J. D. Irawan, E. Adriantantri, and A. Farid, "Rfid and iot for attendance monitoring system," in *MATEC Web of Conferences*, vol. 164. EDP Sciences, 2018, p. 01020.

[10] B. C. Melinamath, N. B. Jamadar, N. B. Patil, and O. S. Killedar, "Smart attendance application," 2018.

[11] G. Kavitha and K. Sankara, "Design and implementation of smart attendance system using raspberry pi," 2018.

[12] M. M. D. Savio and S. Yuvaraj, "Attendance marking system using face recognition & rfid and prevention of examination malpractice system," *International Journal of Engineering & Technology*, vol. 7, no. 2.8, pp. 599–602, 2018. [Online]. Available: https://www.sciencepubco.com/index.php/ijet/article/view/10541

[13] "Mfrc522 - nxp semiconductors," available online: https://www.nxp.com/docs/en/data-sheet/MFRC522.pdf, last visit: 23.05.2018.

[14] "Arduino uno rev3," available online: https://store.arduino.cc/usa/arduino-uno-rev3, last visit: 01.05.2018.

[15] "Arduino-software," available online: https://www.arduino.cc/en/Main/Software, last visit: 15.05.2018.

[16] "Welcome to visual studio 2015-msdn-microsoft," available online: https://msdn.microsoft.com/en-us/library/dd831853.aspx, last visit: 15.05.2018.

[17] "Download microsoft sql server 2014 express," available online: https://www.microsoft.com/en-us/download/details.aspx?id=42299, last visit: 10.05.2018.

[18] J. Brooke *et al.*, "Sus-a quick and dirty usability scale," *Usability evaluation in industry*, vol. 189, no. 194, pp. 4–7, 1996.

[19] A. Bangor, P. Kortum, and J. Miller, "Determining what individual sus scores mean: Adding an adjective rating scale," *Journal of usability studies*, vol. 4, no. 3, pp. 114–123, 2009.