Design and Implementation of an Automated Class Attendance System for Educational Purposes

Erdoğan Alper

Submitted to the
Institute of Graduate Studies and Research
in partial fulfillment of the requirements for the degree of

Master of Science
in
Information and Communication Technologies in Education

Eastern Mediterranean University
January 2019
Gazimağusa, North Cyprus

Approval of the Institute of Graduate Studies and Research

Assoc. Prof. Dr. Ali Hakan Ulusoy
Acting Director

I certify that this thesis satisfies all the requirements as a thesis for the degree of Master of Science in Information and Communication Technologies in Education.

Assoc. Prof. Dr. Ersun İşçioğlu
Acting Chair, Department of Computer Education and Instructional Technologies

We certify that we have read this thesis and that in our opinion it is fully adequate in scope and quality as a thesis for the degree of Master of Science in Information and Communication Technologies in Education.

Asst. Prof. Dr. Hüsnü Bayramoğlu
Supervisor

Examining Committee

1. Asst. Prof. Dr. Hüsnü Bayramoğlu

2. Asst. Prof. Dr. Emre Özen

3. Asst. Prof. Dr. Kamil Yurtkan

ABSTRACT

This research was conducted in order to easily get the attendance of the students at the Eastern Mediterranean University (EMU) in the Turkish Republic of Northern Cyprus. The research technique used in the project is Design Base Implementation Research (DBIR) and it is a combination of hardware and software system. The main purpose of this research is to reduce the time taken by the existing student attendance system on a paper or via calling the name of the students and to increase the efficiency of the attendance system. Thus, it aims that the time spent by the lecturer for the attendance taking to be effectively acquired in teaching and the taking attendance to be more accurate.

The developed system uses a microprocessor called Raspberry Pi. For the project, RC 522, which is a radio frequency identification (RFID) reader, and liquid crystal display (LCD) used. In addition, the application code for the Raspberry Pi is written with Python 2.7 programming language. This project contains five different phases. The first one is that the student read his/her card to the system. The second operation is the person's information which is processed by the system and the message is displayed on the LCD screen either as an error message or success message. The third operation is the transfer between Raspberry Pi and the server. The fourth operation is the server process where the database is available. With the information read from the card, attendance is also recorded to the Relational database management system (RDBMS) if it meets the requirements. Finally, the fifth process is web based management application that is available to student, teacher and administrators. The web-based

application is developed using the HTML, PHP and JavaScript programming languages.

This project offers an automated attendance system, which is faster and reliable, alternative to the current traditional methods of taking attendance at EMU.

Keywords: Attendance, Raspberry Pi, Python, RFID, LCD

ÖZ

Bu araştırma Kuzey Kıbrıs Türk Cumhuriyeti, Doğu Akdeniz Üniversitesi'nde (DAÜ) bulunan öğrencilerin yoklamalarını kolaylıkla alabilme amacıyla yapılmıştır. Projede bulunan araştırma tekniği tasarım temelli uygulama araştırması olup hem yazılım kodarı hem de donanım devrelerinden oluşmaktadır. Bu araştırmanın temel amacı, üniversite içerisinde bulunan kağıt üzerinde veya isim çağırarak yapılan mevcut öğrenci yoklama sisteminin aldığı zamanı azaltmak ve yoklama sisteminin verimliliğini artırmaktır. Böylece, öğretim görevlisinin yoklama için dersinden harcadığı zamanın öğretime etkili bir şekilde kazandırması ve yoklamanın daha doğru alınması hedeflenmiştir.

Proje, Raspberry Pi adında bir mikro işlemci tarafından çalışmaktadır. Bu projenin hayata geçirilmesi için Raspberry Pi ve içerisinde Python programlama dili ile yazılan koda ek olarak RC 522 denilen bir radyo frekansı ile tanımlama okuyucu ve sıvı kristal ekranı kullanılmıştır. Proje beş farklı işlem barındırmaktadır. Bunlardan ilki öğrencinin kartını okutup sisteme yoklamasını girmesidir. İkinci işlem, okutulmuş kişinin bilgilerinin sistem tarafından işlenmesi ve LCD ekranda dönen hata mesajı veya başarı mesajını görmesidir. Üçüncü işlem, Raspberry Pi ile sunucu arasında olan aktarma işlemidir. Dördüncü işlem ise sunucu işlemidir. Burada veri tabanı bulunmaktadır ve karttan okutulan bilgiler doğru kriterlere uygun ise yoklama buraya kaydedilir. Son olarak, beşinci işlem öğrenci, öğretmen ve yöneticilere sunulan web tabanlı bir uygulamadır. Web tabanlı uygulama ise HTML, PHP ve JavaScript programlama dilleri kullanılarak yapılmıştır.

Bu proje, DAÜ'nün mevcut geleneksel yoklama yöntemlerine alternatif olarak daha hızlı ve güvenilir bir otomatik yoklama sistemi sunmaktadır.

Anahtar Kelimeler: Yoklama, Raspberry Pi, Python, RFID, LCD

To my family
and
beloved ones

ACKNOWLEDGMENT

I would have never been able to finish my thesis without the guidance of my supervisor, help from my dearest girlfriend and friends, and support from my family members.

Firstly, I would like to thank to my supervisor, Asst. Prof. Dr. Hüsnü Bayramoğlu, for his patient guidance, encouragement, and advice he has provided during this thesis.

I would also like to take this opportunity to thank to my family members, especially my parents for opening this door of opportunity and for giving me strength to persist and succeed. They were always supporting me and encouraging me with their best wishes. In additionally, I would like to thank my friends and my colleagues from Credit West Bank, they were always with me with their help and best wishes. Finally, special thanks to Hatice Elagöz for her patience and support during my study.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF ABBREVIATIONS

AI	Artificial Intelligence
DBIR	Design Base Implementation Research
DC	Direct Current
EMU	Eastern Mediterranean University
ER	Entity Relationship
GSM	Global System for Mobile Communication
GUI	Graphical User Interface
HTML	Hypertext Markup Language
IC	Integrated Circuit
LBPH	Local Binary Patterns of Histogram
LCD	Liquid Crystal Display
NFC	Near-Field Communication
PCA	Principal Component Analysis
PHP	Hypertext Preprocessor
RAM	Random Access Memory
RDBMS	Relational Database Management System
RFID	Radio Frequency Identification
SD	Secure Digital
SMS	Short Message Service
USB	Universal Serial Bus
W3C	World Wide Web Consortium
WHATWG	Web Hypertext Application Technology Working Group

Chapter 1

INTRODUCTION

The current traditional systems used for tracking attendance such as calling names and then marking present or absent accordingly, appear to be time consuming, inconsistent and also requires paper work for the lecturers. Moreover, it has various other drawbacks such as class attendance are not secure and it is more likely for lecturers to do errors when attendance is taken verbally and visually (Kazi et al., 2017). It is also believed that taking class attendance manually might ruin class routine and discipline. In addition, in order to make sure of the student's presence at the end of the period, lecturers may need to recheck the status of the students manually by checking the updated attendance list that shows the matching weights during or after class (Choi, Park, & Yi, 2015).

It is very important that attendance data is kept securely and coordinately to avoid problems in recording and analyzing attendance data. The factors that are dependent variable of the value of attendance data are accuracy, speed, consistency and accessibility of the data. In paper-based systems, the data coordination problems are caused by many individuals being responsible for attendance data where there is a lack of central point where all the attendance data is collected together. Therefore, accessing to student's attendance records by individuals who have an interest in student attendance becomes more difficult. From a practical point of view, data might also be missing from centralized storage even if such storages are exist. In addition,

attendance trends and patterns such as problematic timetabling might be difficult to observe and analyze because of the nature of paper-based attendance systems (Bowen et al., 2005). Traditional attendance system is still used in Eastern Mediterranean University.

To resolve these problems, this study offers a card based attendance system which combines the Radio Frequency Identification (RFID) and Raspberry Pi technologies and allows lecturers to take students' attendance with their identification cards that are scanned on the RFID readers attached to the Raspberry Pi within the class.

Nowadays, the number of applications based on RFID cards are projected to rise which effectively applied to the attendance system. For an environment such as Eastern Mediterranean University (EMU), which is largely based around computer systems, lecturers spent particular amount of time for taking the attendance and have difficulties for the name pronunciations of the students.

1.1 Definition of Radio-Frequency Identification

RFID is a short-range wireless technology that uses electromagnetic radiation to automatically identify a person or object. The RFID system is made up of two main components: a transponder (e.g. tag, card) which holds data and interrogator or reader which reads data, as shown in Figure 1 (Finkenzeller, 2010). It also consists middleware (interface) which forwards data.

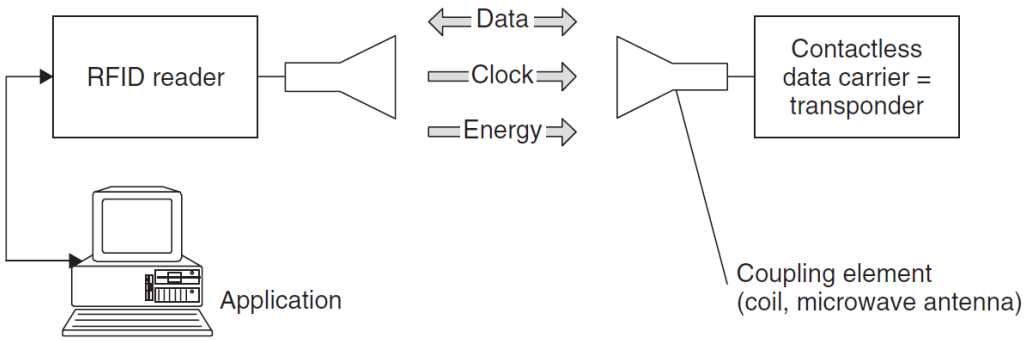

Figure 1. Main Components of an RFID System

There are 3 types of RFID tags that can be used. They are passive, semi-passive and active tags. The passive tags are the most commonly used and least expensive tags that requires no power source such as battery within the tag. On the other hand, in order to achieve better performance, semi-passive tags require battery built into the tag. In contrast, active tags are able to generate radio waves even in the absence of an RFID reader, but requires batteries for the whole systems (Srivastava, 2005).

Over the past couple of decades, there has been a considerable increase in the use of RFID technology. This technology is extensively used in modern industries such as public transport, fare systems using electronic payment, access control mechanisms, transport systems and animal identification. Other applications include secure toll payments, inventory management systems, contactless payments systems and electronic passports (Finkenzeller, 2010). The history of RFID technology is as shown in the Figure 2 below (Srivastava, 2005).

Decade	Event
1940-1950	Radar defined and used. Major World War II development efforts. RFID invented in about 1948.
1950-1960	Early explorations of RFID technology. Laboratory experiments.
1960-1970	Development of the theory of RFID. Early field trials.
1970-1980	Explosion of RFID development. Tests of RFID accelerate. Early adopter implementation of RFID.
1980-1990	Commercial RFID applications enter the mainstream
1990-2000	Emergence of standards. RFID more widely deployed.
2000-2010	Innovative applications emerge. Combination of RFID with personal mobile services. Subcutaneous RFID emerges for animals, humans. RFID becomes part of daily life.

Figure 2. The Evolution of RFID Technology

RFID systems are used to describe living things and objects in motion using radio frequencies. For example, people would be able to pay their fees just by scanning their cards on the bus or employees would be able to access their casing machine with these cards provided to them. In these situations, the RFID system provides more secure and easier access to the user. Another example of RFID tags used in industries, is an RFID tag attached to an automobile during production that can be used to track its progress through the assembly line or implanting RFID microchips in pets allows identification of animals.

1.2 Single-Board Computer Technologies

The Raspberry Pi is a small single-board computer developed in the United Kingdom by the Raspberry Pi Foundation. The main aim is to open up the world of computing and programming by creating a hardware device which is affordable and accessible to everyone (Robinson et al., 2014). There are different models of the Raspberry Pi. The Figure 3 and Figure 4 shows the picture (Upton & Halfacree,

2016) and the mechanical drawing (Raspberry Pi Foundation, 2018) of the Raspberry Pi 3 Model B+.

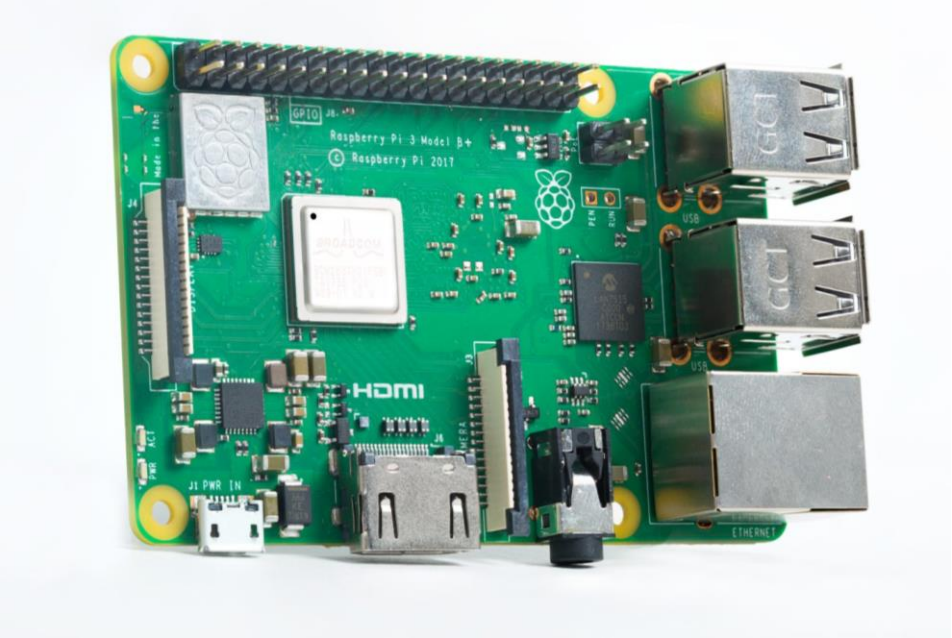

Figure 3. The Raspberry Pi 3 Model B+

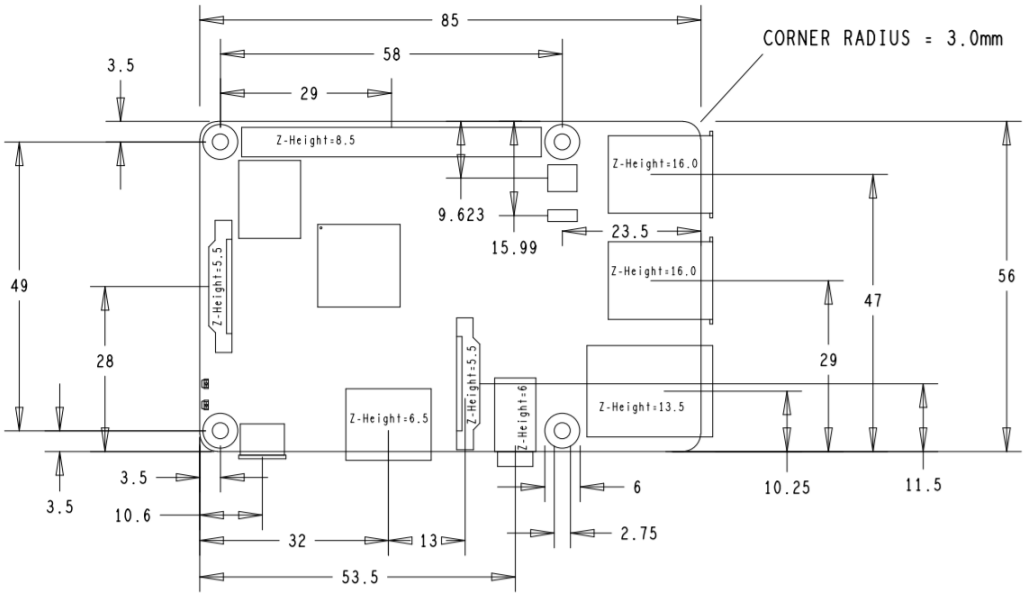

Figure 4. The Mechanical Drawing of the Raspberry Pi 3 Model B+

The Raspberry Pi is slower than a modern laptop or desktop. However, it is still a complete Linux computer and it can provide all the expected abilities that a computer implies, at a low-power consumption level with low cost. The Raspberry Pi is designed for the Linux operating system, and many Linux distributions now have a version optimized for the Raspberry Pi. Two of the most popular options are Raspbian, which is based on the Debian operating system, and Pidora, which is based on the Fedora operating system.

1.3 Research Aims and Objectives

The aim of this study is to develop the required software and hardware for lecturers and students in order to overcome difficulties faced during the lecture hours and to reduce the amount of time that lecturers spent on taking attendance of the students in the classrooms. By creating a card based attendance system that is designed with combining RFID and Raspberry Pi technologies, lecturers is be able to focus on the lessons and use the lecture time more effectively without losing time by traditional attendance systems.

Moreover, this project prevents frauds that can happen with respect to the traditional paper-based attendance system. Via using RFID cards, students scan their own cards at the RFID readers so that the number of students who attends the class clarified more effectively. Besides, a righteous and reliable attendance list is provided by the end of the session which is seen by both lecturers and students from the system.

The following are the core objectives of the research in achieving the aim mentioned above:

1. To make the attendance system more effective to improve reliability and time consumption which is currently found to be challenging at taking attendance.
2. To design an automated class attendance system in order to overcome the current difficulties.
3. To implement an automated class attendance system.
4. To develop a prototype web based attendance management software.

After the essential review process is finished, process of the work is identified and ready to be developed with the best practices and materials, all faculties are able to benefit from this study. The long-term approach to this study is to decrease the time wasted during attendance period, which the students sign for each other or sometimes forget to sign the list. It also eliminates the possibility of the lecturer forgetting the attendance list in the class. In addition, some lecturers prefer to call the names in the attendance list one by one to take the attendance which is also a time consuming process.

The following are the research questions of the study:

1. How to design and implement the RFID and Raspberry Pi for educational purposes?
2. How to apply RFID and Raspberry Pi in order to take and record students' attendance?

1.4 Significance of the Study

Traditional systems of taking attendance is time consuming process for lecturers. It is also prone to human errors especially when the number of students in a class is too high. In order to avoid cheating and human manipulation, the attendance of students can be recorded automatically by using a RFID technology. By this way, the attendance of the student is recorded automatically to the database in a faster and secure way than the traditional methods. By using automated attendance system, lecturers can have more time for teaching. Additionally, attendance reports of the students can be produced through a website so that users of the system both teachers and students are able to see the reports. For every student, a unique RFID card should be provided which can be recorded to the database of the system. When RFID reader reads the card, the system compares the unique number on the card to the one in the database.

The model of the RFID card that is used in this system is M1-S50 which has frequency of 13.56 MHz. The frequency of the card only supports 3cm of reading range. In addition, RFID cards with frequency of 13.56 megahertz costs around 0.50$ in the market, nowadays.

So, this study tries to design a system which is more time efficient, accurate and secure than the current attendance system at Eastern Mediterranean University.

1.5 Research Methodology

The research method of the study is Design Base Implementation Research (DBIR) which includes a literature review of RFID based attendance tracking systems and an implementation of RFID based automatic attendance tracking system for EMU. DBIR

is an increasingly being used as a research method by academic researchers and specialist when implementing, developing and testing academic interventions in authentic settings.

In DBIR, researchers investigate, ‘what works when, for whom, and under what conditions?’ (Fishman et al., 2013). In order to handle the persistent challenges of effectiveness, measurability and sustainability, the DBIR concentrates on understanding how educational innovations are implemented within local settings.

In addition, Raspberry Pi is used for the computational power of the system. This study includes the development of the related circuitry, software program and the database, so that it implements with respect to everything with the interconnection (Kazi et al., 2017). This study focuses on the tracking which saves time and with its user friendly interface, lecturers are able to save time during this process and they do not carry any manual attendance paper with them. Therefore, they can focus onto their work within the class and achieve their goals more easily and efficiently.

The rest of this report is organised as follows: Chapter 2 includes the literature review, Chapter 3 discusses the implementation strategy, Chapter 4 examines the developed class attendance system and finally Chapter 5 discusses the conclusion which also includes the implication of the study and recommendations for future work.

Chapter 2

LITERATURE REVIEW

There is a rapidly growing literature on the implementation of different technologies such as Raspberry Pi, RFID and face recognition into different environments; especially into the classroom environment, which are used as a part of their automation systems. A growing body of literature has examined the use of RFID technology for the automation of time and attendance in organisations especially in academic sector.

In their study, Qaiser and Khan (2006) claim that the RFID technology has a significant impact with being a promising technology. It is noted that not only one tag can be used but also multiple tags can be read at the same time from significant distances. The main scope of their study is to design a system to automate and enhance the time and attendance system in educational settings. The designed system is different than the other attendance systems. The time of the tag's enter/exit information is stored in the database as a log and then the total stay time of the students in the class is worked out. Next, this information is compared with the required stay in time necessary for each lesson. If the time matched with the require time for the lesson to be "Present", a students is marked as a present in the database. If it does not match with the required time, then the student is marked as an "Absent" in the database. Furthermore, the whole system does not require human interventions. The system can do are marking unauthorised entry, probation analysis, attendance weightage calculations, submission of warnings via emails, short message service (SMS) to

parents to keep them updated about their child's progress at the university and a website for the availability of the processed data for the users of the system.

In a different study investigated in this field, researchers proposed a system based on RFID over Ethernet which provides an automatic control of students' attendance in classrooms (Silva, Filipe, & Pereira, 2008). This study have similar patterns with the previous studies. The similarities are RFID readers and being able to see attendance report by web application. In contrast, there are several servers which communicate with each other to process the data from the readers to web application. One issue that needs to be discussed is whether using K300 proximity card reader is cost effective for all the schools or not. However, it should be mentioned that this reader is more efficient due to its features.

Moreover, another study on RFID based Attendance System was performed by Lim Sim and Mansor (2009), with an aim of taking attendance for students more secure, faster and easier than traditional methods in schools, colleges and universities. The system can also be used for working places to track the attendance of the workers. In their study, the overall system is controlled by a microcontroller and the program is written in modules which is combined together to form the final software system. When workers or students place their ID card on the reader, their attendance is taken immediately with the timestamp by using real time clock capability of the system as it makes attendance taken more accurate. The attendance taken is then stored inside database by connecting the system to the computer using RS232 or Universal Serial Bus (USB) port. Optionally, the attendance taken can be analysed by using a HyperTerminal software.

In a more recent research by Arulogun et al. (2013), a simple RFID attendance management system is designed. When RFID tag is read by RFID reader, the data is sent to the computer system. As an RFID reader, TR-R01-OEM reader board is used which performs the reading functions. Unlike other projects, a higher generation programming language is used to develop the graphical user interface (GUI), which is Microsoft Visual C#. The database of the system is Microsoft SQL. The authors noted that the study can be improved by adding a facial recognition capability as a biometric security and using active tags rather than passive low range frequency tags.

On the other hand, Nainan, Parekh and Shah (2013) described another attendance management system that utilises RFID technology. They used Microsoft Access as a database management system and Visual Basic 6.0 to develop the GUI. The Atmel is used as a microcontroller which is the main part of the system. The results of the study conclude that the total average time taken for recording the attendance of 60 students by using manual entry, Bar Code and RFID Technology is 600 seconds, 120 seconds and 12 seconds, respectively. Consequently, the time consumption taking attendance tracking by RFID is fifty times less than the time consumption in attendance tracking by manual entry; which proves that RFID saves lots of time for learning and highly improves the efficiency of the operation.

A similar project was developed by Bocheng and Shen (2013). The major difference of their study is the use of remote RFID reader which is a new mode of barrier-free channel that has RFID reader fixed inside. When a person carrying a card embedded RFID tag, passes the RFID reader channel, attendance data is received by the reader which then send to the data acquisition computer to be transmitted into database server via Ethernet.

A more recent study conducted by Sutar, Patil and Waghmare (2016) also uses microcontroller as a heart of the system to perform communication with all input and output devices such as RFID reader, liquid crystal display (LCD), buzzer and keypad. The study was conducted with the same aim of the previous researches which establishes a system where the attendance of the students taken automatically without any human interaction except the initial time setting. In their study, it is pointed out that system can be improved by adding voice announcement system, by sending the data online to the user and by implementing Global System for Mobile Communications (GSM) technology to the project which all should be considered for further researchers.

Similarly, Nguyen and Chew (2017) investigated an automated attendance management system to be used at different types of professional environments such as conferences, exhibitions, training and workshops. The system does not only provide information on attendance, but it also produces other reports such as a report on participants' least and most preferred activities and inflow and outflow of them during the events. The technology used in the system are RFID, mobile communication.

Zaman et al. (2017) designed an RFID based attendance system and used Arduino along with RFID as a microcontroller of their system. In their study, they replaced Barcode with RFID which has many advantages such as being faster, needing no direct of sight and storing encrypted data. For their system, an Arduino Mega, SD card Shield, RFID Reader, Wi-Fi Module, a LCD Display and also a Real Time Clock Module are interfaced. As a result, a successful and efficient automated attendance recording system is designed which believed that it will save time and manpower for any organization.

On the other hand, Rjeib et al. (2018) designed a system with the aim of registration, recording and managing of a student attendance using RFID tag which also provides student information service. For their system, an Arduino Uno R3 as a microprocessor, RC-522 RFID Reader, MySQL as a database, PHP as a programming language for GUI, WAMP server and an Ethernet shield are used as an interfaced.

Kazi et al. (2017) have also explored attendance tracking system using RFID technology and Raspberry Pi. They designed a system which processes three types of data; student information, class enrolment information and attendance reports. On the local server, the RDBMS of the system is stored which is located at the university's network. The application is developed on Pi Linux so that it can be accessible from a laptop or tablet without a special operating system required. The university gives access to all users including employees and students. Students are just need to enter the room with their RFID enabled student ID card in order to be present in the class. In addition, the attendance data can be seen by students via a web based application. On the other hand, lecturers do not need to do anything, they just need to click submit in order to run system properly. One of the advantages of the system is, it enables instructors to access class attendance data where they are able to view, update and submit a class attendance. The designed system also generate reports which can be exported to other inter organizational departments to improve accuracy.

Moreover, a study conducted by Yumang et al. (2017) is found to be successful in checking the attendance of the students without traditional methods such as pen and paper method with a design of an attendance checker by using RFID tags embedded in the school ID card of the students. The design also includes android application which is used to operate the entire system and Raspberry pi as the processing unit of

the system. Once Raspberry pi processes all the attendance information from the RFID reader, the attendance list is sent to database via Raspberry Pi. These records is not only seen through PC, but it can also be seen by smart phone devices by lecturers.

A similar study has been conducted to design and implement smart attendance system using Raspberry pi, which is conducted to be controlled from central unit in large factories or universities (Abdulsada, 2017). While central unit is formed using Arduino, GSM board and a computer, the central unit includes Raspberry pi, screen GUI, RFID and a transponder card. Instead of using web application as a GUI, desktop application is created via Python programing language. In addition, Wi-Fi network system is replaced by 2-way GSM modem so that file transfers between Raspberry pi and Arduino can be secured.

Recently, a similar study is conducted by Maramis and Rompas (2018) in order to develop an RFID based employee attendance management system to resolve problems of manual systems like the data accuracy and staff performance efficiency. In this study, the database system is integrated to the developed software of RFID attendance system aiming to store information of every single employee. The VB.Net programming language application is used along with a RDBMS to design two modules which are control unit and GUI of the system.

There has been a considerable interest in automated person identification method called face recognition using devices like Raspberry pi, RFID and Arduino in the last decade (Priya & Umasankar, 2016). Although there are many methods of taking attendance automatically such as RFID cards, biometric identifiers like fingerprint, palm print, face recognition, hand geometry and iris recognition are also widely

available in the literature. Face recognition has been receiving much attention as it is considered to be one of the most successful applications of image analysis and processing.

A study by Shah et al. (2015) proposed a cloud based biometric system architecture in order to make the remote enrolment more effective and economic by using Raspberry Pi. The hardware components of the system includes Raspberry Pi, Futronic FS88 Fingerprint scanner, PiCamera, Wifi Adapter, secure digital (SD) card and a power supply. Raspberry Pi is used for image capturing, encryption module is used for security and a cloud service is used for scalability and performance. Although the biometric system is low cost and portable due to the use of Raspberry Pi, the proposed system needs portability, scalability and low implementation cost for further research. In addition, the security of the system needs to be strengthen at the end-to-end encryption part.

In a study conducted by Sangewar, Waychol and Manekar (2015), Raspberry Pi and NFC (Near-Field Communication) is used with the aim of changing traditional attendance tracking systems to a more modern, fast and efficient attendance system. The components within the system include NFC/RFID, Raspberry Pi with camera module and an Open Source Computer Vision (Open CV). Due to the NFC tag and facial recognition, this authentication system is an extremely secure way of checking and managing student attendance.

A similar study was conducted with the same aim of maintaining attendance details of the students by using face detection (Priya & Umasankar, 2016). The hardware of the system are Raspberry pi, cameras, monitor, USB adaptors and SD card whereas the

software of the system are Wheezy Raspian and Open CV. A high definition camera connected above the white board captures an image of the whole class which then sends these images to the computer that is connected for processing. After that, the image is converted into gray scale which later a histogram is equalized and noise is removed. In real time video, face detection is applied to detect faces. If an image matches with in one in the database, attendance will be registered, but if it does not match, those images will be send to a mail ID and an SMS will also be send. The main limitation of the study is sometimes it is difficult to recognize faces if the picture is not accurate. Therefore, there is a need to improve the techniques that can recognize the faces in order to improve the efficiency and performance of this study (Priya & Umasankar, 2016).

In an another study, Gupta et al. (2016) designed a project to enrol and detect faces using camera connected to the ARM Cortex of Raspberry Pi board. The main aim of this project was to increase security and surveillance by investigating the feasibility of implementing the face detection and recognition techniques such as Haar detection and Principal Component Analysis (PCA) using Raspberry Pi. The system is believed that it can replace the use of passwords and RFI-Cards for accessing to systems and buildings. The hardware design of the system includes Raspberry Pi2 development kit, connector cables, LCD, direct current (DC) motor driver integrated circuit (IC), DC motor, power supply, universal serial bus (USB) keyboard, USB mouse and USV-B webcam. On the other hand, the software of the system includes python language which is used to code for hardware setup. Their designed project offers cost effective and easy to use system with high performance by using Raspberry Pi.

In a different study, Rajkumar and Prakash (2016) proposed an automated system by using Raspberry Pi in order to avoid losing productive time of the class with taking attendance manually. They used five modules called face detection, face preprocessing, face training, face recognition and attendance database. In order to take photos, USB webcam was used which is connected to core of the system, Raspberry Pi. To avoid environmental changes like facial expression, position and brightness, face preprocessing module was used in a constant position. MySQL is used as a RDBMS. However, their study was lack of security.

In a more recent research by Kiran and Kumar (2017), image processing based student attendance system was designed using Raspberry Pi with an aim of taking more accurate attendance at the universities. The architectural outline of their proposed system includes a power supply and a camera connected to the Raspberry Pi slot which captures images as input images. Images then send to the face detection and face recognition software in the Raspberry Pi. Next stage includes recording attendance, generating report and displaying it on an LCD or monitor. The system had no physical interactions and it completely eradicate the proxy attendance which are seen as an advantage of their proposed system.

Similar studies were also done in order to develop a comprehensive embedded class attendance management system using face recognition (Salim, Olanrewaju, & Balogun, 2018; Shrestha et al., 2018). Shrestha et al. (2018) focused on changing the manual system of attendance at the educational institutions to an automated attendance system by using face recognition. Within the system, the dataset of known faces of the students are stored so that when a real time video of the classroom is captured, the faces are detected and images are recognized by using Local Binary Patterns of

Histogram (LBPH) algorithm. By this way, the successfully recognized image of the students made them present in the system. In contrast, the face is detected as intruder if it is an unknown face which activates the siren as an action for the security.

Correspondingly, Salim et al. (2018) developed and implemented a comprehensive embedded class attendance management system using face recognition. It is not only the face recognition that is proposed in their study but also the door is also controlled to whether the student is allowed to access the class or not. The Local Binary Patterns (LBPs) algorithm was implemented to handle the face recognition. One of the drawbacks of the LBPs algorithm is its light sensitivity which makes face recognition difficult. It is noted that the algorithms of face recognition is very heavy so instead of using Raspberry Pi, ODROID-XU4 Portable Computer, which has more advanced hardware than Raspberry Pi 3 is recommended in their study. The system was also accessible for the authenticated web clients through an online web server. If a lecturer wants to check the attendance report, it is possible to access the user interface on the server through any Internet connected computer.

Giving a general panorama of past-to-present literature, it is evident that there is a transition from the traditional signature on a paper sheet to an automated attendance system. For time and management efficiency, there is a need of an alternative attendance system to be used in the classroom environment. In the literature, numerous studies were proposed with this aim by using different technologies such as raspberry Pi, RFID, Arduino UNO, finger print and face recognition. All have different advantages and disadvantages when compared with each other. The use of biometric systems like finger print and face recognition technologies need high budget when compared with the other technologies despite they can provide more secure system for

the institutions. Therefore, while designing an automated attendance system for an institution, the requirements of each methods should be considered carefully.

Chapter 3

IMPLEMENTATION STRATEGY

RFID is designed for short-range product identification. In order to store the student attendance via MySQL database, RFID cards should be given to the students instead of their student cards and let them scan their cards into the system. In addition, during the class, lecturer can tell the students to scan their cards to the reader so that there is no double scan or fake scan within class. In order to see if the scan is successful or not, there is an LCD monitor attached to the Raspberry Pi. Development of the application is managed by Python programming language. Once the Raspberry Pi records the attendance into the MySQL database, both the student and the lecturer are able to see the attendance taken through a website.

For the website that is developed using server side scripting, data transfers are made easily. Implementation can be discussed in six steps. First step is to run the system and to see “Scan your Card” message displayed on the LCD monitor which means the system is ready to read the cards. In the second step, the card is scanned through the RFID reader by a student. In the third step, Raspberry Pi compares the card’s data with the MySQL database. If the data is matched, “Attendance Taken” message is shown on the LCD monitor for 6 seconds as the fourth step. Data is sent to the server by the Raspberry Pi as the next step. The final step is to let the students and lecturers check the website to see the attendance reports. The block diagram of the proposed system is as shown in Figure 5.

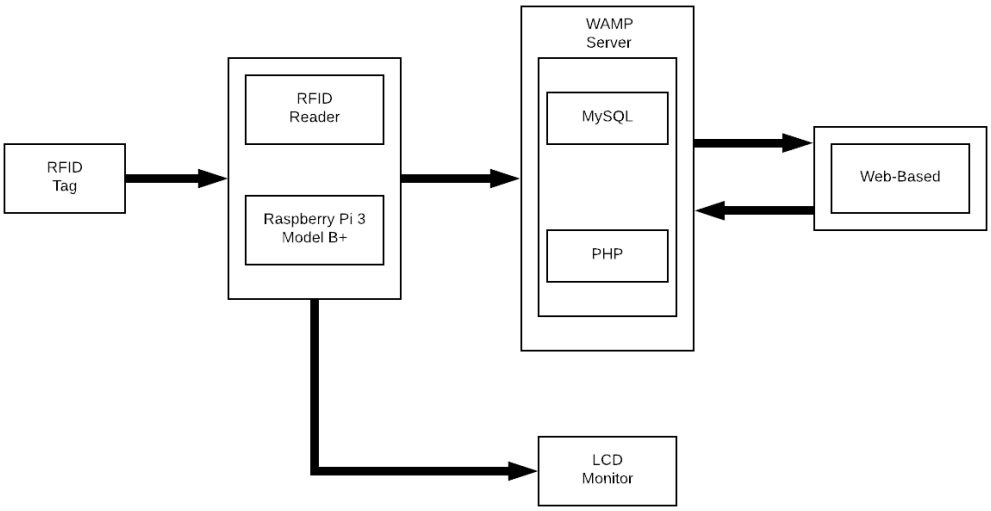

Figure 5. Block Diagram of the Proposed System

The system consists several phases to be accomplished in order to take an attendance of the student. The first phases is reading which requires the interaction of students and the system. After this process is completed successfully, the microcontroller's phases start. The data sent to the Raspberry Pi start to be processed within the Python application which runs already in the Raspberry Pi. Then, the Raspberry Pi sends a request to the server in order to receive a response. Next, the submitted data is processed by the RDBMS and if data meets the requirements, it is inserted into the database tables.

Additionally, as it is mentioned before, all the users are able to see the attendance details according to their privileges on the system. There are three different privileges for three different type of users on the system: student, teacher and administrator. The details of the attendance records and basic student information including personal details can be seen on the website depending on the privileges of the users. All these processes can been seen in the process diagram shown in Figure 6.

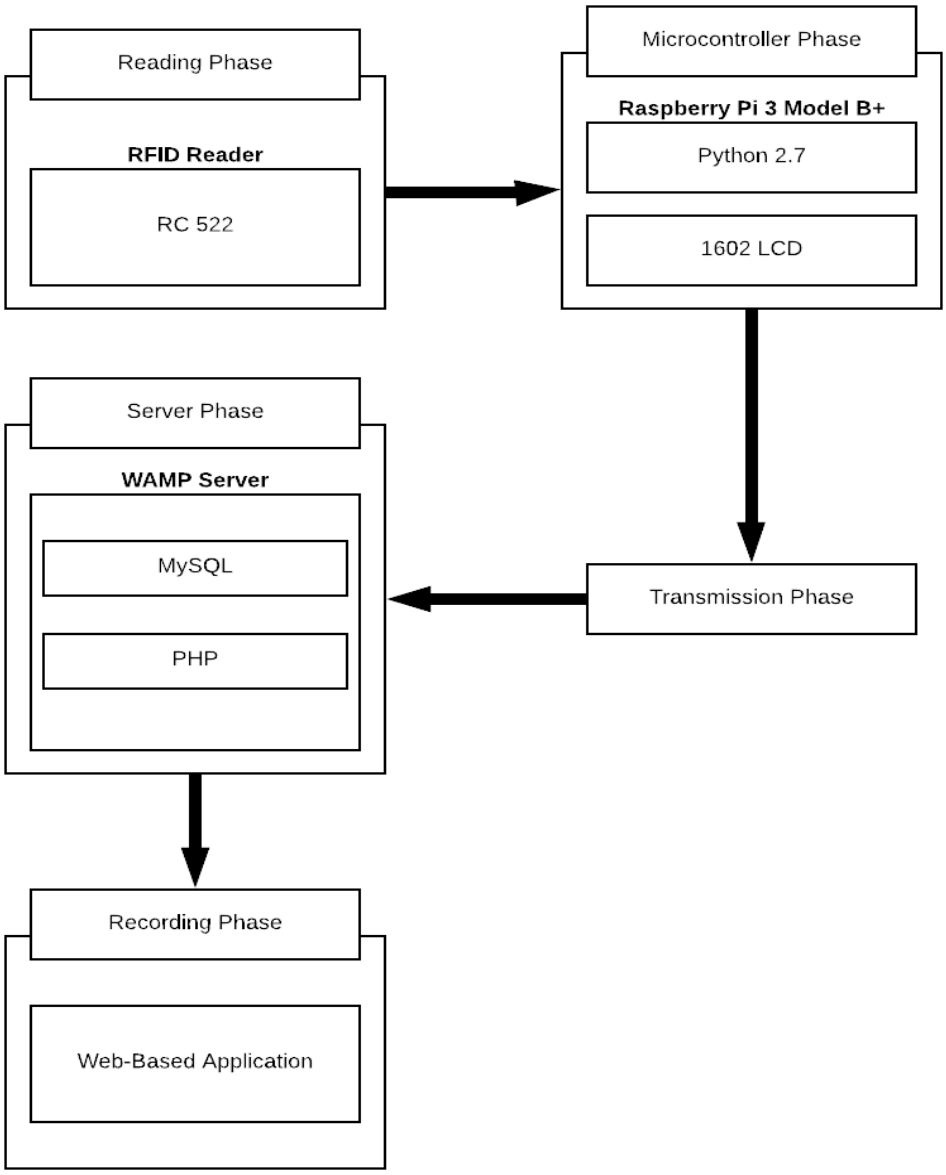

Figure 6. Process Diagram of the Proposed System

3.1 Microprocessor of the System

Raspberry Pi is a microprocessor which is developed by Raspberry Pi Foundation to increase the basic computer science in the schools and universities on developing countries. This system works with a Linux operating system which is known as Raspbian. First release of Raspberry Pi was in 2012 named as Raspberry Pi 1 Model B. Until 2018, there was few more releases made by Raspberry Pi Foundation. Moreover, Raspberry Pi 3 Model A+ had its own day view on November 2018. On the

other hand, Raspberry Pi 3 Model B+ has more advances and powerful features compared to it with its former counterpart and the newly released Model A+.

Raspberry Pi 3 Model B+ is the first model which has a 1.4 gigahertz CPU that makes Model B+ more powerful than it to the previous models. However, Model A+ has the same specification that Model B+ has. Nonetheless, Model A+ has smaller Random Access Memory (RAM) and Ethernet is not available on it. Raspberry Pi 3 Model B+ is the core of this project. It has additional more advanced features with combination of Adeept RFID Starter Kit. The proposed class attendance system is performed by the combination of Raspberry Pi, RFID and LCD display that can be seen in the figure below.

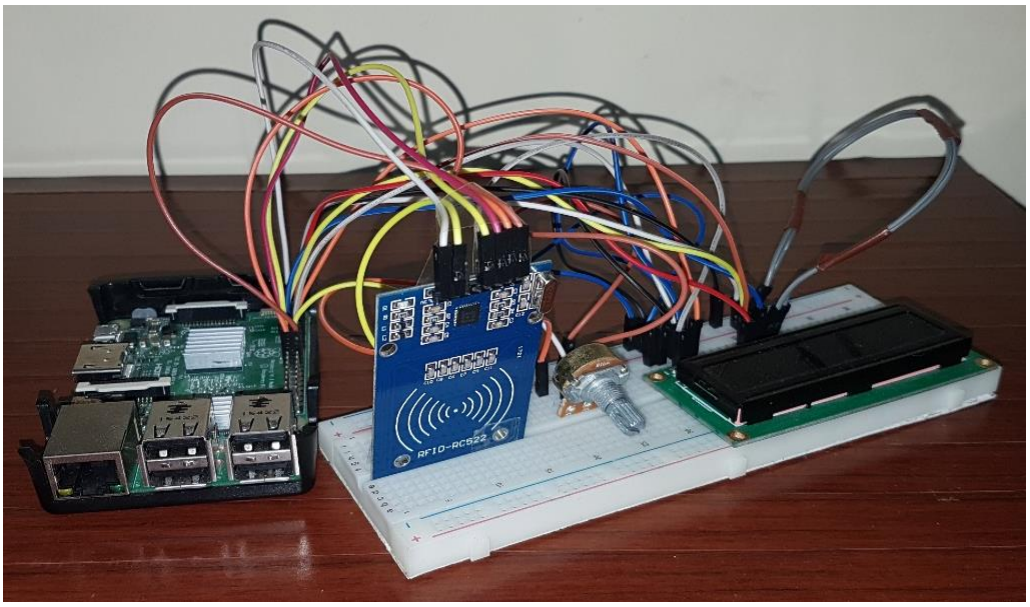

Figure 7. The Developed Circuitry for the Proposed System

With the combination of these components, a student who has a class can read his/her RFID card to the RFID reader in order to register their attendance to the system. The received data is searched within the database through Raspberry Pi and a response message is seen on LCD display which is attached onto breadboard. If the student is

found in the database and he/she has an appropriate criteria, attendance of the student is added to the database and a success message can be seen through LCD display. In contrast, if student do not have the suitable criteria, a written message of the reason why their attendance is not inserted to the database is seen through LCD display. Along with these response messages, students are able to know what kind of criteria they are missing such as wrong time of the lesson or incorrect classroom.

In order realize the proposed system, several components are needed. First component is the RC522 RFID Module, which makes the reading and transferring data to the Raspberry Pi. The RC522 RFID Module has 13.56 MHz operation frequency and it is working with 3.3volt that is compatible with Raspberry Pi's voltage. The LCD is the second component. The model of the LCD is LCD1602, which works with 5 volt and 3.3 volt along with the Potentiometer (10kΩ) that can adjust the contrast of the LCD. The rest of the components are the jumper wires and breadboard which is the bridge between Raspberry Pi, RFID and LCD monitor.

A more detailed drawing showing the connections between the components can be seen in Figure 8.

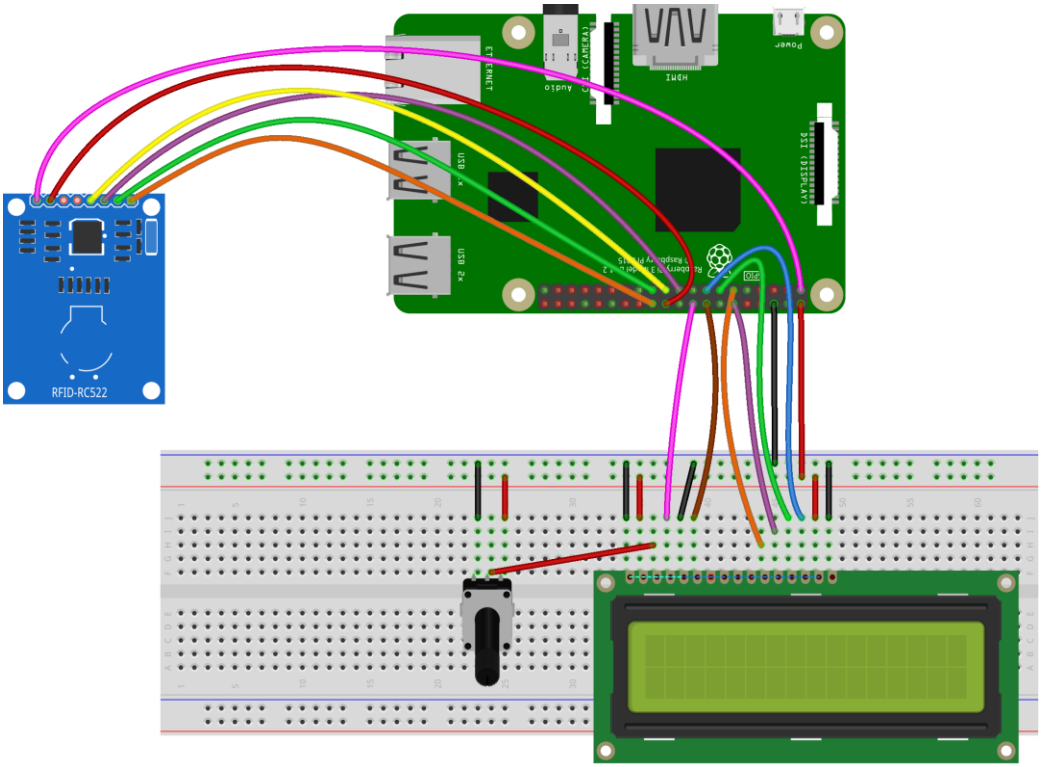

Figure 8. Connections between the Components

3.2 Programming the Raspberry Pi

Python is an object oriented programming language which is developed by Guido van Rossum. It was released to the market on 1991. In the early stages, Python was not preferred by the community. However, Python is getting in the top five spot within the programming languages nowadays (TIOBE, 2018). Although the recent release of this programming language is Python 3.5, it has bugs. One of the most obvious feature is changing syntax with the updates on Python 3.5 where people face with errors so often. On the other hand, Python 2.7.15 stop publishing its updates recently on May 2018, which makes Python 2.7.15 more stable and easy to access its component on the Internet. In recent years, Python was used to create home automation systems by Fabian Affolter and Charles Garwood. Affolter and Garwood developed this application as an open source one and are supported by big companies in the world (Home Assistant, 2018). Furthermore, Python is capable of developing an artificial intelligence (AI) which makes it a strong and a dynamic language. In order to use its

different features, Python needs additional libraries like Tensor Flow, Keras and Scikit-learn.

Python 2.7 is used in this study with additional libraries so that additional features can be used while programming. RPI.GPIO library programs Raspberry Pi`s pins so that they can work. When program starts running, some of the input and output pins on Raspberry Pi are assigned to a variable. Another library included within the application is MFRC522. This library contains several functions so that the RFID reading process can be performed. Adafruit_CharLCD library have same specifications similar to MFRC522 library. This library include functions that help the system to print some messages to the LCD monitor which is attached to the breadboard. These two libraries attached to the system. The source code of the main functions written with Python is in Appendix A.

On the other hand, rest of the libraries are the proper Python libraries and these libraries can work without Raspberry Pi. MySQLdb library helps Python to communicate with MySQL database that is on the server. The time library is used for creating the delays in the project. When card is read by the MFRC 522 library, time library gives 6 seconds delay to the system before it can read the next one. This helps students' card to be read by the system without any confliction in the RDBMS. Last library used on the system is datetime. It is used to send system timestamp, which is the time that the card is read by the system, to the database so that attendance registration can be performed.

The combination of libraries and their proper use creates the whole system. In addition, a separate program is developed which helps the system to keep the attendance properly in case of a connection loss with the RDBMS. If connection loss occurs

during the process of attendance registration, the system still keeps the attendance of the students in a log file which then can be used to commit to the database (Lutz, 2013).

The programming language choice of the users is often influenced by the possible potential problems that they might rise in their projects or the concept of their projects. When Python is compared with the other programming languages, some features become prominent. The simplicity is one of its features. For example, C++ or Java are the well know programming languages today, but developers spent very long time while working with these languages, which makes them time consuming. On the other hand, Python can be used to implement the same application 5-10 times faster than the other two programming languages (Python Foundation, 2018). This makes Python mare attractive as the implementation language. The JavaScript language is the one of the equitant language of Python. Both of them work with simple functions and variables which make these two programming languages fluent and flexible. Moreover, Python can engage with much larger programs and can be coded better with an object oriented programming style. Figure 9 shows IEEE Spectrum Ranking of the Python language around the world (IEEE Spectrum, 2018).

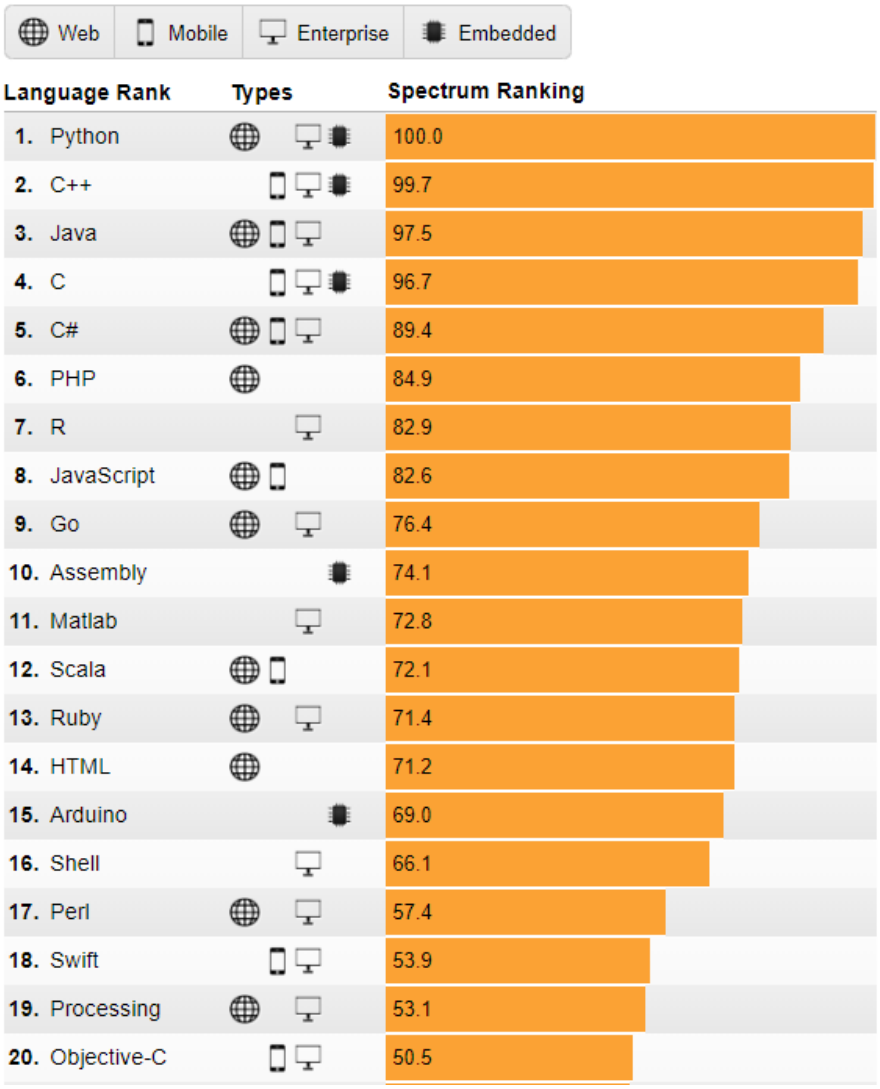

Figure 9. IEEE Spectrum Ranking

As seen in the above figure, Python programming language leads the Web, Enterprise and Embedded language types in 2018. Additionally, it is the most popular programming language within the rankings of IEEE in 2018, it becomes more popular than many other well-known languages such as C#, C++ and Java. This ranking source arrives from 9 different sources. Search and index information retrieved from Google search and trends, number of tweets retrieved from Twitter, repositories are calculated within GitHub, number of questions posted retrieved from Stack Overflow, “upvote” and “downvote” are retrieved from Reddit, comment and links are calculated within

Hacker News, job demands are calculated on CareerBuilder and the data are gathered from Dice.

3.3 MySQL as the RDBMS

One of the most well-known relational database management system is called MySQL. It is invented by Michael Widenius and David Axmark. In addition, `My` part of the MySQL takes its name from Widenius`s daughter. Recent release of MySQL was on April 2018 with a version of 8.0 which was the first release of the company after three years. Current version of MySQL is best known by its speed when it is compared with the last released version. It is two times faster. In addition, the current version of MySQL can record with up to 1.8 Million Queries/Second (MySQL, 2018). These specialties make MySQL 8.0 more reliable in today's technology. Furthermore, one of the main advantage of the MySQL is being an open source. By this feature, other programs are able to interact with the system. In order to maintain data and keep the files safe and secure, using a RDBMS in the system is the only option. Secure design RDBMS helps the system to protect itself from outside attacks. In addition, shortened data entry, cache, and retrieval costs are another advantages of using RDBMS on creating these complex systems.

This proposed system consists of eleven different database tables named as attendance, classes, course, course_hours, days, department, student_course, time_table, users, user_photo and user_type. The Entity Relationship (ER) diagrams are the key part of designing a database. Before creating a database every student or professional should create the ER diagrams to have an idea about their projects. The ER diagram of the database used in this study is as shown in Figure 10.

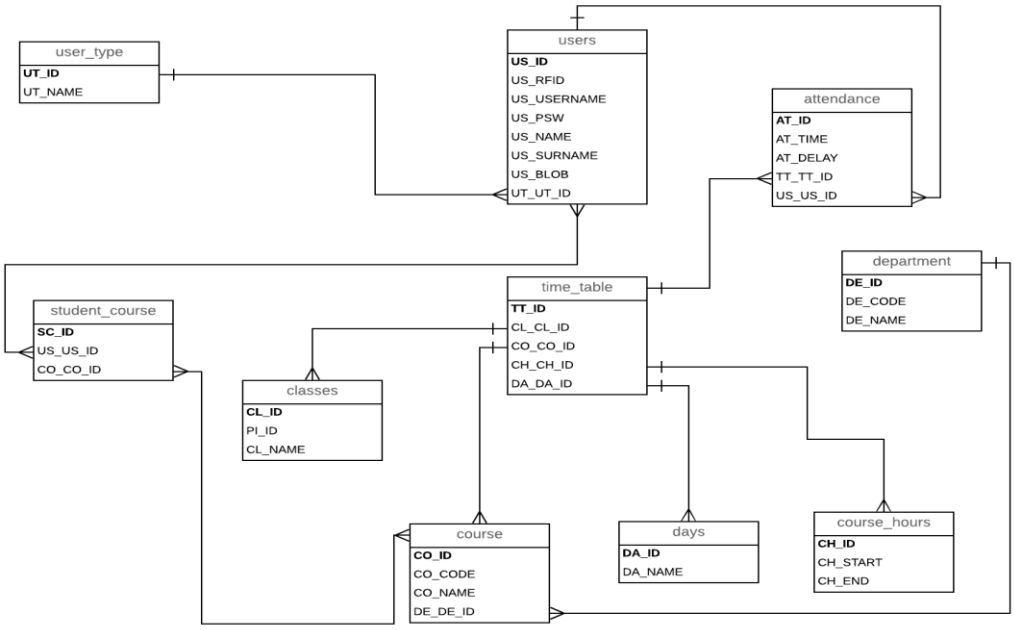

Figure 10. ER Diagram of the Proposed System

As seen in the graph, every table consist of several fields where the data is kept. The users table is the one where the information of the student, teacher and admin are kept. In these fields, every user had a unique US_ID and US_RFID which represent the real time university ID and RFID of the students or teachers. In the next fields, there are usernames, names, surnames and passwords. The password field should be kept securely so it is hashed using MD5 algorithm within MySQL database. The last two fields are the foreign keys which data is transferred from different tables.

Two tables, which have a connection with the users table, are user_photo and user_type. Both of these tables contain their own unique ID and each represent a field. The students, teachers and admins groups are represented in user_type table as UT_NAME. In addition, every user have their own unique photos and this field is named as UP_BLOB. The name blob means binary large object which means that the database can contain image files as a hash algorithm in a different way.

Another table in the database is student_courses which contains a unique ID and two foreign key fields. One of the field name is US_US_ID. This field takes users ID so that it can contain required fields from users table. The other field is CO_CO_ID which is for the courses table that takes ID of the course. These fields together represent which user takes which course.

Similarly, the courses table contains a unique ID field. Despite of other tables, course table has CO_CODE and CO_NAME field. CO_CODE field represents course code and CO_NAME represents the name of the course. In addition, the course table has a foreign key connection with the department table which takes the needed fields from the department table. The department table also has the same features like course table except foreign key. The only difference of this table is instead of course name and course code, it contains department name and department code as their fields.

Classes, days and course_hours tables have similar features. Firstly, classes table has CL_NAME fields which is the name of the class and two unique fields which are ID and PI_ID. The reason of why classes table has PI_ID as unique field is because every Raspberry Pi has their own unique ID and these ID need to be match within the code in order to document attendance registrations or any problems occurred within the system. The second table is the days table which contain the day of the week. Finally, course_hours table consists the eight lesson periods in a day. Each lesson period have CH_START and CH_END. CH_START is the starting hour of the period and CH_END is the ending of that period.

The table of the time_table is generated by the combination of course, classes, days and course_hours tables. The time_table has its own unique ID and the rest of the fields are foreign keys. In addition, time_table is the main table on this ER Diagram.

Lastly, every attendance registration inserted to the attendance table by using Raspberry Pi. This table contains its own unique ID, AT_TIME, AT_DELAY and two foreign keys which keep time_table data and user table data. AT_TIME is the attendance registration time of the student and AT_DELAY represents calculation of the data about how long students are late to their classes after their attendances has taken.

3.4 Server Side Programming with PHP, JavaScript and HTML

Last phase of the system is the development of the graphical user interface (GUI). This phase is the real interaction between end user and the system. The system is designed as an online web-based application. Student is able to see their data within this system. In addition, teachers can get their class reports within this web application. This system is developed using PHP, JavaScript and HTML programming languages and main interface of this system is designed by SB Admin 2.0 which is an open source web-application to create user interfaces.

The PHP, JavaScript and HTML are the three programming languages which are well known languages while developing a web-based application. The critics sometimes compare PHP and JavaScript, but the purpose of these languages are different. The PHP is a server-side scripting programming language. On the other hand, JavaScript is a client side. The PHP programming language is developed by Rasmus Lerdorf in 1994. Previously, Personal Home Page’s abbreviation is PHP but nowadays it is

known as Hypertext Preprocessor. The main idea of PHP is getting information through a server. For this system, Wampserver is used as a server that contains the database. With the help of Wampserver, the developed web-based application can take the data through database. Wampserver is a combination of Apache web server with a version of 2.4.35, OpenSSL for SSL support, MySQL database with a version of 5.7.23 and PHP programming language with a version of 7.2.10 which is created by Romain Bourdon

Another programing language used in this system is JavaScript. This programming language is known as high-level, interpreted programming language. It is developed by Brendan Eich in 1995. JavaScript is one of the three core languages while developing a web-based application. With the help of JavaScript, web-based applications can be more interactive for the users. Syntax and lots of other features of JavaScript has some similarities with Python programming language which is used within this project to develop the Raspberry Pi side of the project.

World Wide Web Consortium (W3C) and Web Hypertext Application Technology Working Group (WHATWG) have developed HTML in 1993. After 25 years, the last version of this language was released which is named as HTML5. HTML is the body of the web based application

Chapter 4

DEVELOPED CLASS ATTENDANCE SYSTEM

4.1 RFID-Based Attendance System

The RFID-based attendance system's work flow is based on the queries within the python code. The approval of attendance is managed by the help of queries in the system. For the first approval, student needs to be registered to the university. This requires specific RFID number to exist within the database when an RFID card is read to the system. In contrast, "Student is not registered" error message is displayed on the LCD when RFID number is not registered within the database. "Student Is Not Registered" error message can be seen in the Figure 11.

Figure 11. "Student Is Not Registered" Message on LCD

The next one is the class query which is the beginning step for the completion of time_table query. As it is mentioned before, time_table query consists of foreign keys

and every field of time_table query filled by the help of other tables. The class table's query help to identify the class on the system. Then, this data is inserted to a variable. In addition, every system contains the class ID as a static variable within them so it is impossible for the system to return an error message. If class is defined or reader is registered correctly.

The next query is for the day of the week. To identify the right day, the datetime library's function is used within the system. The query can be activated with the help of this function so that specific data can be stored to a variable. On the other hand, if there is not a lecture on that day to handle these situations, "No Lecture On This Day" error message is shown on the LCD to the students as shown in Figure 12.

Figure 12. "No Lecture On This Day" Message on LCD

Moreover, the period of the lesson is identified by course_hours table. When student's card is read by the system. This query uses "sysdate" function to identify the current time. However, the system can show "No Lecture At This Time" error message, if student's card read by the system at the time that is not within the periods in the

database. The error message can be seen in the Figure 13. Similarly to other queries, course_hours data are also inserted to a variable within the system.

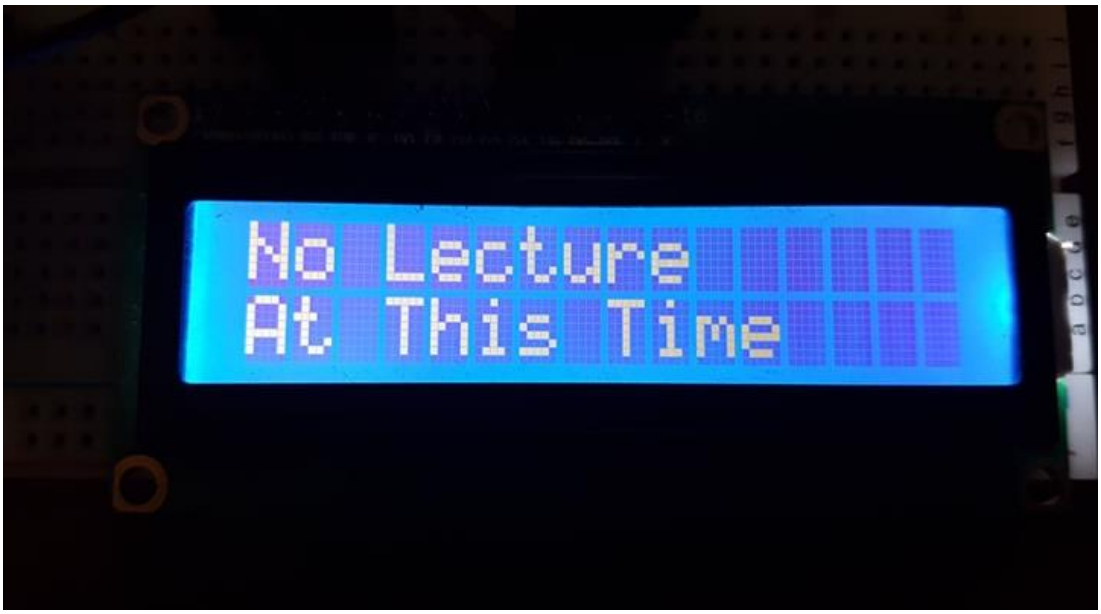

Figure 13. “No Lecture At This Time” Message on LCD

All the variables are used to prepare the time_table query in order to finalize the query. As it is mentioned before, the time_table fields consist the data from four tables which are classified as classes, course, course_hours and days. Due to the sufficient data in tables to complete the time_table query, this query does not need any data from course table. Moreover, if one of these fields are not matched within the fields in the database tables “Time Table Not Correct” error message is shown on the LCD as shown in Figure 14.

Figure 14. “Time Table Not Correct” Message on LCD

The next query is for finding out if student is registered to the course. To run this query, data are needed from the time_table and users tables. The user ID from users table and course ID from time_table is used to complete this query. This query helps to identify whether the student is registered to that specific course. If a student is not registered to that specific course, an error message is displayed on the LCD which is written as “Not Registered To This Course”. This error message can be seen in the Figure 15.

Figure 15. “Not Registered To This Course” Message on LCD

The last query on the system is the insert query. This query is managed by the data from time_table and users tables. Current time, delay time, ID of the user and time_table are added to the database table with this query. In addition, “Attendance

Taken" message can be seen at the end of this process. With this step, the cycle of the system is completed. The success message can be seen in Figure 16.

Figure 16. "Attendance Taken" Message on LCD

In addition, it is also possible to consider some scenarios where the developed system may fail to take the attendance of a student. These include power loss, RFID card failure, forgotten RFID card or connection loss between the components of the system. In order to prevent these problems, some backup plans can be proposed.

For the power loss problem, lecturer may need to register the students with traditional paper based system and registering them to the system afterwards. RFID cards can be damaged. For example card may have an electromagnetic error. Until the card problem solved, the attendance of the student can be taken using traditional paper based system and registered to the system afterwards. Furthermore, students might forget to bring their RFID cards so that they will not able to scan their card to be seen as present in the lesson. Similarly, lecturers can register the student to the systems after taking the attendance using traditional paper based system. Finally, connection loss problem can

be prevented by keeping the student ID, student name and system date as a log file. So, the lecturer can register the students by using those log files afterwards.

4.2 Developed Web-Based Application

The main part of the web-based application is developed using HTML. The HTML forms the structure of the web based application of the system. By using HTML, every web page of the system is designed. In addition, PHP and JavaScript are the programming languages which supports web-based application so that it become more dynamic.

The designed web-based application is assumed to use EMU's RDBMS so that web-based application have some limited features. Student and teacher user types are able to interact with their courses, time tables and the report of the students only. These user types are not able to manipulate the data which is registered to the system. In addition, teacher can search students within the student report page by their user id, name and course code. Report page is always sorted by the user id. Furthermore, this page can be downloaded as an excel file by the teacher. It also contains pagination which limits the data table to store twenty data and the next data can be seen on the second page of the report page.

On the other hand, admin user can see the users, defined courses and all of the attendance reports of the students. Similarly, admin user cannot change any information on the system. Furthermore, admin can make search according to user Id, name, surname, username, RFID number and user type within the pages. Additionally, admin can download the excel file within their report page.

Users are able to login to the system with their usernames and passwords. Additionally, the proposed system uses Google's ReCAPTCHA 2 in order to prevent login attacks to the system. This type of attacks are called brute force attacks. ReCAPTCHA is an application of google which shows you several pictures to match or select so that there will be a gap between every login attempt. After the users successful login to the system, they are redirected to the Dashboard page. After login, dashboard is the main page of the system. On the other hand, every user group has their own pages to use in the system except from dashboard page. For example, student user type is able to access to their student details within Dashboard page. These information include their student numbers, RFID numbers, names and user type. However, these information are same on other user types as well. On the left hand side of the application, the navigation bar can be seen which helps the user to navigate to the other pages and every user type have a different navigation bar from each other. Students are able to access their courses and their timetable within this bar. Within these pages, students can also see how many courses in total they are registered and the time of their classes. Moreover, students can see their reports. In these reports, students can access to their attendance details. Dashboard page, which belongs to the all type of groups, can be seen in the Figure 17.

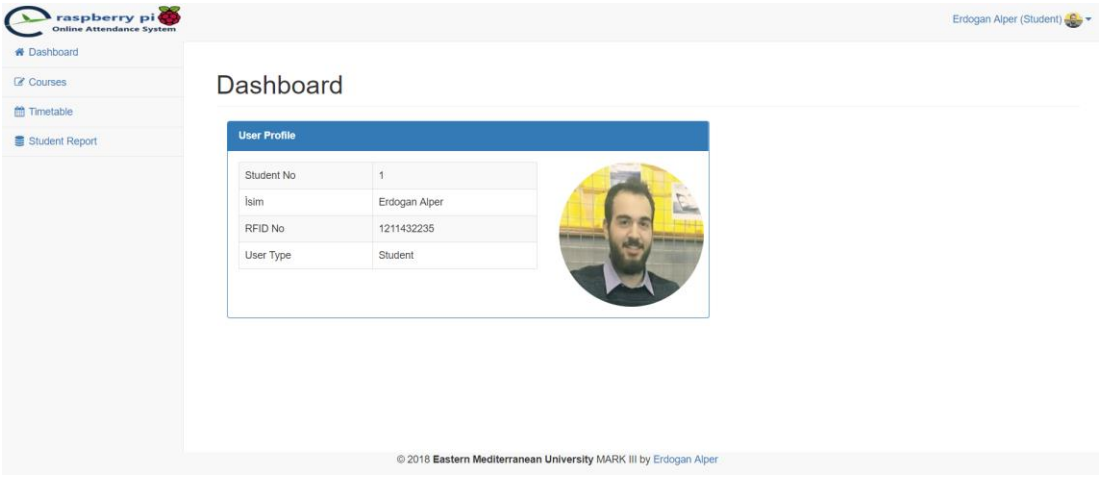

Figure 17. Dashboard Page of the Web-Based Application

The teachers have same navigation bar like the students, but some features of the pages are different than students' pages. The teachers can access the number of courses they are currently teaching and the periods of the courses. The teachers can also access to the reports of the students which they are teaching. In the teacher report page, teachers are able to see all their students or give a specific filter by searching their names or student numbers with the search bar. Finally, the teachers can download the report in Excel (.xlsx) format whenever they need. The teacher report page, which belongs to the teachers groups only, can be seen in Figure 18.

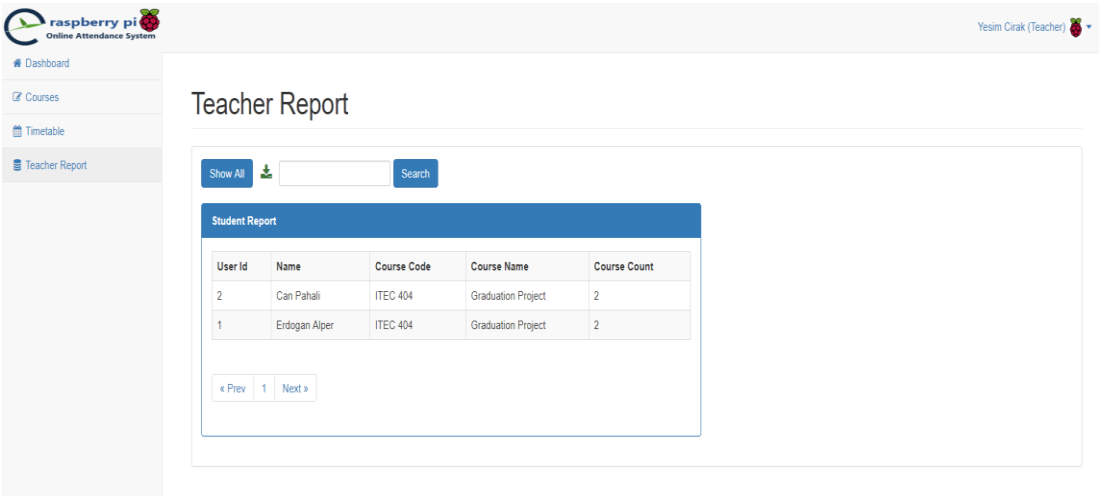

Figure 18. Teacher Report Page of the Web-Based Application

Admin type is the last user type on the system which have all the privileges in the system. These users have different navigation bar from other two groups. They have access to all the user's ID, names, RFID numbers, usernames and types. On the other hand, admin cannot interfere any of the data within the system, because this system assumed to have an external connection with a university database. Therefore, deleting information can cause some problems within the database of the university. This is the main reason why every user can only have a read only access on the system. Similar to the teacher reports, admin can download reports and filter every student using admin report page. Moreover, every user group has own logout link on the top right of every page so the users can logout from the system. The page called "All Users" belongs to the admin group only which can be seen in Figure 19.

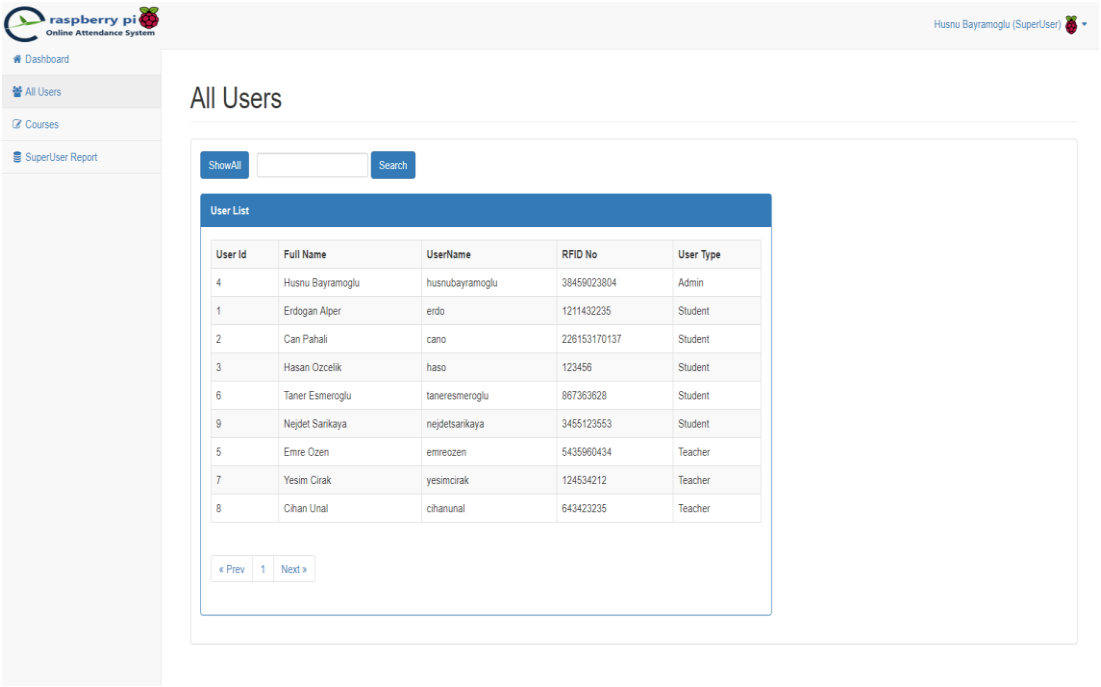

Figure 19. Admin's All Users Page of the Web-Based Application

4.3 Cost Effectiveness of the System

There are several components that are implemented together to build the designed system. These components include Raspberry Pi 3 Model B+, RFID reader RC522,

LCD 16x2, Breadboard, Potentiometer and 22 piece of jumping wire. In the market, the cost of the Raspberry Pi 3 Model B+ is around 35.00$. Moreover, the cost of the RFID reader RC522, LCD 16x2, Breadboard, Potentiometer and 22 piece of jumping wire are approximately 2.08$, 6.49$, 8.99$, 6.19$ and 1.28$, respectively. Therefore, the designed system costs approximately 60$ in total, for one classroom. The cost of the system for the university should be calculated based on the cost that is assumed for one classroom multiplied by the number of classrooms. For example, if the designed system will be implemented for 30 classrooms at the university, the total cost of the system will be around 1,800$ totally. In addition, the designed system requires a RFID Card for each student which costs 0.50$. Even for students, the required cost of the RFID card is affordable.

4.4 Test Results

"Student Is Not Registered" Message

Test Case #	RFID Card	Expected Outcome	Reason	Actual Result
1.1	1211432235	Valid	RFID card number is match one of the numbers within database	
1.2	123456	Invalid	RFID card number is not match the numbers within database	"Student Is Not Registered"

"No Lecture on This Day" Message

Test Case #	RFID Card	Expected Outcome	Reason	Actual Result
2.1	1211432235	Valid	Student have lesson today	
2.2	123456	Invalid	Student does not have any lesson today	"No Lecture on This Day"

“No Lecture at This Time” Message

Test Case #	RFID Card	Expected Outcome	Reason	Actual Result
3.1	1211432235	Valid	Student have lesson within that period	
3.2	123456	Invalid	Student does not have any lesson within that period	“No Lecture At This Time”

“Time Table Not Correct” Message

Test Case #	RFID Card	Expected Outcome	Reason	Actual Result
4.1	1211432235	Valid	Class, course hour or date is correct for that student	
4.2	123456	Invalid	Class, course hour or date is not correct for that student	“Time Table Not Correct”

“Not Registered To This Course” Message

Test Case #	RFID Card	Expected Outcome	Reason	Actual Result
5.1	1211432235	Valid	Student is registered to this course	
5.2	123456	Invalid	Student is not registered to this course	“Not Registered To This Course”

“Attendance Taken” Message

Test Case #	RFID Card	Expected Outcome	Reason	Actual Result
6.1	1211432235	Valid	Student Attendance Registered to the database	“Attendance Taken”

Chapter 5

CONCLUSION

The main aim of this project is to develop an automated attendance tracking system using RFID technology in order to provide more accurate results than the traditional tracking systems that are based on pen and paper or calling out names of the students. The proposed system works with Raspberry Pi with the combination of RFID reader and LCD components. The Raspberry Pi, RFID reader and LCD display are connected to each other using jumping wires on a breadboard. The Python programming language is used within the Raspberry Pi. When a card is read to the system, the code written in Raspberry Pi help to run MySQL queries within the code. Then, the transfer between Raspberry Pi and the server take place where the database is searched by the MySQL queries within the code and return a message either as a success message or error message. The web based application is available to students, teachers and administrators, where they can track the attendance reports, timetable or user's information.

5.1 Implications of the Study

The traditional attendance system has some disadvantages. Lecturers spent significant amount of time from their teaching time to take the attendance of the students which makes the current system time consuming. In addition, there are many students coming from different countries that make it harder for the lecturer to call out names with the right pronunciation. When there is also lots of students in the classroom, the efficiency of the traditional system becomes more questionable. Moreover, extra paper works and

additional lockers for saving data are needed. These are the main problems which is found within the current attendance system at EMU.

The proposed system provides more effective and easy way of taking attendance. The time spent on taking attendance manually by the lecturers are reduced with the designed automated attendance system. Instead of spending time on taking attendance, this time can be used as the teaching time. Moreover, lecturers do not need to call out names one by one due to the automated attendance system. Students just need to show their RFID cards to the reader in order to sign themselves as present within the system. In terms of extra paper works and additional lockers for saving data, the study offers easiness to the teachers as they are able to see the attendance of the system online with the web based application.

5.2 Future Work

The reader should bear in mind that the system is designed to be used in EMU by examining the vulnerability of the current attendance system. The designed system might be a milestone for the more advanced automated attendance tracking systems to be implemented at the EMU. For example, biometric identifiers like fingerprint and face recognition might provide more secure and reliable system for the future studies. Although automated attendance system is efficient and secure, students can give their RFID cards to other students which can make the system vulnerable to fake attendance. Fake attendance can be avoided with the additional security components that might be attached to the system such as 4*4 Matrix Keyboard. This keyboard is used to type a pin code. When a card of the student is read by the reader, the four digit pin code might also be requested by the system. Currently, the card provided by the bank is also used

as a student ID of students at EMU. Considering this fact, student ID card pins can be entered as a pin code for an additional security.

In addition, RFID reader which is implemented in the system has approximately 3cm of range which might prevent the card to be read from far distance by the reader. The range of the reader might be longer than 3cm which might be improved by changing the safe reading range so that the card can be detected from far distance by the reader.

REFERENCES

Abdulsada, H.F. (2017). Design and Implementation of Smart Attendance System Based on Raspberry pi. *Journal of Babylon University/Engineering Sciences*. 5 (25), 1610-1618.

Arulogun, O.T., Olatunbosun, A., Fakolujo, O.A., & Olaniyi, O.M. (2013). RFID-Based Students Attendance Management System. *International Journal of Science & Engineering Research*. 4(2), 1-9.

Bocheng, Z., & Shen, Y. (2013). Remote Attendance System Based on RFID. *Applied Mechanics and Materials*. 380-384, 4080-4083.

Bowen, E., Price, T., Lloyd, S., & Thomas, S. (2005). Improving the quantity and quality of attendance data to enhance student retention. *Journal of further and Higher Education*, 29(4), 375-385.

Choi, M., Park, J., & Yi, G. (2015). Attendance check system and implementation for wi-fi networks supporting unlimited number of concurrent connections. International. *Journal of Distributed Sensor Networks*, 11(7), 1-10.

Finkenzeller, K. (2010). *Fundamentals and applications in contactless smart cards, radio frequency identification and near-field communication*. (D. Müller, Trans.). Chichester, West Sussex: Wiley.

Fishman, B.J., Penuel, W.R., Allen, A., Cheng, B.H., & Sabelli, N. (2013). Design-based implementation research: An emerging model for transforming the relationship of research and practice. *National Society for the Study of Education*, 112(2), 136-156.

Gupta, I., Patil, V., Kadam, C., & Dumbre, S. (2016). *Face Detection and Recognition using Raspberry Pi*. Paper presented at International WIE Conference on Electrical and Computer Engineering (WIECON-ECE). Pune, India. doi: 10.1109/WIECON-ECE.2016.8009092.

Home Assistant (2018). Documentation. (2018, December 30). Retrieved from: https://www.home-assistant.io/docs/.

IEEE Specturm (2018). Interactive: The Top Programming Languages 2018. (2018, February 04). Retrieved from: https://spectrum.ieee.org/static/interactive-the-top-programming-languages-2018

Kazi, S., Pasha, F., Gorme, F., & Bata, H. (2017). Raspberry Pi in Attendance Tracking System. *International Journal of Computer Applications*, 162 (7), 15-19.

Kiran, P.P., & Kumar, R.G. (2017). Image processing based student attendance system using Raspberry Pi. *International Journal of Engineering and Computer Science*, 6(4), 21145-21148.

Lim, T.S., Sim, S.C., & Mansor, M.M. (2009). *RFID Based Attendance System*. Paper presented at IEEE Symposium on Industrial Electronics and Applications. doi:10.1109/ISIEA.2009.5356360

Lutz, M. (2013). *Learning Python*. Sebastopol, CA: O'Reilly Media.

Maramis, G.D.P., & Rompas, P.T.D. (2018). *Radio Frequency Identification (RFID) Based Employee Attendance Management System*. Paper presented at IOP Conference Series: Materials Science and Engineering. 306 (1). doi:10.1088/1757-899X/306/1/012045.

MySQL (2018). MySQL 8.0: Up to 2x Faster. (2018, December 30). Retrieved from: https://www.mysql.com/products/enterprise/database/

Nainan, S., Parekh, R., & Shah, T. (2013). RFID Technology Based Attendance Management System. *International Journal of Computer Sciences Issues*. 10(1), 516-521.

Nguyen, H.K., & Chew, M.T. (2017). *RFID-Based Attendance Management System*. 2nd Workshop on Recent Trends in Telecommunications Research, Palmerston North, New Zealand. doi:10.1109/RTTR.2017.7887874.

Priya, D.S. & Umasankar, M. (2016). Modern Attendance System. *International Research Journal of Engineering and Technology*. 3(8), 128-132.

Python Software Foundation (2018). Comparing Python to Other Languages. (2018, December 30). Retrieved from: https://www.python.org/doc/essays/comparisons/.

Qaiser, A., & Khan, S.A. (2006, November 13-14). *Automation of Time and Attendance using RFID Systems*. Paper presented at the 2nd International Conference on Emerging Technologies. doi:10.1109/ICET.2006.335928

Rajkumar, S., & Prakash, J. (2016). Automated attendance using Raspberry Pi. *International Journal of Pharmacy &Technology*, 8(3), 16214-16221.

Raspberry Pi Foundation (n.d). Raspberry Pi 3 Model B+. (2018, December 30). Retrieved from: https://www.raspberrypi.org/documentation/hardware/raspberrypi/mechanical/rpi_MECH_3bplus.pdf.

Rjeib, H.D., Ali, N.S., Farawn, A.A., Al-Sadawi, B., & Alsharqi, H. (2018). Attendance and Information System using RFID and Web-Based Application for Academic Sector. *International Journal of Advanced Computer Science and Applications*, 9 (1), 266-274.

Robinson, A., Cook, M., Evans, J., & McManus, S., (2014). *Raspberry Pi Projects.* Chichester, West Sussex: Wiley

Salim, O.A.R., Olanrewaju, R.F., & Balogun, W.A. (2018). *Class Attendance Management System Using Face Recognition*. Paper presented at International

Conference on Computer and Communication Engineering (ICCCE). Kuala Lumpur, Malaysia. doi: 10.1109/ICCCE.2018.8539274

Sangewar, M.P., Waychol, S.R., & Manekar, A. (2015). Proposed automated students attendance management system using Raspberry Pi and NFC. *International Journal of Research in Computer & Information Technology.* 1(1), 93-95.

Shah, D.K., Bharadi, V.A., Kaul, V.J., & Amrutia, S. (2015). *End-to-End Encryption Based Biometric SaaS: Using Raspberry Pi as a Remote Authentication Node.* Paper presented at International Conference on Computing Communication Control and Automation (IEEE). Pune, India. doi:10.1109/ICCUBEA.2015.19.

Shrestha, R., Pradhan, S.M., Karn, R., & Shrestha, S. (2018). *Attendance and Security Assurance using Image Processing.* Paper presented at International Conference on Computing Methodologies and Communication (ICCMC). Erode, India. doi: doi:10.1109/ICCMC.2018.8487788.

Silva, F., Filipe, V., & Pereira, A. (2008). *Automatic control of students' attendance in classrooms using RFID.* Paper presented at 3rd International Conference on Systems and Networks Communications. doi:10.1109/ICSNC.2008.70

Srivastava, L. (2005). RFID: *Technology, Applications and Policy Implications*. Presented at the International Telecommunication Union, Kenya.

Sutar, M., Patil, M., & Waghmare, S. (2016). Smart Attendance System Using RFID in IOT. *International Journal of Advanced Research in Computer Engineering & Technology (IJARCET)*. 5(4), 1155-1159.

TIOBE (2018). Index for December 2018. (2018, December 30). Retrieved from: https://www.tiobe.com/tiobe-index/.

Upton, E., & Halfacree, G. (2016). *Raspberry Pi user guide* (4th edition). Chichester, West Sussex: Wiley.

Yumang, A. N., Padilla, D., Sejera, M., Pajarillo, A. C. U., Palmiano, G. V. L. B., & Racho, M. M. F. (2017). *Attendance Checker for Students of Mapúa University*. Paper presented at 9th International Conference on Humanoid, Nanotechnology. Manila, Philippines. doi:10.1109/HNICEM.2017.8269558

Zaman, H.U., Hossain, J.S., Anika, T.T., & Choudhury, D. (2017). *RFID Based Attendance System*. Paper presented at 8th International Conference on Computing Communication and Network Technologies. Delhi, India. doi: 10.1109/ICCCNT.2017.8204180

APPENDIX

Appendix A: Raspberry Pi Source Code

```
#!/usr/bin/env python
# -*- coding: utf8 -*-

import RPi.GPIO as GPIO
import Adafruit_CharLCD
import MFRC522
import MySQLdb
import signal
import time
import datetime
import WriteFile


continue_reading = True
display = Adafruit_CharLCD.lcd()# Capture SIGINT for cleanup when the script is
aborted

# Capture SIGINT for cleanup when the script is aborted
def end_read(signal,frame):
    global continue_reading
    print "Ctrl+C captured, ending read."
    continue_reading = False
    GPIO.cleanup()

# Hook the SIGINT
signal.signal(signal.SIGINT, end_read)
WriteFile = WriteFile.WriteCSV()
# Create an object of the class MFRC522
MIFAREReader = MFRC522.MFRC522()
# Welcome message
print "Welcome to the MFRC522 data read example"
print "Press Ctrl-C to stop."
display.clear()
print("Please Read The Card")
display.message("Please Read\nThe Card")
piId = 1
# This loop keeps checking for chips. If one is near it will get the UID and
authenticate
while continue_reading:

    # Scan for cards
    (status,TagType) =
MIFAREReader.MFRC522_Request(MIFAREReader.PICC_REQIDL)

    # If a card is found
    if status == MIFAREReader.MI_OK:
        print "Card detected"
```

```
    # Get the UID of the card
    (status,uid) = MIFAREReader.MFRC522_Anticoll()

    # If we have the UID, continue
    if status == MIFAREReader.MI_OK:

        # Print UID
        print "Card read UID: "+str(uid[0])+str(uid[1])+str(uid[2])+str(uid[3])

        ReadedUID = str(uid[0])+str(uid[1])+str(uid[2])+str(uid[3])
        #print "Variable: " + ReadedUID

        # This is the default key for authentication
        key = [0xFF,0xFF,0xFF,0xFF,0xFF,0xFF]

        # Select the scanned tag
        MIFAREReader.MFRC522_SelectTag(uid)

        # Authenticate
        status =
MIFAREReader.MFRC522_Auth(MIFAREReader.PICC_AUTHENT1A, 8, key,
uid)

        # Check if authenticated
        if status == MIFAREReader.MI_OK:
            MIFAREReader.MFRC522_Read(8)
            MIFAREReader.MFRC522_StopCrypto1()
            connection = None
            try:
                # Connect to the database
                connection = MySQLdb.connect(host='192.168.254.10', #change Ip on a
different network
                                             user='root',
                                             passwd='',
                                             db='attendance_system')

                piId = 1

                dayofweek = datetime.datetime.today().weekday() + 1



                with connection.cursor() as cursor:
                    # Read a single record

                    sql_query_users = """SELECT * FROM `users` WHERE
users.US_RFID= """+ReadedUID

                    cursor.execute(sql_query_users)
```

```
                resultUser = cursor.fetchone()

                if resultUser is not None:

                    with connection.cursor() as cursorPiId:

                        sql_query_piId = """SELECT * FROM `classes` WHERE
classes.PI_ID= """+str(piId)

                        cursorPiId.execute(sql_query_piId)
                        resultpiId = cursorPiId.fetchone()

                    with connection.cursor() as cursorDays:

                        sql_query_days = """SELECT * FROM `days` WHERE
days.DA_ID= """+str(dayofweek)

                        cursorDays.execute(sql_query_days)
                        resultDays = cursorDays.fetchone()

                    with connection.cursor() as cursorTime:

                        sql_query_time = """SELECT * FROM `course_hours`
                                    WHERE course_hours.CH_START <= sysdate()
                                    AND course_hours.CH_END > sysdate()"""

                        cursorTime.execute(sql_query_time)
                        resultTime = cursorTime.fetchone()

                        if resultTime is not None:

                            with connection.cursor() as cursorTimeTable:

                                sql_query_timeTable = """SELECT * FROM  `time_table`
                                            WHERE time_table.CL_CL_ID
="""+str(resultpiId[0])+""" AND time_table.CH_CH_ID
="""+str(resultTime[0])+""" AND time_table.DA_DA_ID="""+str(resultDays[0])

                                cursorTimeTable.execute(sql_query_timeTable)
                                resultTimeTable = cursorTimeTable.fetchone()

                                if resultTimeTable is not None:

                                    with connection.cursor() as cursorStudentCourse:

                                        sql_qury_studentCourse = """SELECT * FROM
`student_course`
                                                    WHERE
student_course.US_US_ID="""+str(resultUser[0])+""" AND
student_course.CO_CO_ID="""+str(resultTimeTable[2])
```

```
                        cursorStudentCourse.execute(sql_qury_studentCourse)
                        resultStudentCourse = cursorStudentCourse.fetchone()

                        if resultStudentCourse is not None:

                            x = datetime.datetime.now()
                            cursorAtten = connection.cursor()
                            sql_insert_attendance = """ INSERT INTO
`attendance`
                                                    (`AT_TIME`, `AT_DELAY`,
`TT_TT_ID`,`US_US_ID`) VALUES (%s,%s,%s,%s)"""

                            delay= x - resultTime[1]

                            insert_tuple = (x,
delay,resultTimeTable[0],resultUser[0])


                            resultAtten  =
cursorAtten.execute(sql_insert_attendance,insert_tuple)
                            connection.commit()
                            print ("variable inserted successfully into test table")
                            display.clear()
                            display.message("Attendance \nTaken")
                            time.sleep(5)
                            display.clear()
                            print("Please Read The Card")
                            display.message("Please Read\nThe Card")
                        else:
                            print("Not registerd to this course")
                            display.clear()
                            display.message("Not Registerd\nTo This Course")
                            time.sleep(5)
                            display.clear()
                            print("Please Read The Card")
                            display.message("Please Read\nThe Card")

                    else:
                        print("No Lecture On This Day")
                        display.clear()
                        display.message("No Lecture \nOn This Day")
                        time.sleep(5)
                        display.clear()
                        print("Please Read The Card")
                        display.message("Please Read\nThe Card")
                else:
                    print("No Lecture At This Time")
                    display.clear()
                    display.message("No Lecture \nAt This Time")
```

```
                time.sleep(5)
                display.clear()
                print("Please Read The Card")
                display.message("Please Read\nThe Card")


        else:
            print("Student Is Not Registered")
            display.clear()
            display.message("Student Is \nNot Registered")
            time.sleep(5)
            display.clear()
            print("Please Read The Card")
            display.message("Please Read\nThe Card")

    except MySQLdb.Error as e:
        print("connection error")
        display.clear()
        display.message("Connection\nError")
        WriteFile.WriteTo(ReadedUID,piId)
        time.sleep(5)
                            display.clear()
        print("Please Read The Card")
        display.message("Please Read\nThe Card")

    finally:
            try:
                connection.close()
            except:
                None

else:
    print "Authentication error"
```