DEVELOPING ACADEMIC INFORMATION SYSTEM

FOR LATRIKUNDA UPPER BASIC SCHOOL IN THE GAMBIA.

THESIS

ALAGIE SINGHATEH

11160930000118

DEPARTEMENT OF INFORMATION SYSTEM

FACULTY OF SCIENCE AND TECHNOLOGY

STATE ISLAMIC UNIVERSITY, SYARIF HIDAYATULLAH JAKARTA

2020

ABSTRACT

Concentrating on the digitization of different sectors by the Gambian government, it came under the education system of the country that is lagging. Considering an individual's educational information, people are still using paper or documented sheets and no structured or automated system helps to store on student's information from the initial of their educational carrier. Therefore, Academic information system has a curial roll in terms of digitalization of technology. An automated system having information stored and update information frequently can overcome this issue and help the students to access information, share them with national and international educational institutions for higher studies. Furthermore, an academic information system can also help to monitor the educational progress of the country under digital methods. The researcher proposes to break this barrier and bring all the student information under one roof where schools can easily manipulate information to bring important needs in changes. In the same manner, it may help to make the work much easier and help the education sector grow much better than before.

Keywords: Academic information system, developing, designing digitalization.

TABLE OF CONTENT

LIST OF FIGURES

LIST OF TABLES

CHAPTER I

PRELIMINARY

1.1 Introduction

In the Gambia, many Primary and Secondary Schools today, data management is a problem for the school managers and principals they have limited opportunities and capacity to manage their administrative activities using technological advancements such as Mobile Phones, Tablets, and even Computers, through the Internet.

Currently, it is difficult for schools to manage their students and make the workload by the help of technology, Therefore, principals and teachers are facing difficulties every day because of the manual system the schools are using is not helpful and need to have a better system that will make the work easier for the schools within a single click, Latrikunda upper basic as (Latrikunda secondary school) is a school that students and staff are having a massive problem in terms of data accessing. Beside, in term of knowing the capacity of the students that are attending classes, the Parents' contact details of each student, or generating reports on school fee payments for each student. These are just some of the many solutions the researcher is providing by developing the Academic Information System.

According to (Arabaci, 2013) mentioned that human morals, passions a subjectivity have been ignored deliberately the field of education management. Therefore, human activities such as cultural and social contact to understand the distributed management system. Besides, the morals of human activities concern not

only themselves but also the community, it is important that the education system which aims to form human activities should be programmed effectively as value. In this time of modernization, people are tending have information in the blink of an eye. An individual's education information is not out of that need. A centralized system of The Gambian's educational system having student's information stored with all of his performance ratings, academic management information, Although the education management system, its categories as three function of the existence of development the scholars have developed a new strategy to oriented the school management systems as a project that managers will be able to manage and control the system, hence while the students and the teachers will able to link via the system. (Androniceanu & Mascu, 2015).

The server space for sharing lectures, statistical analysis, etc. It can be an effective solution that can help the educational system to execute fast and be more efficient. The system that the researcher developed is a centralized Academic Information System using a distributed database. However, other researchers have showed that the outcomes as per the perspectives of respondents as educators and administrators uncover that there have been bunches of issue sources in connection with the schools in this exploration. Absence of family intrigue, physical deficiency, and problematic school condition, social and social structure is a portion of the issues referred to by the majority of the managers and instructors. What's more, fights inside and outside the school, annoying and debilitating practices have been generally observed (Dermaku et al, 2016)

Although the whole system will be divided based on education and then based on schools. During searching for information and viewing Education exam results, According to (Anca-Georgiana, 2015).

The educational framework is important for the youngsters likewise the instructors and parents are the way to effective to be a knowledgeable working grown-up for a superior and just society. However, the parent's task is one of extraordinary trouble when attempting to offer the right, if there is one, training for their kids. Parental association in school is characterized as "parent announced cooperation at any rate, once amid the school year is going to a general school meeting; going to a planned gathering with their youngster's educator; going to a school occasion, or volunteering in the school or serving on a school council. The distributed system will reduce the server jams and execute efficiently. Considering the recent scenario, education boards have their websites and while some of the schools are not having their websites, However, it appears that secondary schools are not living in the high educational way to be obligated the school should be able to provide quality education for the entire the community to have benefited from it. Moreover, the higher institutions without their useful life in society. (Yuguda kotirde & Jailani, 2015).

In this project, the researcher provided an academic information system for Latrikunda upper basic school to ensure that the system works with them easier and make things flexible. The process of forcing schools to open their website & execute all the digital systems to run school management processes. To make the

school-going generation competencies for the 21st century, technology has a great impact in this era the government has installed two modern laboratories in each school across the country. So, the public and private schools can content classrooms access to computers to make work easier and complex. On the other hand, few schools are trying to set up websites and develop the management which is enabling a scattered scenario for storing data. These schools are under their network, not under any centralized system. So, data accessibility and analyzing in any sort of educational project is very challenging. I ensure a centralized system where all the educational institution's database is less than one platform.

1.2 Identification of Problems

Based on the background above, it can be identified the problems that occur in this study, namely;

1. The representation of the schools that implement academic information system now is increasing stage competition in the same domain. Therefore, schools must be able to process the results of students and even staff desires and apply them on the academic information system based on schools must know the extent of the application made with that program, to be able to compete with satisfaction of the student's staff even the principle on the same platform.
2. The development of Academic information system in the same stage makes the schools must be able to analyze students and staff expectations on the

use of the application platforms, how to provide uniqueness of the application that will help the schools to lead on their success due to implementation of the platform which will enable new features to increase flexibilities and make students and teachers access without having difficulties toward the system. And it will make awareness to the students and staff; this will have an impact on the school if it cannot analyze expectations quickly and thoroughly compared to other knowledge management, in terms of implementation.

1.3 Problem Statement

Based on the problem identification above, the problem formulation can be obtained namely "How to develop academic information system (AIS) using Laravel 6 PHP Framework in the Latrikunda"

1.4 Limitations

This study is limited by various things, namely;

1. This study was conducted at the Latrikunda Upper Basic School.
2. The method used is the stage of the Design Implementation approach, this framework is expected to provide a convenient way to audit the knowledge management, and thus enterprises can make corrections and adjustments accordingly to greatly enhance their chances of success while implementing the knowledge management systems for school. (Shu-Mei, 2005)

3. Include a discussion of the six major activities for the implementation stage as described within the text: (1) coding, (2) testing, (3) installation, (4) documentation, (5) training, and (6) support. The discussion of these six activities should describe specifically how each activity would be planned for the individual project situation.

1.5 Research Objectives

1. Provide separate profiling for students and teachers
2. Provide accessibility for students to review their profile, see results
3. Provide accessibility for teachers to share the lectures for the students, schedule exams, and review all the contents needed for the lectures.
4. .Provide file-sharing space for Teachers throughout the system will be able to share all their important educational content with other teachers which can help distribute equal standards of education all over the country.
5. Provide auto exam scheduler for the schools. However, class routines and calendars can also be made using this system.
6. Provide auto mark sheets and email sending to the desired email of students in case of publishing. Common syllabus and lecture patterns can be provided centrally by the education board and it can ensure a similar pattern of education throughout the country.
7. Provide a unique ID number using their roll number, board, and school ID no. respectively. This ID cannot be changed and remains the same even if a student changes his or her institution.

8. Provide a Login credentials for each student such as (username and password) to log into the system for viewing their profile and updates related to them.
9. Provide three different levels for the system, the Admin will control the whole system based on their priorities. The three different levels are supper administrator, Teacher, and Students. The school administrator will control the information related to their respective schools, supper administrator controls everything under the system and student has the accessibility over everything based on his/her credentials and role system.

   The system will be able to handle a huge number of users at the same time.

1.6 Research Benefits

The benefits of this research are;

1. **For Researchers**

a. Implementing knowledge to obtained during the lecture board.

b. Comparing the information system theory that has been obtained during lectures with the actual problems that occur

c. Provide a general description of the application of information management systems in Latrikunda Upper Basic School.

2. **For the University**

1. Knowing the ability of the students in understanding the subject or topic obtained in the institute.
2. Knowing the ability of students in applying their understanding of evaluation material.

3. Provide a summary of the readiness of students in terms of facing the real world of work.
4. Become one of the references and can be re-developed for future researchers.

3. For the School

1. Facilitating a firm to know more deeply about information systems based on data management in the existing era.
2. The schools/organizations know the importance of permanence to the sector that is capable of a good academic information system application that needs to be developed.

1.7 Research Methodology

The research method that used in this development of the academic information system case study method, by using this method may help the researcher to able to gathered a correct and receive accurate data with the help of the organization or school. Data collection was carried out by observation, interview, and questionnaire.

Observation, as the name implies, is a way of collecting data through observing. The observation data collection method is classified as a participatory study because the researcher has to immerse himself in the setting where his respondents are while taking notes and/or recording. The results achieved are looking at the business processes of the organization or even through a survey that occurs, and see all activities or find data that is needed for the research. This direct observation is carried out under the headmaster of the parties concerned.

The Design Implementation approach, this framework is expected to provide a convenient way to audit knowledge management, and thus enterprises can make corrections and adjustments accordingly to greatly enhance their chances of success while implementing the knowledge management systems for a school. (Shu-Mei, 2005). The methodology for applying the Information system in this study was carried out using the Design Implementation approach supported by the Framework Design Libraries. Include a discussion of the six major activities for the implementation stages as described within the text: (1) coding, (2) testing, (3) installation, (4) documentation, (5) training, and (6) support. The discussion of these six activities should describe specifically how each activity would be planned for the individual project situation. The stakeholder platform analysis, services analysis, core value propositions analysis, infrastructure and core components analysis, transaction analysis, channel and contexts analysis, partner's analysis and work analysis. After that, the researcher will design a use case with its functionalities by using design tools that will be provided using the tools UML to describe the user's expectations on the use of the web application version of the Latrikunda Upper basic school platform.

1.8 Systematics of Writing

The discussion on the preparation of the research report is divided into five chapters that will be briefly described as follows:

CHAPTER 1 PRELIMINARY

This chapter provides a brief explanation of the background to the problem, the formulation of the problem, the limitations of the study, the objectives and benefits of research, research methods, and systematic writing.

CHAPTER II THEORETICAL FRAMEWORK

This chapter explains general theories, specific theories, supporting theories, and reference data on research-based materials.

CHAPTER III RESEARCH METHODOLOGY

This chapter explains the data needed the mindset, data collection methods, and analytical methods used at the research activity stage.

CHAPTER IV RESULTS AND DISCUSSION

This chapter explains the profile of Latrikunda Upper Basic and discusses curial points to explain the analysis and the results of Development of Academic information system.

CHAPTER V CONCLUSION

This chapter explains the conclusions of the stages that have been made from the development of the academic information system and suggestions for further developers.

CHAPTER II

THEORETICAL FRAMEWORK

2.1 Basic Concepts of Academic Information System (AIS)

Academic is a field that learns about the curriculum or inner learning its function is to increase knowledge in terms of education that can be managed by a school or educational institution. Therefore, the Academic information system is a system that provides information services in the form of data in matters relating to academic data (Agnes et al., 2018).

2.2 Definition of System

A system is a group of interrelated components working together toward a common goal by accepting inputs and producing outputs in an organized transformation process. The system will have the following basic interacting components (functions). (Sullivan, 2012) . A set of interrelated components, with a clearly defined boundary, working together to achieve a common set of objectives by accepting inputs and producing outputs in an organized transformation process. Therefore, elements or components that interact to accomplish goals. The elements themselves and the relationships among them determine how the system works (Stair & Reynolds, 2010). The system can be defined as a collection of components that are interconnected with one another form a unity to achieve certain goals. (Soediono et al., 2014).

Based on the above definition it can conclude regarding the system. The system is a collection of interconnected parts forming a unit to achieve certain goals.

2.2.1 System Characteristics

A system has characteristics namely component or element, the system boundary, environment outside the system, interface, input, process, output, objective, and goals. (Herdiana, 2013).

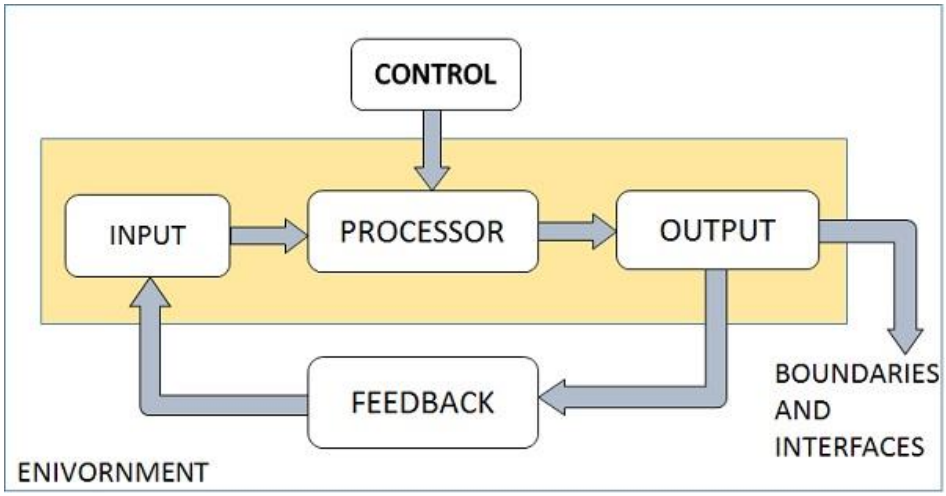


Figure 2. 1 System Characteristics (Source: Herdiana, 2013).

1. System Components

According to (Herdiana, 2013). A system is not an empty environment but a system is located and functioning inside another system environment. A system consists of several components that interact with each other, work together to find one unite.

A system is from large system components, it can be call subsystem, while the bigger one is its environment. Each subsystem has the properties of the system to run a certain function and affect an overall system process.

2. System Boundary

System boundaries are constraints or separators between a system with other systems or with the environment outside the system boundaries determine the scope configuration or even system capability. These system limits allow a system to be seen as one. The system boundary also shows the scope (scope) of the system.

3. System Environment

The outside environment is anything outside limits of the system that can affect system operations, both of which beneficial or harmful. This beneficial effect must be guarded so that it will support the continuity of operation of the system.

The influence of adverse environments must be resisted and controlled control so as not to interfere with the continuity of a system.

4. System Interface

According to (Untuk et al., 2017). Explained that, through this link allows resources to flow from the subsystem to the other subsystems. By connecting, one subsystem can integrate with other subsystems to form a single unit.

Therefore, According to R, Stair in his book (Stair & Reynolds, 2010) mentioned that. “A system’s has four components consist which as follows”:

1. **Input**

In information systems, the input is the activity of gathering and capturing raw data. In producing paychecks, for example, the number of hours every employee works must be collected before paychecks can be calculated or printed. In a university grading system, instructors must submit student grades before a summary of grades for the semester or quarter can be compiled and sent to the students.

2. **Process**

In information systems, process means to convert or transform data into useful outputs. The process can involve making calculations, comparing data and taking alternative actions, and storing data for future use. Processing data into useful information is critical in business settings.

3. **Output**

In information systems, output involves producing useful information, usually in the form of documents and reports. Outputs can include paychecks for employees, reports for managers, and information supplied to stockholders, banks, government agencies, and other groups. In some cases, the output from one system can become input for another. For

example, the output from a system that processes sales orders can be used as input to a customer billing system.

4. **Feedback**

In information systems, feedback is information from the system that is used to make changes to input or processing activities. For example, errors or problems might make it necessary to correct input data or change a process. Consider a payroll example. Perhaps the number of hours an employee worked was entered as 400 instead of 40. Fortunately, most information.

2.3 Basic Concept of Information

2.3.1 Definition of Data

According to (Untuk et al., 2017). Explained that, through this link allows resources to flow from the subsystem to the other subsystems. By connecting, one subsystem can integrate with other subsystems to form a single unit. Therefore, According to R, Stair in his book (Stair & Reynolds, 2010) mentioned that. “A system’s has four components consist which as follows”:

1. **Input**

In information systems, the input is the activity of gathering and capturing raw data. In producing paychecks, for example, the number of hours every employee works must be collected before paychecks can be

calculated or printed. In a university grading system, instructors must submit student grades before a summary of grades for the semester or quarter can be compiled and sent to the students.

2. Process

In information systems, process means to convert or transform data into useful outputs. The process can involve making calculations, comparing data and taking alternative actions, and storing data for future use. Processing data into useful information is critical in business settings.

3. Output

In information systems, output involves producing useful information, usually in the form of documents and reports. Outputs can include paychecks for employees, reports for managers, and information supplied to stockholders, banks, government agencies, and other groups. In some cases, the output from one system can become input for another. For example, the output from a system that processes sales orders can be used as input to a customer billing system.

4. Feedback

In information systems, feedback is information from the system that is used to make changes to input or processing activities. For example, errors or problems might make it necessary to correct input data or

change a process. Consider a payroll example. Perhaps the number of hours an employee worked was entered as 400 instead of 40. Fortunately, most information

2.4 Basic Concept of Information System

The basic concepts of an information system will be clarified with some understanding starting from the system, system characteristics, information, information systems, and components of information systems.

2.4.1 Definition of Information System

An information system can be defined technically as a set of interrelated components that collect (or retrieve), process, store, and distribute information to support decision making and control in an organization. In addition to supporting decision making, coordination, and control, information systems may also help managers and workers analyze problems, visualize complex subjects, and create new products. (Laudon & Laudon, 2014). According to (Aliyu, 2015) explained that "implemented within an organization to improve the effectiveness and efficiency of that organization". Beside (Sullivan, 2012) described the kind of An information system differs from other kinds of systems in that its objective is to monitor/document the operations of some other system, which we can call a target system. Therefore, an information system (IS) can be any organized combination of people, hardware, software, communications networks, data resources, and policies and procedures that stores, retrieves, transforms, and

disseminates information in an organization. People rely on modern information systems to communicate with one another using a variety of physical devices (hardware), information processing instructions and procedures (software), communications channels (networks), and stored data (data resources).

An Information System is a system that gathers data and disseminates information with the sole purpose of providing information to its users. (Wall et al., 2015).

An information system can be defined technically as a set of interrelated components that collect (or retrieve), process, store, and distribute information to support decision making and control in an organization. In addition to supporting decision making, coordination, and control, information systems may also help managers and workers analyze problems, visualize complex subjects, and create new products. (Is, 2011). Mentioned in (Darmalaksana, 2018). Information systems that connect each stakeholder to the main processes and supporting activities at the university. The product and service of a university vary in nature in which each of the products and services will have different users, both externally and internally

An information system (IS) is a set of interrelated elements or components that collect (input), manipulate (process), store, and disseminate (output) data and information, and provide a corrective reaction (feedback mechanism) to meet an objective (Stair & Reynolds, 2010).

Based on the several definitions above can conclude that an information system is a group of component that consists, people, data, software, hardware, and network.

The basic components of information systems are listed below

2.4.2 Components of Information Systems

1. **Resources of people:** (end users and IS specialists, system analysts, programmers, data administrators, etc.).
2. **Hardware:** (Physical computer equipment and associate device, machines, and media).
3. **Software:** (programs and procedures).
4. **Data:** (data and knowledge bases).
5. **Networks:** (communications media and network support).

1. People Resources

a. End-users: (also called users or clients) are people who use an information system or the information it produces. They can be accountants, salespersons, engineers, clerks, customers, or managers. Most of us are information system end-users.

b. IS Specialists: people who develop and operate information systems. They include systems analysts, programmers, testers, computer operators, and other managerial, technical, and clerical IS personnel.

c. Briefly, systems analysts design information systems based on the information requirements of end-user, programmers prepare computer programs based on the specifications of systems analysts, and computer operators operate large computer systems.

2. Hardware Resources

a. Machines: as computers and other equipment along with all data media, objects on which data is recorded and saved.

b. Computer systems: consist of a variety of interconnected peripheral devices. Examples are microcomputer systems, mid-range computer systems, and large computer systems.

3. Software Resources

Software Resources include all sets of information processing instructions. This generic concept of software includes not only the programs, which direct and control computers but also the sets of information processing (procedures). Software Resources includes:

a. System software, such as an operating system
b. Application software, which are programs that direct processing for a particular use of computers by end-users.

4. Procedures

Which are operating instructions for the people, who will use an information system? Examples are instructions for filling out a paper form or using a particular software package**.**

5. Data Resources

Data resources include data (which is the raw material of information systems) and database. Data can take many forms, including traditional alphanumeric data, composed of numbers and alphabetical and other characters that describe business transactions and other events and entities. Text data, consisting of sentences and paragraphs used in written communications; image data, such as graphic shapes and figures; and audio data, the human voice and other sounds, are also important forms of data. Data resources must meet the following criteria:

6. Comprehensiveness:

This means that all the data about the subject are present in the database.

7. Non-redundancy:

Means that each piece of data exists only once in the database.

8. Appropriate structure:

Means that the data are stored in such a way as to minimize the cost of expected processing and storage. The data resources of IS are typically organized into: Processed and organized data-Databases. Knowledge in a variety of forms such as facts, rules, and case examples about successful business practices.

9. Network Resources

Telecommunications networks like the Internet, intranets, and extranets have become essential to the successful operations of all types of organizations and their computer-based information systems. Telecommunications networks consist of computers, communications processors, and other devices interconnected by communications media and controlled by communications software. The concept of Network Resources emphasizes that communications networks are a fundamental resource component of all information systems. Network resources include:

a. Communications media: such as twisted-pair wire, coaxial cable, fiber-optic cable, microwave systems, and communication satellite systems.
b. Network support: This generic category includes all of the people, hardware, software, and data resources that directly support the operation and use of a communications network.

Examples include communications control software such as network operating systems and Internet packages.

c. Those are the components that integrated to perform the role of the information system, software, hardware, and the network is a technical sub-system.
d. Data and people can be categorized as social sub-systems.

Below is the figure that shows the interaction of the fundamental concepts of the components and their activities of the information system.

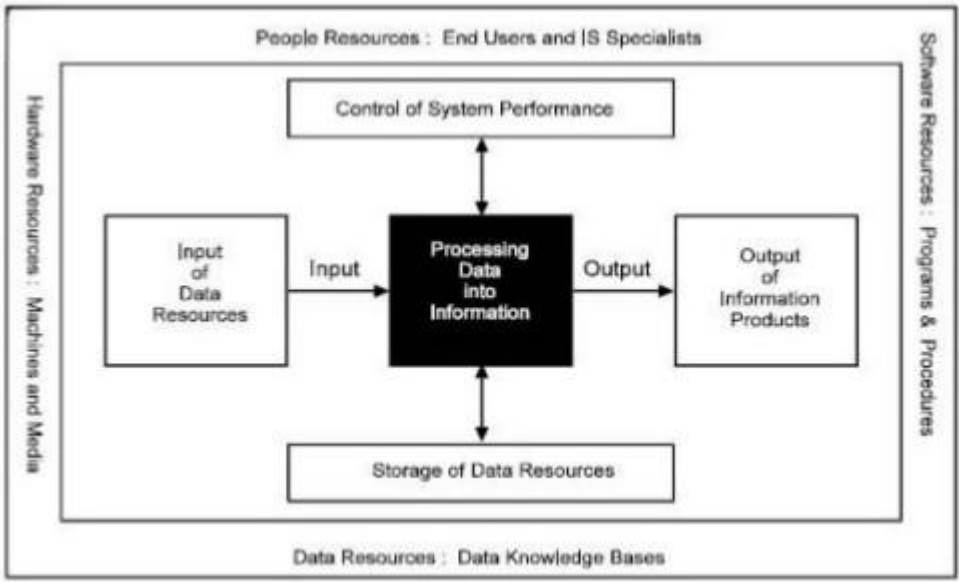


Figure 2. 2 Components of Information System (Source: Is, 2011).

2.5 Definition of Application

On the other hand, an application program (app or application for short) is a computer program designed to perform a group of coordinated functions, tasks, or activities for the benefit of the user (Eddy Tukamushaba Kurobuza & Godfrey Ahumuza Ikaanga, 2008). Furthermore, an application is any program or group of programs that are designed for the end-user. Applications software (also called end-user programs) includes such things as database programs, word processors, Web browsers, and spreadsheets. Therefore, applications sit on top of systems software because they are unable to run without the operating system and system utilities. Systems software consists of low-level programs that interact with the computer at a very basic level. This includes operating systems, compilers, and utilities for managing computer resources. (https://www.webopedia.com/TERM/A/application.html, n.d.). Information system also referred as an application program or application software is a computer software package that performs a specific function directly for an end-user or, in some cases, for another application. An application can be self-contained or a group of programs. The program is a set of operations that runs the application for the user. (https://searchsoftwarequality.techtarget.com/definition/application, n.d.).

2.5.1 Types of Applications

Applications can vary in many ways, including how they're built, what platform they run on, whether they are open source or proprietary, or for which market they are used.

1. **Native application**

To run on specific hardware, such as a camera or GPS, and in the same programming language as the underlying OS.

2. **Web Application**

Therefore, an end-user typically accesses a web application via a web browser, such as Google Chrome. A developer can write web applications in several languages, including JavaScript, CSS, and HTML, but they cannot access the hardware on which the application is installed.

3. **Hybrid Application**

Hybrid application has APIs that can access device resources, similar to a native application but are typically written in languages such as HTML and CSS. Additionally, developers often code mobile applications to work on a specific device platform.

2.6 Basic Concept of Information Technology (IT)

2.6.1 Definition of Technology

According to (Hassan & Gil-Garcia, 2011). Stated that Internet growth makes feasible their use by an increased number of people around the world. Furthermore, the use of this technology is a right for everybody and more in the public administration scope in which, a lot of services must be available for users

and in a correct way. Therefore, the use of electronic technology instead of a paper-based system usually results in increased efficiency, besides; the movement to use technology to improve public administration services has been launched. It is thus critical that all necessary steps be taken to make these new ventures a success.

2.6.2 Definition of Information Technology

To explain the impact of information technology we must look beyond the functional aspects of business organizations. Therefore, it is important to give extra care and attention to planning the resources needed to set up a sound information base and a workable system for keeping it up to date.

According to (Putra et al., 2018) in his book Tactical steps for e-government development stated that "Information technology can help build trust by enabling citizen involvement in the policy process, promote open and accountable government and help prevent corruption. Therefore, when you come to IT, it helps improve transparency in decision- making processes by making it easier for information to be accessible - publishing debates and meetings, budgets and expenditures, results, and governmental reasons for taking important decisions. Furthermore, it encourages citizens to provide constructive advice on public issues and assess the impact of technology implementation to open policy processes".

According to (Mark Charlton, 2009).in his "Handbook of Information Technology" stated that the impact of information technology is shown to have a

far wider effect upon organizations than the traditional models may lead us to suppose more. Therefore, technology implementation is sensitive to this. Information, however, is only part of the process. On its own, it is not particularly a dynamic of the organization. Individuals may receive information and do nothing with it.

According to (Sutabri, 2012). In his book "Information technology can be defined as a combination of computer and telecommunications technology with other technologies such as hardware, software, database, network technology, and other telecommunications equipment".

Information technology broadly defined as the collection of computer systems used by an organization. Therefore, Information technology, in its narrow definition, refers to the technological side of an information system. It includes hardware, software, databases, networks, and other electronic devices. It can be viewed as a subsystem of an information system. (Is, 2011).

2.7 Research Methods (RM)

2.7.1 Data Collection Method

1. Observation

Observation is a technique or approach to get primary data by observing the data object directly. (Jogiyanto, 2013).

2. Interview

An interview is a two-way communication to get data from respondents. (Jogiyanto, 2013).

3. Questionnaire

Questionnaires are data collection techniques by giving a set of questions or written statements to respondents to answer them. (Sugiyono, 2013).

4. Literature Review

A literature study is a research conducted to retrieve theoretical data which is then used as supporting literature to support the research conducted. Data can be obtained from books, literature, journals, articles, notes, and reports that can be used as a reference and its relation to the problem under study. (Sugiyono, 2013).

2.8 Rapid Application Diagram (RAD)

In this process which is rapid application development (RAD), it's a method that links between analysis and design process.

2.8.1 Definition of RAD

According to (Whitten, 2007). RAD is a system development strategy that emphasizes speed of development through extensive user involvement in the rapid, iterative, and incremental construction of a series of functioning prototypes of a system that eventuality evolves into the final system (or a version). According to (Dennis, 2015). In his book states that Rapid application development (RAD) based methodologies attempt to address both weaknesses of structured design methodologies by adjusting the SDLC phases to get some part

of the system development in the 1990s to get some part of the system developed quickly and into the hands of the users. In this way, the users can better understand the system and suggest revisions that bring the system closer to what is needed. Rapid Application Development (RAD) is an incremental software development process, which emphasizes a very short development cycle. For the development of normal information, the system requires a minimum of 180 days but using the RAD method a system can be completed in only 30-90 days. (Saputra, Pramana Yoga and Siahaan, 2014).

According to (Kendall & Kendall, 2008).Rapid application development (RAD) is an object-oriented approach to systems development that includes a method of development as well as software tools while talking about the SDLC methodology the main purpose is to provide developers and the end-users to directly able to use the system in a very easiest way and more interactive, when you come to the human-computer interaction (HCI) is simply collaborate with RAD to enable the access to different, because RAD can produce a system quickly, by providing the methods that mentioned by the searchers, through the course an information system which is made far distance with the development process. Also, there are various ways that developers interact with rapid application development.

Once RAD is implemented, users can become part of the overall SDLC system process by acting as decision-makers at each stage of development. Rapid application development can produce a system that enables the time for redeveloping after the implementation process. Therefore, an application

builder is a tool that allows the programmers to assemble the software in the way that computer manufacture assembles hardware. In the past when talking about the application builder are many knows as programmers, they face many challenges while trying to build a system, hence when this system came it helped them to build any application within few months. The object that fits the system in building any application has the right to these phases which are planning, analysis, and implementation. Those are the process that many application builders need to concentrate mostly to understand the large components of the RAD. Moreover, in (Kendall & Kendall, 2008). Explained the three board phases of RAD that engage both users and analysts in assessment, design, and implementation. Therefore, RAD involves users in each part of the development effort, with intense participation in the business part of the design.

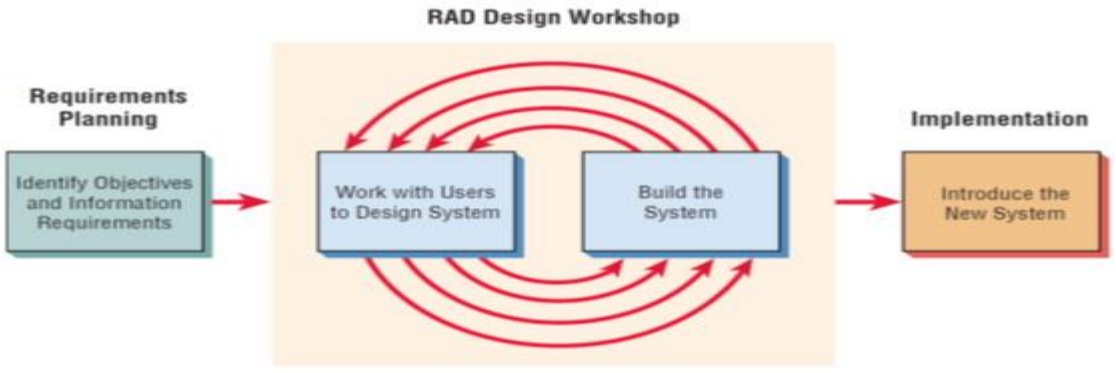


Figure 2. 3 The RAD design workshop. (Kendall & Kendall, 2008)

1. **Requirements Planning Phase**

In the requirements planning phase, users and analysts meet to identify objectives of the application or system and to identify information requirements arising from those objectives. This phase requires intense involvement from both groups; it is not just signing off on a proposal or document. In the requirements planning phase, when information requirements are still being addressed, you may be working with the CIO (if it is a large organization) as well as with strategic planners, especially if you are working with an e-commerce application that is meant to further the strategic aims of the organization. The orientation in this phase is toward solving business problems. Although information technology and systems may even drive some of the solutions proposed, the focus will always remain on reaching business goals.

2. Rad Design Workshop Phase

The RAD design workshop phase is a design-and-refine phase that can best be characterized as a workshop. When you imagine a workshop, you know that participation is intense, not passive and that it is typically hands-on. During the RAD design workshop, users respond to actual working prototypes and analysts refine designed modules based on user responses. Usually, participants are seated at round tables or in a U-shaped configuration of chairs with attached desks where each person can see the other and where there is space to work on a notebook computer.

3. Implementation Phase

During the implementation phase, users are trained, organizational changes, and, in parallel with the implementation of the new system, work is carried out with the existing system According to (Kendall & Kendall, 2014). Mentioned that as soon as these aspects are agreed on and the systems are built and refined, the new systems or part of systems are tested and then introduced to the organization. Because RAD can be used to create new e-commerce applications for which there is no old system. According to (Dennis et al., 2015). User training and rolling out the final product onto a production platform. The primary deliverable is the actual executable information system. The other deliverables include user manuals, a plan to support the users, and a plan for upgrading the information system in the future.

2.8.2 When to Use Rad

As an analyst, you want to learn as many approaches and tools as possible to facilitate getting your work done most appropriately. Certain applications and systems work will call forth certain methodologies. Consider using RAD when:

1. Your team includes programmers and analysts who are experienced with it; and
2. There are pressing business reasons for speeding up a portion of application development; or
3. When you are working with a novel e-commerce application and your development team believes that the business can sufficiently benefit over

their competitors from being an innovator if this application is among the first to appear on the Web; or

4. When users are sophisticated and highly engaged with the organizational goals of the company.

2.9 Object-Oriented System Analysis and Design Using UML

According to (Kendall & Kendall, 2014). Object-oriented techniques work well in situations in which complicated information systems are undergoing continuous maintenance, adaptation, and redesign. Moreover, Object-oriented analysis is a technique that focuses on modeling objects that encapsulate the concerns of data and processes that act on the data. An object-oriented (OO) approach to programming requires techniques for object-oriented analysis (OOA) and object-oriented design (OOD). Some of the object-oriented diagrams, such as class diagrams and sequence diagrams would be inappropriate except when the system will be implemented in an object-oriented environment.

In 1995, Rational Software brought three industry leaders together to create a single approach to object-oriented systems development. Grady Booch, Ivar Jacobson, and James Rumbaugh worked with others to create a standard set of diagramming techniques known as the Unified Modeling Language (UML). The objective of UML was to provide a common vocabulary of object-oriented terms and diagramming techniques rich enough to model any systems development project from analysis through implementation. In November 1997, the Object Management Group (OMG) formally accepted UML as the standard for all object developers. During the

following years, the UML has gone through multiple minor revisions. (Dennis et al., 2015)

2.9.1 System Concepts for Object Modeling

Object-oriented analysis is based on several concepts, some of these concepts require a new way of thinking about systems and the development process, Furthermore, the static structure and dynamic behavior models of the information system instead of defining data and process models, which is the goal of traditional development approaches.

2.9.1.1 Objects, Attributes, Methods, and Encapsulation

According to (Kendall & Kendall, 2014). An object-oriented approach. Is the system analyst that must understand the principles of organizations and have a working knowledge of data-gathering techniques? In cooperating with (Whitten, 2007).

Explained the functionalities that need to be known as follows:

1. ***Object*** something that is or is capable of being seen, touched, or otherwise sensed and about the users store data and associate behavior
2. ***Attribute*** the data that represents characteristics of interest about an object

3. ***Object Instance*** each specific person, place, thing, or event. As well as the values for the attributes of that object. Sometimes re 'erred too simply as an object.

4. ***Behavior*** the set of things that an object can do and that correspond to functions that act on the object's data (or attributes).ln object-oriented circles, an object's behavior is commonly referred to as a method, operation, or service (we may use the terms interchangeably throughout our discussion).

5. ***Encapsulation*** **of** the packaging of several items together into one unit.

2.9.1.2 Classes, Generalization, and Specialization

Another important concept of object modeling is the concept of categorizing objects into object classes. Let's consider some of the objects within your current environment.

1. ***Object Class*** is a set of object instances that has the same attributes and behaviors. Often referred to simply as a class.
2. ***Inheritance*** the concept wherein methods and/or attributes defined in an object class can be inherited or reused by another object class.
3. ***Generalization/ Specialization*** technique wherein the attributes and behaviors that are common to several types of object classes are grouped (or abstracted) into their class, called a super type. The attributes and

methods of the super type object class are then inherited by those object classes (subtype). Sometimes abbreviated as gen/spec.

4. ***Super type*** an entity that contains attributes and behavior that are common to one or more class subtype, also referred to as abstract or parent class.
5. **Subtype** an object class that inherits attributes and behaviors from a super type class and then may contain other attributes and behaviors that are unique to it also re. Referred to as child class and, if it exists at the lowest level of the inheritance hierarchy, as a concrete class.

2.9.1.3 Object Class Relationships

Conceptually, objects do not exist in isolation. The things that they represent interact with and impact one another to support the business mission. Thus an object c1ass relationship is inevitable. You, communicating with them. Similarly, objects interact with other objects within a systems environment. The object classes customer and order that may exist in a typical information system. We can make the following business assertions about how customers and orders are associated (or Interact).

2.9.1.4 Messages and Message Sending

Object classes Interact or communicate" with one another by passing messages. Recall the concept of encapsulation, wherein an object is a

package of attributes and behavior. Only that object can perform its behavior and action its data. (Whitten, 2007).

2.10 System Development Tools (SDT)

(*Unified Modeling Language*) UML is a powerful tool that can greatly improve the quality of your systems analysis and design, and it is hoped that the improved practices will translate into higher-quality systems. By using UML iteratively in analysis and design, you can achieve a greater understanding between the business team and the IT team regarding the system requirements and the processes that need to occur in the system to meet those requirements. (Kendall & Kendall, 2008).

During 1995, Rational Software brought three industry leaders together to create a single approach to object-oriented systems development. Grady Booch, Ivar Jacobson, and James Rumbaugh worked with others to create a standard set of diagramming techniques known as the Unified Modeling Language (UML). The objective of UML was to provide a common vocabulary of object-oriented terms and diagramming techniques rich enough to model any systems development projects from analysis through implementation. In November 1997, the Object Management Group (OMG) formally accepted UML as the standard for all object developers. During the following years, the UML has gone through multiple minor revisions. The current version of UML is Version 2.5. (Dennis et al., 2015).

1. Use case diagram

The major strength of the use-case diagram is that it provides the user with an overview of the business processes. However, remember that any time a use case changes, it could affect the use case diagram. There are four major steps in drawing a use-case diagram. (Dennis, 2015).

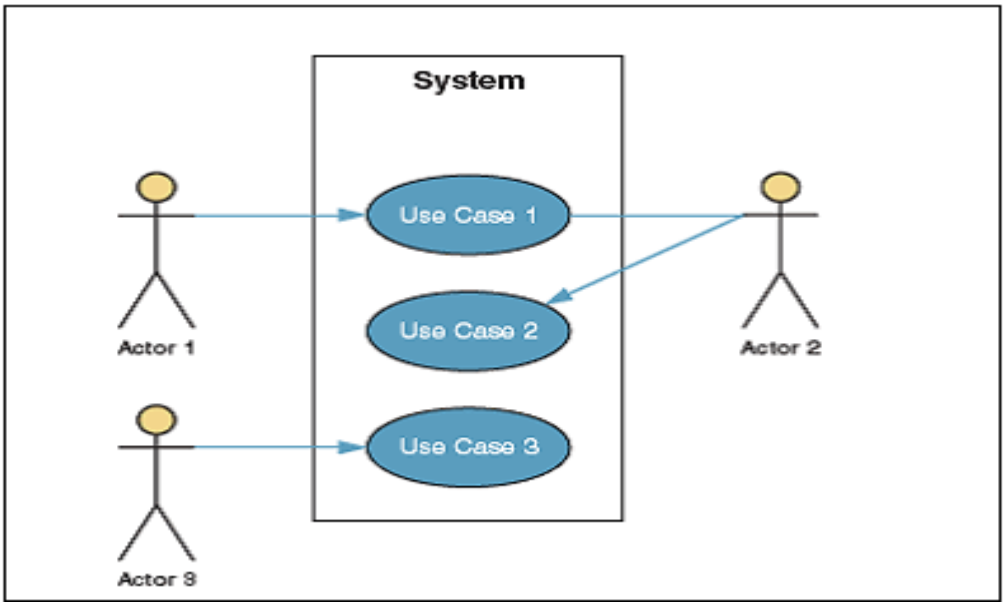


Figure 2. 4 Use Case Diagram (Whitten, 2007).

The three main keys elements in the use case diagram are as follows:

1. Use Case

Use cases are the primary drivers for all the UML diagramming techniques. A use case communicates at a high level what the system needs to do, and all the UML diagramming techniques build on this by presenting the use-case functionality in a different way for a different

purpose. Use cases are the building blocks by which the system is designed and built. (Kendall & Kendall, 2014).

2. Actor

An actor is not a specific user but instead is a role that a user can play while interacting with the system. Therefore, an actor can also represent another system in which the current system interacts. The actors represent the principal elements in the environment in which the system operates. Actors can provide input to the system; receive output from the system, or both. (Dennis, 2015).

3. Relationships

A relationship is depicted as a line between two symbols on the use-case diagram. The meaning of the relationships may differ depending on how the lines are drawn and what types of symbols they collected. (Whitten, 2007).

Use cases are connected to actors through association relationships; these relationships show with which use cases the actors interact. A line drawn from an actor to a use case depicts an association. The association typically represents two-way communication between the use case and the actor. If the communication is only one way, then a solid arrowhead can be used to designate the direction of the flow of information. (Dennis et al., 2015).

2. Activity Diagram

A diagram that can be used to graphically depict the flow of a business process, the steps of a use case, or the logic of an object behavior (method). Therefore, one activity diagram can be constructed for each use case. More than one can be constructed if the use case is long or contains complex logic. System analysts use activity diagrams to better understand the flow and sequencing of the use-case steps. (Whitten, 2007).

In the activity diagram process, several notations need to be considered list as flows:

1. ***Initial node***: the solid circle representing the start of the process.
2. ***Actions:*** the rounded rectangles representing individual steps. The sequence of actions makes up the total activity shown by the diagram.
3. ***Flow***: the arrows on the diagram indicating the progression through the actions. Most flows do not need words to identify them unless coming out of decisions.
4. ***Decision***: the diamond shapes with one flow coming in and two or more flows going out. The flows coming out are marked to indicate the conditions.
5. ***Merge*** the diamond shapes with two or more flows coming, in and one flow going out. This combines flows that were previously separated by decisions. Processing continues with anyone flow coming into the merge.

6. ***Fork:*** a black bar with one flow coming in and two or more flows going out. Actions on parallel flow beneath the fork can occur in any order or concurrently.
7. ***Join:*** a black bar with two or more flows coming in and one flow going out, noting the end of concurrent processing. All actions coming into the join must be completed before processing continues.
8. ***Activity Final:*** the solid circle inside the hollow circle representing the end of the process.
9. ***Sub activity Indicator:*** the rake symbol an action indicates that this action is broken out in another separate activity diagram. This helps you keep the activity diagram from becoming overly complex.
10. ***Connector:*** A letter inside a circle gives you another tool for managing complexity. A flow coming into a connector jumps to the flow coming out of a connector with a marching letter.

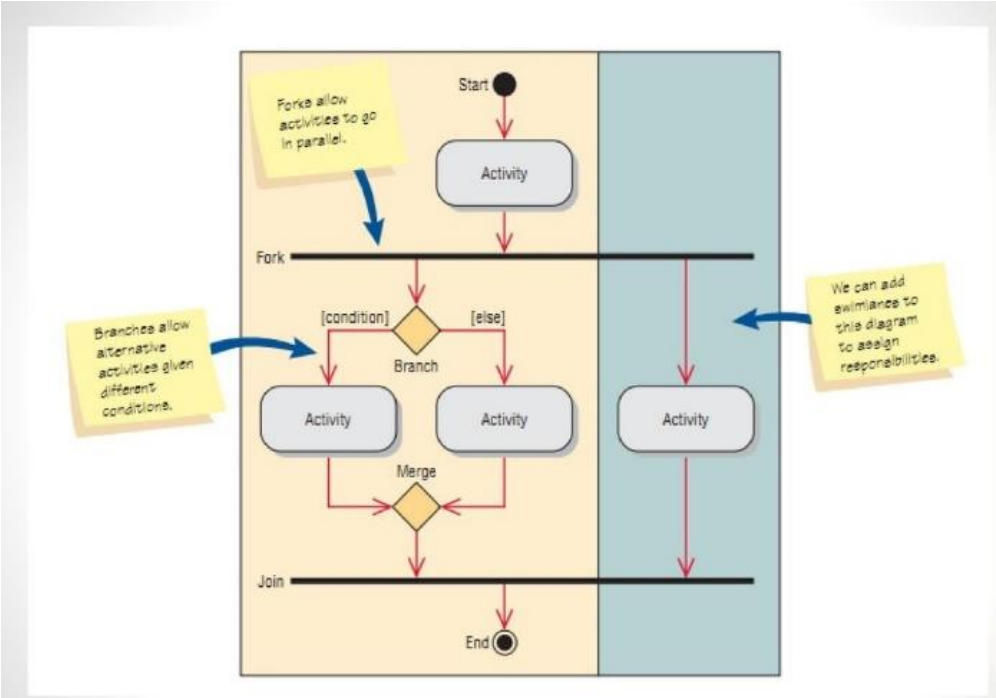


Figure 2. 5 Specialized symbols are used to draw an activity diagram. (Kendall & Kendall, 2014).

3. Class Diagram

Object-oriented methodologies work to discover classes, attributes, methods, and relationships between classes. Because programming occurs at the class level, defining classes is one of the most important object-oriented analysis tasks. Class diagrams show the static features of the system and do not represent any particular processing. A class diagram also shows the nature of the relationships between classes. Therefore, a class diagram may show just the class name; or the class name and attributes; or the class name, attributes, and methods. Showing only the class name is useful when the diagram is very

complex and includes many classes. If the diagram is simpler, attributes and methods maybe include. (Kendall & Kendall, 2008).

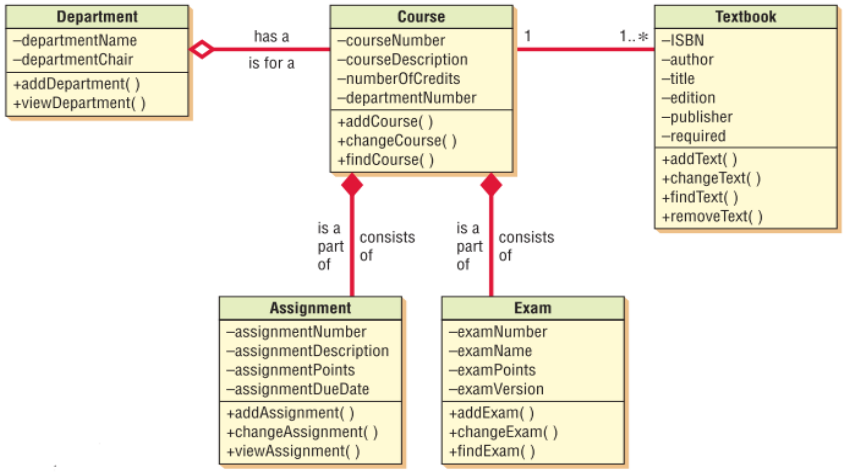


Figure 2. 6 A class diagram for course offerings. (Kendall & Kendall, 2008)

4. Sequence Diagrams

A diagram that depicts the interaction between an actor and the system for a use case scenario. In the execution of a use case or operation. We have not yet started analyzing the individual object classes; that will come next as we build our version of the class diagram. Further, the system sequence diagram does not include any of the alternative courses of the use case. (Whitten, 2007).

The sequence diagram has 5 main important keys as follows:

1. ***Actor:*** The initiating actor of the use case is shown with the use case actor symbol.

2. ***System:*** the box indicates the system as a "black box" or as a whole. The colon (:) is standard sequence diagram notation to indicate a running '" hesitance" of the system.
3. ***Lifelines:*** the dashed vertical lines extending downward from the actor and system symbols, which indicate the life of the sequence.
4. ***Activation bars:*** the bars that are set over the lifelines indicate the time when the participant is active in the interaction. Some methodologists leave them off the system sequence diagram, but we have included them to be consistent with the full sequence diagram.
5. ***Input Messages:*** horizontal arrows from the actor to the system Indicate d1e message inputs. The UML convention for messages is to begh1 the first word with a lowercase letter and appends additional words with an initial uppercase letter and no space. In patend1e.ses include any parameters that you know at this point, following the same naming convention and separating individual. (Whitten, 2007).

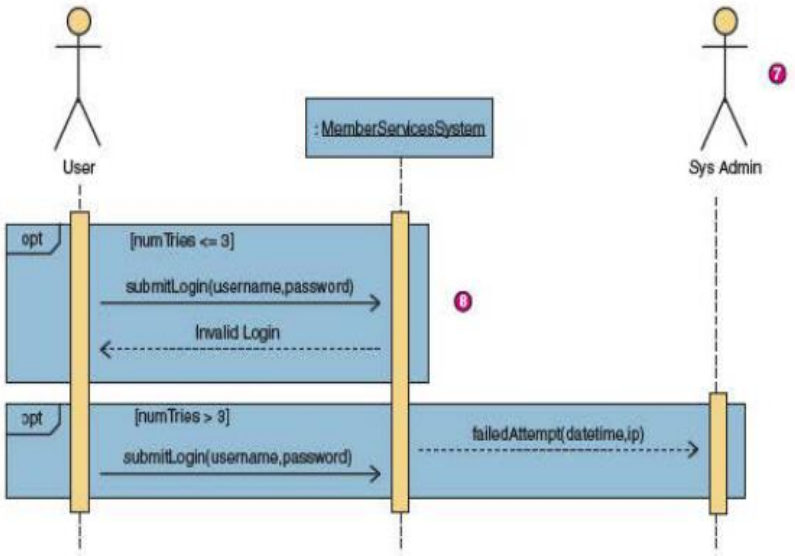


Figure 2. 7 System Sequence Diagram for Login Validation (Whitten, 2007).

2.11 Framework

A web application framework is a bunch of source code organized into a certain architecture that can be used for rapid development of web applications. You can think of frameworks as half-produced applications that you can extend and form to make them take shape according to your needs. Well, that means half of your work has already been done, but for some, it is as much a blessing as a curse because this work was done in a particular way, without your supervision. (Bartosz Porebski, Karol Przystalski, 2011).

A. Design Pattern

It is a general solution to a commonly occurring problem in software design. There is not much more formal foundation because design patterns are a generally practical means that make up for a lack of informal mechanisms. Therefore, if you want another programmer to add some functionality to a fixed class and then tell him to use the Decorator pattern, you can expect that it will be done the way you want it rather than with a random makeshift solution. Thus design patterns have great potential for preventing future problems.

B. Main Structural of Design Pattern (MVC)

Web frameworks take advantage of most, if not all, design patterns. However, MVC is the absolute structural backbone of all frameworks. The main idea of MVC is dividing the application into three layers:

1. **Model:** Represents the business logic of the application. It is more than just the raw data; the Model has to represent the structure of data with all relationships and dependencies. It may comprise one or more classes that correspond to logic objects of the application and provide an interface for manipulating them. The Model is the only layer that uses persistent storage. It should completely encapsulate all database connections. The model should also notify the View when its internal state changes, so the View can be refreshed.
2. **View:** The output displayed to the user. The most important thing is that the View never modifies the application data; it only presents it. There may be

multiple Views for the same data, such as traditional HTML, PDF, Flash, or WML for mobile devices. They should be interchangeable without modifying the other layers.

3. **Controller:** The part of an application responsible for handling user interaction and taking all other actions. The Controller should be created with simplicity in mind—it should be the controlling part that uses methods provided by the Model and the View; it shouldn't do everything by itself. (Bartosz Porebski, Karol Przystalski, 2011).

Image source here.

2.12 Laravel

Laravel is an MVC web-development framework written in PHP. It has been designed to improve the quality of your software by reducing both the cost of initial development and ongoing maintenance costs and to improve the experience of working with your applications by providing clear expressive syntax and a core set of functionality that will save you hours of implementation time.

Besides that, Laravel is one of the few PHP frameworks that offer true code modularity. This is achieved through a combination of drivers and its bundle system. Drivers allow you to easily change and extend caching, session, database, and authentication functionality. Using bundles, you're able to package up any kind of code for either your re-use or to provide to the rest of the Laravel community. (McCool, 2012).

2.13 COMPOSER

A composer is a tool for dependency management in PHP. It allows you to declare the libraries your project depends on and it will manage (install/update) them for you. Using composer, you can include third party ready-to-use packages and libraries in your Laravel project and manage them all in one place using composer.json file. The composer.json file is a plain JSON file placed under the project root directory, which defines the metadata for all project package dependencies.

Laravel is packaged with out of the box dependency management Composer and Packages. Using composer, you can include third party ready-to-use packages and libraries in your Laravel project and manage them all in one place.

2.14 WAMP

WAMP Stands for "Windows, Apache, MySQL, and PHP." WAMP is a variation of LAMP for Windows systems and is often installed as a software bundle (Apache, MySQL, and PHP). It is often used for web development and internal testing, but may also be used to serve live websites. The most important part of the WAMP package is Apache (or "Apache HTTP Server") which is used to run the webserver within Windows. By running a local Apache webserver on a Windows machine, a web developer can test webpages in a web browser without publishing them live on the Internet. WAMP also includes MySQL and PHP, which are two of the most common technologies used for creating dynamic websites. MySQL is a high-speed database, while PHP is a scripting language that can be used to access data from the database. By installing these two components locally, a developer can

build and test a dynamic website before publishing it to a public webserver. While Apache, MySQL, and PHP are open source components that can be installed individually, they are usually installed together. One popular package is called "Wamp Server," which provides a user-friendly way to install and configure the "AMP" components on Windows.

NOTE: The "P" in WAMP can also stand for either Perl or Python, which are other scripting languages. The Mac version of LAMP is known as MAMP. (Bartosz Porebski, Karol Przystalski, 2011).

2.15 MySQL

Used by millions of developers, MySQL is the most popular Open Source database, supporting numerous large dynamic websites and applications. MySQL has acquired this wide popularity by its open-source nature, reliability, robustness, and support for various platforms. This popularity has also been aided by the existence of phpMyAdmin, the industry-standard administration tool that makes database management easy for both the experienced developer and novice. The powerful graphical interface that it provides to MySQL has made phpMyAdmin an indispensable tool for MySQL and Web developers.

MySQL is the world's most popular open-source database. Whether you are a fast-growing web property, technology ISV, or large enterprise, MySQL can cost-effectively help you deliver high performance, scalable database applications.

https://www.mysql.com/products

2.16 PHP

PHP, which stands for "PHP: Hypertext Preprocessor" is a widely-used Open Source general-purpose scripting language that is especially suited for web development and can be embedded into HTML. Its syntax draws upon C, Java, and Perl, and is easy to learn. The main goal of the language is to allow web developers to write dynamically generated web pages quickly, but you can do much more with PHP.

https://www.php.net/manual/en/preface.php

2.17 HTML

HTML is the language in which most websites are written. HTML is used to create pages and make them functional. Furthermore, A Markup Language is a way that computers speak to each other to control how text is processed and presented. To do this HTML uses two things: tags and attributes.

2.18 PHPMYADMIN

PhpMyAdmin is a web application written in PHP and contains–like most web applications–XHTML, CSS, and JavaScript client code. It provides a complete web interface to administering MySQL databases and is widely recognized as the leading application in this field.

Being open-source since the start of its existence, it has enjoyed support from numerous developers and translators worldwide (being translated into 50 languages at the time of going to press). The project is currently hosted on SourceForge.

Host providers everywhere have shown their trust in phpMyAdmin (official home page at http://www.phpmyadmin.net) by installing it on their servers. Also, we can install our copy of phpMyAdmin inside our web space, as long as our provider has installed the minimum PHP version required by phpMyAdmin, which is currently PHP 4.1.0 with session support. Moreover, the webserver must have access to a MySQL server (version 3.23.32 or later)–either locally or on a remote machine. The popular Cpanel (a website control application) interfaces with phpMyAdmin.

CHAPTER III

RESEARCH METHODOLOGY

3.1 Data Collection Method

There are several approaches to data collection depending on the nature of the research being conducted. In this research, the methods adopted include the following: Interview, references to the published collection. The data collected for this research can be broadly classified into two categories, namely: the primary and secondary data.

3.1.1 Primary Data

Primary data can be defined as data collected directly from the respondent relevant to the subject under investigation. The primary data used in this case is the interview and the questionnaire method according to (Dennis et al., 2015) says that primary source data collection is the source from firsthand information that can be obtained. There are several tools for gathering the primary source of data collection include; interview, observation, and questionnaire, etc.

Interview:

The researcher performs the interview with the principal of Latrikunda lower and upper basic school. Mr. Ousman Keita the way of question and answer

section to ensure that the researcher obtained the relevant information for the proposed academic information system (AIS).

Observation:

Observation is done by telephone conversation with the school located at Latrikunda Sabiji. This indirect observation was carried out under the supervision of Mrs. Fatima Jamba the assistant principal of Latrikunda lower and upper basic school. The observation was conduct to see how the existing system of the school been working through and the process of registration of students until then end of the process. Therefore, the existing system has not been computerized and has not been integrated with any database system, so the process took more time than expected.

Based on the above information's the researcher gather particular accurate information about the school list is below:

1. The type of system that is running at the school.
2. Description of the school structure and job description for all the positions of the employees.
3. Types of students enroll in the school.
4. The way of registering students in the school.

Questionnaire

Following finding accurate data from the students, teachers, parents the researcher use one of the data collection techniques which are questionnaire, to help the researcher to find the accurate data of the school.

3.1.2 Secondary Sources

These are the source of data collection in which an already made data are been obtained i.e. that information that is already in printed form. Sources of secondary data include textbooks, magazines, journals, etc. in the case of this project, most of the data are published, documents, and references.

Literature Studies

In term of, the literature studies were done by the researcher is to study various theories that related to the topic of researching to support the researchers in preparing reports to be discussed, researchers use various books to able to get related topics, such as Unified Modelling Language (UML), and Rapid Application Developer (RAD) which is used to design the entire system. Therefore, researchers also use references on journals related papers and internal sites which are related to the “development of academic information system for Latrikunda upper basic school in the Gambia” with various types of frameworks and programming languages.

Besides that table **Figure 3.1** below is the literature studies that the researcher uses to conduct the study.

Table 3. 1 Similar Researches

No	Title	Description	Year
1	*A comprehensive definition of technology from an ethological perspective*	Ethological perspective inquires how humans use technology, the use of technology in human beings' life. How technology is used several as a framework. Humankind creating it, or a significant beneficiary of rationally derived knowledge that is "used for" a purpose without itself necessarily being translated into something material that "does" autonomously or dependently when used. Knowledge of technology allows one to distinguish between what is natural (i.e., natural technology) and what is made by humans (i.e., technology made by man).	**2017**
2	*Technology-based management of environmental organizations using an Environmental Management Information System (EMIS): Design and development*	An information system that provides information used by human resource management in decision making. The system to provide information about the staff of the organization, helping the human resource management and in project management, such as education, curriculum vitae, residence, contact information, hiring date (hiring date, salary, position, etc.) and information regarding user's registration to the system. Also, the system facilitates projects by monitoring the implementation and providing information about assigned environmental projects.	**2016**
3	*Implementation of information systems as an organizational construction*	The implementation of an information system is seen to be a gradual organizational process, where learning an innovation step taken by the applying organization and its different actors play a crucial role in the success of the implementation process. However, the implementation and knowledge perspective brings into view the special characteristics of information systems. Information and knowledge perspectives together make a clear difference between information systems and information technology. Information and knowledge aspects can be seen to mark the organizational need to manage the use of information technology concerning an organization's activities and intentions.	**2013**
4	*knowledge management in organizations*	Firstly, knowledge is now the most important and valuable resource in advanced industrial economies. Secondly, knowledge represents the most important economic asset that business organizations possess and that it is the prime determinant of their innovativeness and profitability. Finally, the nature of paid employment and business organizations is changing, with an enormous growth in the number of knowledge workers, and knowledge-intensive organizations	**2018**

Table 3. 2 Similar Researches (*continue*d)

No	Title	Description	Year
5	*Systems Analysis Design with UML Version 2.5: An Object-Oriented Approach*	Systems Analysis and Design (SAD) is an exciting, active field in which analysts continually learn new techniques and approaches to develop systems more effectively and efficiently. All information systems projects move through the four phases of planning, analysis, design, and implementation; all projects require analysts to gather requirements, model the business needs, and create blueprints for how the system should be built; and all projects require an understanding of organizational behavior concepts like change management and team building. Unified Modeling Language (UML). UML provides a common vocabulary of object-oriented terms and diagramming techniques that are rich enough to model any systems development project from analysis through implementation.	**2015**
6	*Computerized Management Information Systems Resources and their Relationship to the Development of Performance in the Electricity Distribution Company in Gaza*	-Computerized management information systems allowed a great opportunity for various business organizations in strengthening their competitiveness and achieve their desired objectives, - Computerized management information systems lead to develop and improve the performance of employees because of their relationship with the professional development of functional employees.	**2016**
9	Academic Information System for Student (Case Study: Victory University of Sorong)	- The problem that arises is a complicated process because of the excessive use of time and paper, causing overhead to the students and the University. - Educational institutions should be able to use and utilize information technology as a supporter of operational activities in producing accurate information.	**2018**

3.2 Systems Development Method

In this thesis, the researcher takes several stages which carried out with rapid application design (RAD) (Kendall & Kendall, 2008)

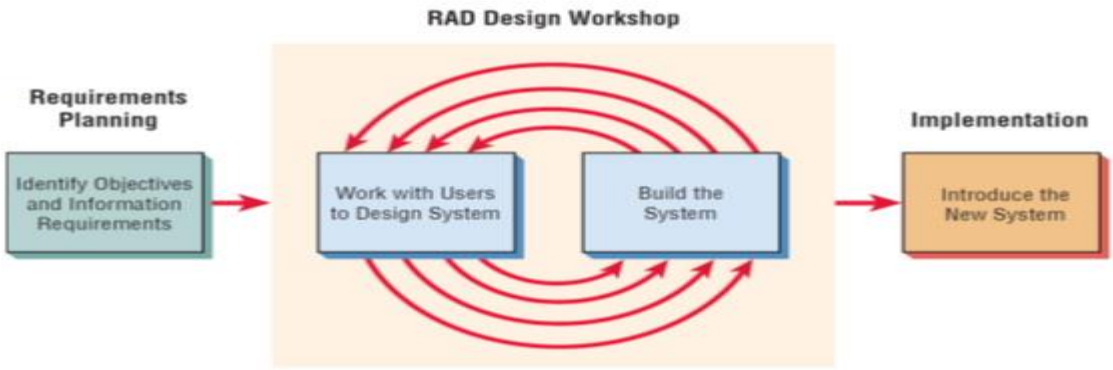


Figure 3. 1 The RAD design workshop. (Kendall & Kendall, 2008)

3.2.1 Requirement Planning Phase

In this phase, the researcher set with the board of LUBS for identify the purpose of the system and identify information needs that arise from these objectives. During the meeting a plan was made in the AIS, namely:

1. The proposed System development goals
2. Analysis of the proposed system requirements
3. Problem definition of the existing system.

3.2.2 RAD Design Workshop Phase

3.2.2.1 Process

During the process design stage, the tools used to design the diagrams are unified modeling language (UML). There are several tools included in UML but the researcher chooses few UML tools to implement the design process. Therefore the diagrams that support in this application are as follows:

1. Use Case Diagrams, during this stage the researcher describes the use case diagrams that explain the activities carried out by the proposed system, which will be built and who will be the actors to interact with this AIS system.

    a. During the (Actor Identification) stage the researcher identifies the actors involved in the proposed system and what are their roles in the system.
    b. During the (Use Case Designing) stage the researcher descript the relationship between the actors and the use cases how they will interact with each other in the system.
    c. During the (Use case Narrative) stage the researcher explains the use cases, which are the features contained in the system. Each use case is shown as an oval and named as a process with a verb to describe the action in the AIS system.

2. Activity Diagrams, during the design stage of the diagram, illustrate the various activities that are carried out by users and systems in the proposed system, as well as how each flow starts until how they end.

3.2.2.2 Database Design

During the database design, the researcher design a database that will be used in AIS by using Class Diagrams, ERP Diagrams which is useful for optimizing databases and Sequence Diagrams.

3.2.2.3 Interface Design

During the interface design stage, the researcher designs the interface that suits the needs of users so that the AIS system can be user friendly and used optimally by its users.

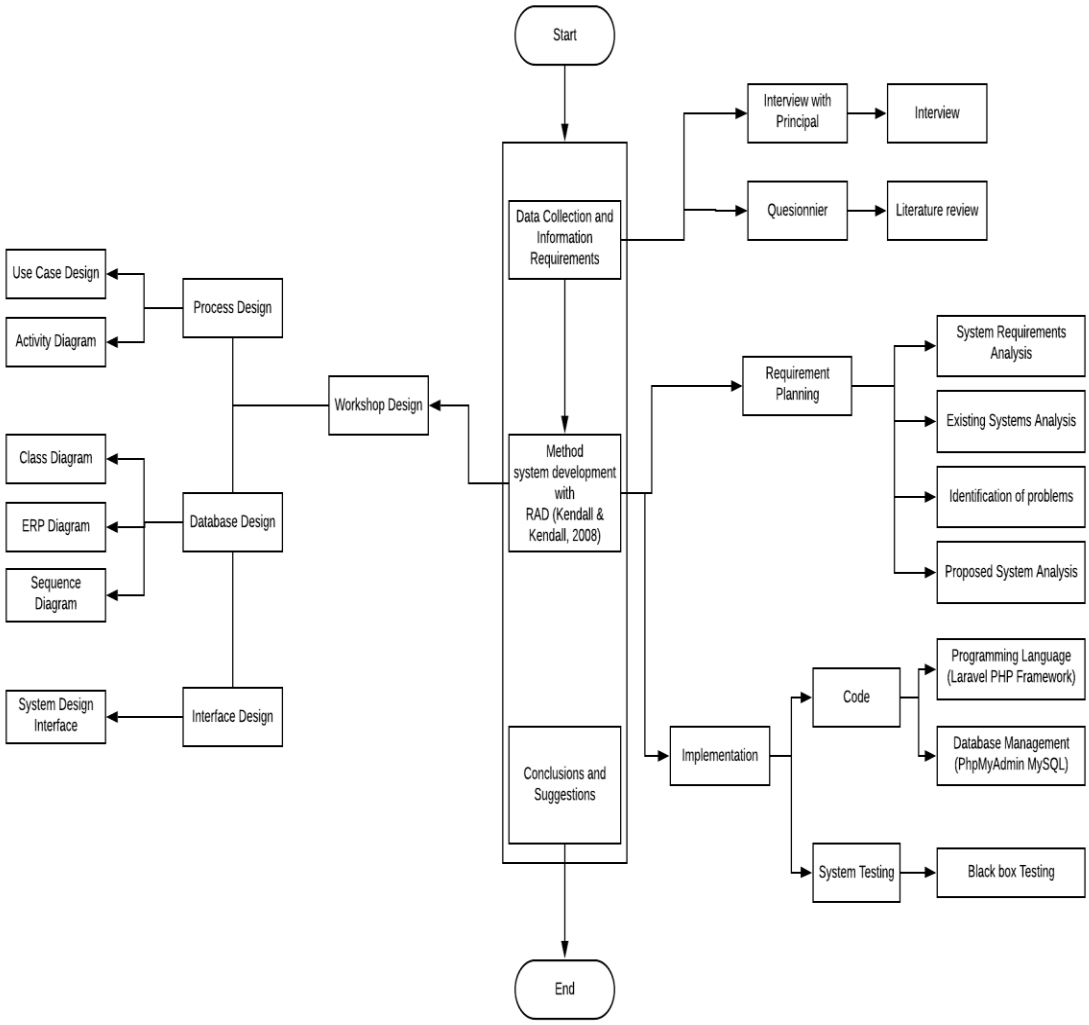
3.2.2.4 Implementation Phase

The implementation phase stage consists of two main crucial stages, namely:

1. The system into the programming language (coding) stage
2. The system testing stage using black-box testing.

   With the completion of this stage, the design process ends AIS on LUBS in the Gambia.

3.3 Research Flow



The implementation of the activities in this research is stated in the mindset that can be seen in the following flowchart:

Figure 3. 2 Research Flow

CHAPTER IV
SYSTEM DESIGN AND DEVELOPMENT

4.1 Institute Profile

Latrikunda Sabiji Lower and Upper Basic School is the educational and academicals institute that has established since 1972 under the authority approval licensed by the Ministry for Basic and Secondary Education (MoBSE); the institute is located in Latrikunda Sabiji, Serrekunda, The Gambia, especially Latrikunda Sabiji Institute has considered and pay his best effort not only for kids in school but also kid's parents have time to hear and know the salvation of institute by maintaining the kids' education quality both believers and non-believers.

Firstly, Latrikunda Sabiji Lower and Upper Basic School have provided one language that is (English Language) for Primary, (Students age: from 7-13 years old). The Institute has become independent and survived with the help of the students from one to one year, in the need of society to develop the country; therefore, education is the most important sector.

Secondly, Latrikunda Sabiji Lower and Upper Basic School have been in progress for updated the academic curriculum since July 1, 2011, because of the need of the kid's parents; Latrikunda Sabiji Lower Basic School has effectively resulted in upgrading from Lower Basic School to Upper Basic School that consisted 3 grades from 7-9 and 6 classes from A-F successfully that provided 2 languages (French, and

English) in the institute curriculum. In 1999, Latrikunda LBS granted and enlarged their classes by adding more classes which consisted of many students who have enrolled.

4.1.1 Vision:

The school shall endeavor to provide three years of basic relevant and quality education for all, provide the highest intellectual and moral development of the pupils strive for self-reliance adjust every child to his/ her environment, provide a vision of greatness, train democratic citizen, lay solid foundation and leeway to pursue higher education's enshrined in education policy hence vision 2020.

4.1.2 Mission:

To pedagogically nurture pupils with the required knowledge necessary skills, concepts, positive attitudes norms, and values, thus helping them to learn how to learn and work amicably with others, inculcate a high sense discipline, self-respect for authority. Participate in intra and inter-schools competition; strive towards the retention and completion of three years of U.B.S Education.

4.1.3 Motor

Discipline, Hard work, Excellent

4.2 System Inquiry

This is an in-depth and comprehensive study carried out upon an existing system to arrive at vital and relevant facts that will help/assist in the design and implementation of the improved /new system or change which will be brought by the proposed system. The main objective of system investigation is to find out or learn how the current system is operating so as the surface and come out with relevant data. The case organization will be properly studied based on its operational mode. Books and records kept and approach to a decision. It will involve the presentation and analysis of data based on the system (Latrikunda Lower and Upper Basic). The main system of the school was not integrated with any database system which will help the school to collaborate with their students and teachers, and also the parents can also able to access the system while checking if their child has been to school or not, that will help the entire school to organize and build a great structure of the environment. Therefore, students and teachers also can able to interact and able to share ideas in the forum.

The researcher took time in studies and the academic operation of (Latrikunda Lower and Upper Basic) which extracts course schedule study and examination time table. Record and studying the unit was also studied. Data were collected from the workers and students which will form the basis for data analysis as below.

4.3 System Analysis

The existing system was analyzed to improve efficiency and studying the specification of the requirements is very essential in meeting the set objectives. For the development of the new system, a preliminary survey of the existing system was conducted and investigated and the prototyping model was adopted.

4.4 Existing System

Today in Latrikunda lower and upper basic schools, students, teachers, and non-staff details are entered manually. These record details are tedious tasks. Referring to all these records and updating is needed. There is a high chance of more manual errors. Others include:

1. Manual course registration tends to be slow and inconsistent.
2. It doesn't provide a faster and effective system.
3. It doesn't provide good coordination between departments.
4. Scheduling and releasing academic calendar pose a challenge, as well as a lot of inconsistencies and clashes, which are always experienced.
5. It doesn't provide an effective forwarding system to move the file from one level to another.
6. Difficulty in generating different reports as per the school requirement.
7. It doesn't facilitate the services online.

4.5 Proposed System

The Academic Information system is an automated system with web-based architecture that can do the following and not limited to:

1. Maintains information related to different departments and stored at a central database, which leads to easy accessibility and consistency.
2. Create different user access with relevant functionality.
3. Register students, teachers, and non-staff.
4. Students' payment history and other details are also available at the click of a mouse.
5. Teachers and non-staff salaries can be monitor effortlessly online.
6. Assign students to different clubs and manage them.
7. The decision process faster and more consistent.
8. It provides good communication between the two departments.
9. It provides a facility to generate reports very easily.
10. Monitors the teacher's attendance.
11. Generate print out the receipt.
12. Provides a summary of finance generated from tuition fees and others.
13. Cost reduction.

4.6 System Design

System design is the solution to the creation of a new system. This phase is composed of several systems. This phase focuses on the detailed implementation of the feasible system. It emphasized on translating design specifications to performance specification. System design has two phases of development, logical and physical design. During the logical design phase

the inputs (sources), outputs (destinations), databases (data stores), and procedures (data flows) are all described in a format that meets the user requirements.

1. The logical design is followed by physical design or coding. The physical design produces the working system by defining the design specifications, which help immensely during development. The necessary programs that accept input from the user perform necessary processing on accepted data is written to produce the required report on a hard copy or display it on the screen.

If the project is to be successful, we will need to answer these questions. The answer to these questions is schema manner and is known as system design.

2. A systematic manner will be followed to achieve beneficial results in the end. It involves starting with a vague idea and ultimately developing it up into a useful system. The design phase is the transition from a user-oriented to a document-oriented to the programmers. Software reports can be broken into a series of steps starting with the basic ideas and ending with the finished project.

The researcher used three types of modeling diagrams in the project, there as follows:

Use case diagram, sequence diagram, and class diagram.

4.6.1 Input Design

The input design is the link between the system and the user. It comprises the developing specification and procedures for data preparation and those steps are necessary to get a usable form for processing data entry. The activity of putting data into the computer for processing can be achieved by inspecting the computer to read data from a written or printed document or it can occur by having people keying the data directly into the system. The design of input focuses on controlling the amount of input required, controlling errors, avoiding delay, avoiding extra steps, and keeping the process simple. The error raising method is also included in the software, which helps to raise error message while the wrong entry of input is done. So, in input design, the following things are considered

What data should be given as input?

1. How the data should be arranged or coded?
2. The dialogue to guide the operating personnel in providing input.
3. Methods for preparing input validations and steps to follow when an error occurs

4.6.2 Output Design

Computer output is the most important and direct information source for the user. Output design is a process that involves designing necessary outputs in the form of reports that should be given to the users according to the requirements. Efficient, intelligible output design should improve the system's relationship with the user and help in decision making. Since the reports are directing referred by the management for taking decisions and to draw conclusions they must be designed with almost care and

the details in the reports must be simple, descriptive, and clear to the user. So, while designing the output, the following things are considered:

1. Determine what information to present
2. Arrange the presentation of information in an acceptable format

4.6.3 Physical Design

The process of developing the program software is referred to as physical design. We have to design the process by identifying reports and the other outputs the system will produce. Coding the program for each module with its logic is performed in this step. Proper software specification is also done in this step.

4.7 Workshop Design

In this phase, the authors create system design using UML tools and of course by keeping in touch with end-users.

The flowing are the stages in system design include:

4.7.1 Use Case Design

The use case diagrams explain the interactions between actors in an AIS system.

Table 4. 1 Actor and Use Case Identification

NO.	*ACTOR*	*DESCRIPTIONS*
1	Administrator Module	This module is responsible for coordinating the other modules. It allows the administrator to set up teachers, non-staff on school payroll with their respective salaries. And it allows admin to have a global view of the school record and it can view teacher attendance and generate student report cards. It also allows the administrator to create a new student, assign a student to a department, and manages the department. It also allows the administrator to publish the school result. Etc.
2	Teacher Module	This module allows the teacher to Mark attendance, enter student marks for each subject undertake and modify it when necessary. It also views new subjects undertake. It creates marks/aggregate and grading of subjects. It also can view students and classes undertake, it also can change password, communicate with students within a message in the system. Etc.
3	Student Module	This module allows the student to view a profile, change password, view timetable select subject, print result, and print transcript, print, and view fee history, and receive notifications from the teacher.

In the following table 4.2 below will show the interaction between actors with the system.

Table 4. 2 List of Use Case Diagrams

NO.	USE CASE	DESCRIPTION	ACTORS
1	Login	This use case describes the activity of entering a username and password to be able to access the system.	Administrator, Teacher, Student
2	Add Students	This use case describes the activity of adding, editing, updating, deleting, and viewing new students to the system.	Administrator
3	Add Teachers	This use case describes the activity of adding, editing, updating, deleting, and viewing new teachers to the system.	Administrator
4	Add Shift	This use case describes the activity of adding, editing, updating, deleting, and viewing a new shift to the system.	Administrator
5	Add Class	This use case describes the activity of adding, editing, updating, deleting, and viewing new classes to the system.	Administrator

Table 4. 3 List of Use Case Diagrams (continued**)**			
	Add Subject	This use case describes the activity of adding, editing, updating, deleting, and viewing new subject to the system.	Administrator
7	Add Faculty	This use case describes the activity of adding, editing, updating, deleting, and viewing new faculty to the system.	Administrator
8	Add Department	This use case describes the activity of adding, editing, updating, deleting, and viewing new departments to the system.	Administrator
9	Add Batch	This use case describes the activity of adding, editing, updating, deleting, and viewing a new batch to the system.	Administrator
10	Add Timetable	This use case describes the activity of adding, editing, updating, deleting, and viewing a new timetable to the system.	Administrator
11	Generate Class Schedule	This use case describes the activity of adding, editing, updating, deleting, and viewing a new class schedule to the system.	Administrator
12	Add Exam	This use case describes the activity of adding, editing, updating, deleting, and viewing new exams to the system.	Administrator
13	Add Semester	This use case describes the activity of adding, editing, updating, deleting, and viewing new semesters to the system.	Administrator
14	Add Degree	This use case describes the activity of adding, editing, updating, deleting, and viewing a new degree to the system.	Administrator
15	Add Exam	This use case describes the activity of adding, editing, updating, deleting, and viewing new exams to the system.	Administrator
16	View Exam	This use case describes the activity of viewing exam to the system.	Teacher, Student
17	Handle Class	This use case describes the activity of handling classes to the system.	Teacher
18	Add Exam Marks	This use case describes the activity of adding, editing, updating, deleting, and viewing new exam marks to the system.	Teacher
19	View Exam Marks	This use case describes the activity of viewing exam marks to the system	Teacher, Student
20	View Timetable	This use case describes the activity of viewing timetables in the system.	Teacher, Student
21	View Profile	This use case describes the activity of viewing profile to the system.	Administrator, Teacher, Student
22	Change Password	This use case describes the activity of updating the password to the system.	Teacher, Student
23	Print Result	This use case describes the activity of printing results in the system.	Teacher, Student

24	Print Transcript	This use case describes the activity of the printing transcript to the system.	Teacher, Student

In Figure 4.1 the following shows the use case diagram as an illustration of the explanation in table 4.2 above.

USE CASE DIAGRAM FIGURE

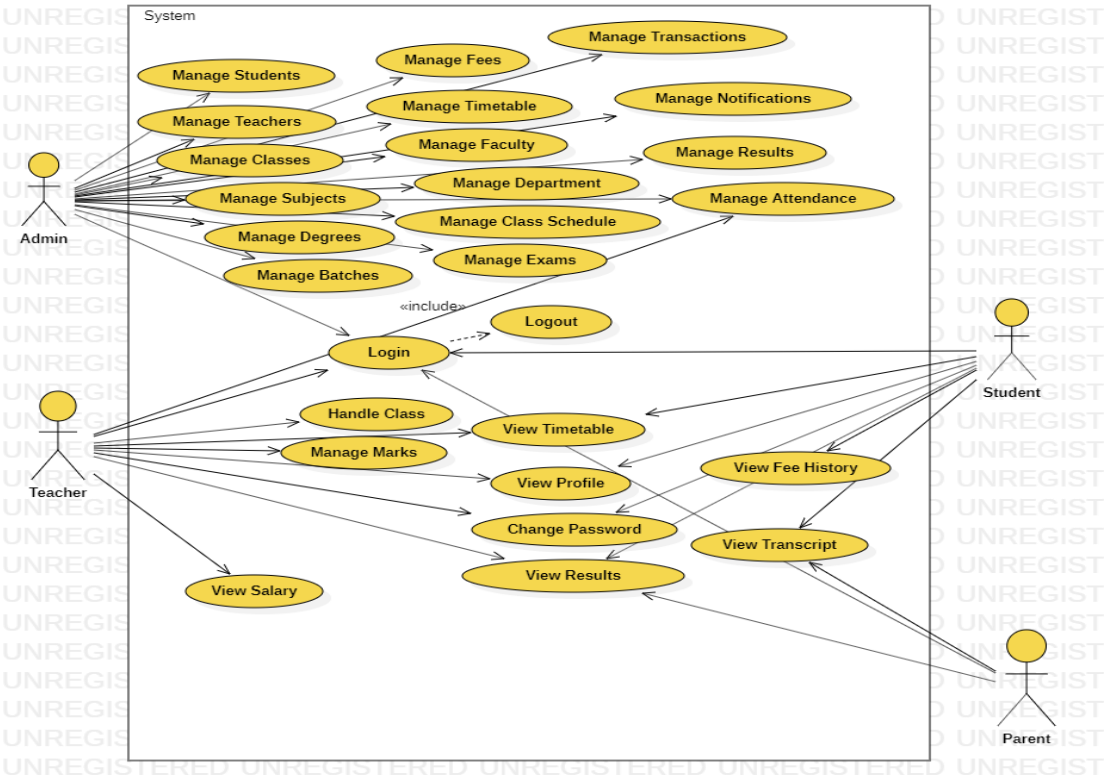


Figure 4. 1 Use Case Diagram

In the use case diagram, there are 4 are actors which can able to interact with the system namely, Admin as (Administrator), Teacher as (Staff), Student and Parents.

1. The first actor is the admin after logging has the privileges of managing the entire system.

2. The second actor is the teacher after logging has the privileges of viewing students, classes, enter exam mark, result card undertake.
3. The third actor is the student after logging has the privileges of viewing profile, change password, print result, print transcript, view timetable, view fee history.

The process that occurs in the use case diagram Figure 4.1 further explained in detail in some of the use case diagram narratives that exist in the proposed system.

Table 4. 4 Login Use Case Scenario

Name of *Use Case*	**Login**	
Actors	Administrator, Teacher, and Student	
Description	This Use Case illustrates the activity of inputting a username and password to have access to the system.	
Precondition Field	The actors need to enter their username and password to get access to the system.	
The typical field of an event	**Activity of actors**	**System Response**
	1. Entering Username and password	2. Check if the credentials are correct to access. 3. Direct to the landing page of each user.
Alternative fields	Alt- step 2. If the username and password are correct, then direct the user to the system. If wrong, the system will automatically display the login menu again so that the actor will fill in the username and password.	
Conclusion	Actor Finally entered into the system.	
Postcondition	System successfully accessed	

Table 4. 5 "Manage Students" Use Case Scenario

Name of *Use Case*	*Manage Students*	
Actor	Administrator	
Description	This Use Case illustrates the activity of registering a new student into the system.	
Precondition Field	The actor will perform CRUD operation for registering a new student data.	
The typical field of an event	**Activity of actor**	**System Response**
	1. Entering Username and Password. 3. Select Students Menu. 5. Input Student Details into the form and submit	2. Direct to the Dashboard 4. Select Add New Student Form. 6. System Check Required Inputs. 7. The System will Save the New Student Data by the Actor.
Alternative fields	At- step 3. if the actor leaves any of the required input fields the system will alert an error message and if the actor clicks the cancel button it will return to the Add New Student Form.	
Conclusion	Actor successfully Save new student data	
Postcondition	A new student has been added	

Table 4. 6 "Manage Teacher" Use Case Scenario

Name of *Use Case*	*Manage Teacher*	
Actor	Administrator	
Description	This Use Case illustrates the activity of registering a new teacher into the system.	
Precondition Field	The actor will perform CRUD operation for registering new teacher data.	
The typical field of an event	**Activity of actor**	**System Response**
	1. Entering Username and Password. 3. Select Students Menu. 5. Input Teacher detail into the form and submit	2. Direct to the Dashboard 4. Select Add New Teacher Form. 6. System Check Required Inputs.

		7. The System will Save the New Teacher Data by the Actor.
Alternative fields	At- step 3. If the actor leaves any of the required input fields the system will alert an error message and if the actor clicks the cancel button it will return to the Add New Teacher Form.	
Conclusion	Actor successfully Save new teacher data	
Postcondition	A new teacher has been added.	

Table 4. 7 "Manage Shift" Use Case Scenario

Name of *Use Case*	*Manage Shift*	
Actor	Administrator	
Description	This Use Case illustrates the activity of registering a new shift into the system.	
Precondition Field	The actor will perform CRUD operation for registering a new shift data.	
The typical field of an event	**Activity of actor**	**System Response**
	1. Entering Username and Password. 3. Select Shift Menu. 5. Input Shift detail into the form and submit.	2. Direct to the Dashboard 4. Select Add New Shift Form. 6. System Check Required Inputs. 7. The System will Save the New Shift Data by the Actor.
Alternative fields	At- step 3. If the actor leaves any of the required input fields the system will alert an error message and if the actor clicks the cancel button it will return to the Add New Shift Form.	
Conclusion	Actor successfully Save new shift data	
Postcondition	A new shift has been added.	

Table 4. 8 "Manage Class" Use Case Scenario

Name of *Use Case*	*Manage Class*
Actor	Administrator
Description	This Use Case illustrates the activity of registering a new class into the system.

Table 4. 9 "Manage Class" Use Case Scenario (Continued**)**		
Precondition Field	The actor will perform CRUD operation for registering new class data.	
The typical field of an event	**Activity of actor**	**System Response**
	2. Entering Username and Password. 4. Select Class Menu. 6. Input Class detail into the form and submit.	3. Direct to the Dashboard 5. Select Add New Class Form. 8. System Check Required Inputs. 9. The System will Save the New Class Data by the Actor.
Alternative fields	At- step 3. if the actor leaves any of the required input fields the system will alert an error message and if the actor clicks the cancel button it will return to the Add New Class Form.	
Conclusion	Actor successfully Save new class data	
Postcondition	The new class has been added.	

Table 4. 10 "Manage Subject" Use Case Scenario

Name of *Use Case*	*Manage Subject*	
Actor	Administrator	
Description	This Use Case illustrates the activity of registering a new subject into the system.	
Precondition Field	The actor will perform CRUD operation for registering new subject data.	
The typical field of an event	**Activity of actor**	**System Response**
	3. Entering Username and Password. 5. Select Subject Menu. 7. Input Subject detail into the form and submit.	4. Direct to the Dashboard 6. Select Add New Subject Form. 10. System Check Required Inputs. 11. The System will Save the New Subject Data by the Actor.
Alternative fields	At- step 3. If the actor leaves any of the required input fields the system will alert an error message and if the actor clicks the cancel button it will return to the Add New Subject Form.	
Conclusion	Actor successfully Save new subject data	
Postcondition	A new subject has been added.	

Table 4. 11 "Manage Faculty" Use Case Scenario

Name of *Use Case*	*Manage Faculty*	
Actor	Administrator	
Description	This Use Case illustrates the activity of registering a new faculty into the system.	
Precondition Field	The actor will perform CRUD operation for registering new faculty data.	
The typical field of an event	**Activity of actor**	**System Response**
	4. Entering Username and Password. 6. Select the Faculty Menu. 8. Input faculty detail into the form and submit.	5. Direct to the Dashboard 7. Select Add New Faculty Form. 12. System Check Required Inputs. 13. The System will Save the New Faculty Data by the Actor.
Alternative fields	At- step 3. if the actor leaves any of the required input fields the system will alert an error message and if the actor clicks the cancel button it will return to the Add New Faculty Form.	
Conclusion	Actor successfully Save new faculty data	
Postcondition	New faculty has been added.	

Table 4. 12 "Manage Department" Use Case Scenario

Name of *Use Case*	*Manage Department*	
Actor	Administrator	
Description	This Use Case illustrates the activity of registering a new department into the system.	
Precondition Field	The actor will perform CRUD operation for registering new department data.	
The typical field of an event	**Activity of actor**	**System Response**
	5. Entering Username and Password. 7. Select Department Menu. 9. Input Department details into the form and submit.	6. Direct to the Dashboard 8. Select Add New Department Form. 14. System Check Required Inputs. 15. The System will Save the New Department Data by the Actor.

Table 4. 13 "Manage Department" Use Case Scenario (Continued)	
Alternative fields	At- step 3. if the actor leaves any of the required input fields the system will alert an error message and if the actor clicks the cancel button it will return to the Add New Department Form.
Conclusion	Actor successfully Save new department data
Postcondition	The new department has been added.

Table 4. 14 "Manage Batch" Use Case Scenario

Name of *Use Case*	*Manage Batch*	
Actor	Administrator	
Description	This Use Case illustrates the activity of registering a new batch into the system.	
Precondition Field	The actor will perform CRUD operation for registering a new batch data.	
The typical field of an event	**Activity of actor**	**System Response**
	6. Entering Username and Password. 8. Select Batch Menu. 10. Input Batch detail into the form and submit.	7. Direct to the Dashboard 9. Select Add New Batch Form. 16. System Check Required Inputs. 17. The System will Save the New Batch Data by the Actor.
Alternative fields	At- step 3. if the actor leaves any of the required input fields the system will alert an error message and if the actor clicks the cancel button it will return to the Add New Batch Form.	
Conclusion	Actor successfully Save new batch data	
Postcondition	A new batch has been added.	

Table 4. 15 "Manage Semester" Use Case Scenario

Name of *Use Case*	*Manage Semester*	
Actor	Administrator	
Description	This Use Case illustrates the activity of registering a new semester into the system.	
Precondition Field	The actor will perform CRUD operation for registering a new semester data.	
The typical field of an event	**Activity of actor**	**System Response**
	7. Entering Username and Password. 9. Select Semester Menu. 11. Input Semester detail into the form and submit.	8. Direct to the Dashboard 10. Select Add New Semester Form. 18. System Check Required Inputs.

		19. The System will Save the New Semester Data by the Actor.
Alternative fields	Alt- step 3. if the actor leaves any of the required input fields the system will alert an error message and if the actor clicks the cancel button it will return to the Add New Semester Form.	
Conclusion	Actor successfully Save new semester data	
Postcondition	A new semester has been added.	

Table 4. 16 "Manage Degree" Use Case Scenario

Name of *Use Case*	*Manage Degree*	
Actor	Administrator	
Description	This Use Case illustrates the activity of registering a new degree into the system.	
Precondition Field	The actor will perform CRUD operation for registering a new degree data.	
The typical field of an event	**Activity of actor**	**System Response**
	8. Entering Username and Password. 10. Select Degree Menu. 12. Input Degree detail into the form and submit.	9. Direct to the Dashboard 11. Select Add New Degree Form. 20. System Check Required Inputs. 21. The System will Save the New Degree Data by the Actor.
Alternative fields	Alt- step 3. if the actor leaves any of the required input fields the system will alert an error message and if the actor clicks the cancel button it will return to the Add New Degree Form.	
Conclusion	Actor successfully Save new degree data	
Postcondition	A new degree has been added.	

Table 4. 17 "Manage Class Schedule" Use Case Scenario

Name of *Use Case*	*Manage Class Schedule*	
Actor	Administrator	
Description	This Use Case illustrates the activity of generating a new class schedule into the system.	
Precondition Field	The actor will perform CRUD operation for registering a new class schedule data.	
The typical field of an event	**Activity of actor**	**System Response**

	9. Entering Username and Password. 11. Select Class Schedule Menu. 13. Input Class Schedule detail into the form and submit.	10. Direct to the Dashboard 12. Select Add New Class Schedule Form. 22. System Check Required Inputs. 23. The System will Save the New Class Schedule Data by the Actor.
Alternative fields	Alt- step 3. if the actor leaves any of the required input fields the system will alert an error message and if the actor clicks the cancel button it will return to the Add New Class Schedule Form.	
Conclusion	Actor successfully Save new class schedule data	
Postcondition	A new class schedule has been added.	

Table 4. 18 "Manage Exams" Use Case Scenario

Name of *Use Case*	*Manage Exams*	
Actor	Administrator, Teacher	
Description	This Use Case illustrates the activity of adding a new exam into the system.	
Precondition Field	The actor will perform CRUD operation for registering a new exam data.	
The typical field of an event	**Activity of actor**	**System Response**
	10. Entering Username and Password. 12. Select Exam Menu. 14. Input Exam detail into the form and submit.	11. Direct to the Dashboard 13. Select Add New Exam Form. 24. System Check Required Inputs. 25. The System will Save the New Exam Data by the Actor.
Alternative fields	Alt- step 3. if the actor leaves any of the required input fields the system will alert an error message and if the actor clicks the cancel button it will return to the Add New Exam Form.	
Conclusion	Actor successfully Save new exam data	
Postcondition	A new exam has been added.	

Table 4. 19 "View Exams" Use Case Scenario

Name of *Use Case*	*View Exams*
Actor	Administrator, Teacher, Student
Description	This Use Case illustrates the activity of viewing exam into the system.

Precondition Field	The actor will perform CRUD operation to add a new exam, view exam, edit exam, delete exam.	
The typical field of an event	**Activity of actor**	**System Response**
	11. Entering Username and Password. 13. Select View Exam Menu. 15. Input View Exam detail into the form and submit.	12. Direct to the Dashboard 14. Select View Exam Form. 26. System Check Required Inputs. 27. The System will retrieve the Exam Data by the Actors.
Alternative fields	Alt- step 3. if the actor leaves any of the required input fields the system will alert an error message and if the actor clicks the cancel button it will return to the View Exam Form.	
Conclusion	Actor successfully retrieve exam data	
Postcondition	The exam has been retrieved.	

Table 4. 20 "Manage Timetable" Use Case Scenario

Name of *Use Case*	*Manage Timetable*	
Actor	Administrator	
Description	This Use Case illustrates the activity of adding new Timetable into the system.	
Precondition Field	The actor will perform CRUD operation to add a new timetable, edit timetable, delete timetable, and view.	
The typical field of an event	**Activity of actor**	**System Response**
	12. Entering Username and Password. 14. Select Timetable Menu. 16. Input Timetable detail into the form and submit.	13. Direct to the Dashboard 15. Select Timetable Form. 28. System Check Required Inputs. 29. The System will Save the Timetable Data by the Actor.
Alternative fields	Alt- step 3. if the actor leaves any of the required input fields the system will alert an error message and if the actor clicks the cancel button it will return to the Add Timetable Form.	
Conclusion	Actor successfully save timetable data	
Postcondition	A new Timetable has been added.	

Table 4. 21 "View Timetable" Use Case Scenario

Name of *Use Case*	*View Timetable*	
Actor	Administrator, Teacher, Student	
Description	This Use Case illustrates the activity of viewing timetables into the system.	
Precondition Field	The actors will perform Read operation for retrieving timetable data.	
The typical field of an event	**Activity of actor**	**System Response**
	13. Entering Username and Password. 15. Select View Timetable Menu. 17. Input View Timetable detail into the form and submit.	14. Direct to the Dashboard 16. Select the View Timetable Form. 30. System Check Required Inputs. 31. The System will retrieve the Timetable Data by the Actors.
Conclusion	Actor successfully retrieve timetable data	
Postcondition	The timetable has been retrieved.	

Table 4. 22 "Manage Exam Marks" Use Case Scenario

Name of *Use Case*	*Manage Exam Marks*	
Actor	Administrator, Teacher	
Description	This Use Case illustrates the activity of adding a new exam mark into the system.	
Precondition Field	The actors will perform CRUD operation to add a new exam mark, edit exam marks, delete exam marks, view.	
The typical field of an event	**Activity of actor**	**System Response**
	14. Entering Username and Password. 16. Select Exam Marks Menu. 18. Input Exam Marks detail into the form and submit.	15. Direct to the Dashboard 17. Select Exam Marks Form. 32. System Check Required Inputs. 33. The System will add new Exam Marks Data by the Actors.
Alternative fields	Alt- step 3. if the actor leaves any of the required input fields the system will alert an error message and if the actor clicks the cancel button it will return to the Exam Marks Form.	
Conclusion	Actor successfully saved exam marks data	
Postcondition	Exam Marks has been added.	

Table 4. 23 "View Exam Marks" Use Case Scenario

Name of *Use Case*	*View Exam Marks*	
Actor	Administrator, Teacher, Student	
Description	This Use Case illustrates the activity of viewing exam marks into the system.	
Precondition Field	The actors will perform Read operation for retrieving an exam marks data.	
The typical field of an event	**Activity of actor**	**System Response**
	1. Entering Username and Password. 3. Select View Exam Marks Menu. 5. Input View Exam Marks detail into the form and submit.	2. Direct to the Dashboard 18. Select View Exam Marks Form. 6. System Check Required Inputs. 7. The System will retrieve the Exam Marks Data by the Actors.
Alternative fields	Alt- step 3. if the actor leaves any of the required input fields the system will alert an error message and if the actor clicks the cancel button it will return to the Exam Marks Form.	
Conclusion	Actor successfully retrieve exam marks data	
Postcondition	Exam Marks have been retrieved.	

Table 4. 24 "View Profile" Use Case Scenario

Name of *Use Case*	*View Profile*	
Actor	Administrator, Teacher, Student	
Description	This Use Case illustrates the activity of viewing profiles into the system.	
Precondition Field	The actors will perform Read operation for retrieving a view profile data.	
The typical field of an event	**Activity of actor**	**System Response**
	1. Entering Username and Password. 3. Select View Profile Menu. 5. Input View Profile detail into the form and submit.	2. Direct to the Dashboard 4. Select View Profile Form. 6. System Check Required Inputs. 7. The System will retrieve the View Profile Data by the Actors.
Alternative fields	Alt- step 3. if the actor leaves any of the required input fields the system will alert an error message and if the actor clicks the cancel button it will return to the View Profile Form.	
Conclusion	Actor successfully retrieve profile data	
Postcondition	View Profile has been retrieved.	

Table 4. 25 "Handle Class" Use Case Scenario

Name of *Use Case*	*Handle Class*	
Actor	Teacher	
Description	This Use Case illustrates the activity of handle class into the system.	
Precondition Field	The actors will perform a management class for the classes that are undertaken with the teacher.	
The typical field of an event	**Activity of actor**	**System Response**
	1. Entering Username and Password. 3. Select Handle Class Menu. 5. Input Handle Class detail into the form and submit.	2. Direct to the Dashboard 4. Select Handle Class Form. 6. System Check Required Inputs. 7. The System will retrieve the Handle Class Data by the Actors.
Alternative fields	Alt- step 3. if the actor leaves any of the required input fields the system will alert an error message and if the actor clicks the cancel button it will return to the Handle Class Form.	
Conclusion	Actor successfully handle class data	
Postcondition	Handle Class has been assigned.	

Table 4. 26 "Manage Attendance" Use Case Scenario

Name of *Use Case*	*Manage Attendance*	
Actor	Administrator, Teacher	
Description	This Use Case illustrates the activity of handle class into the system.	
Precondition Field	The actors will perform manage attendance for the classes that are undertaken with the teacher and Admin.	
The typical field of an event	**Activity of actor**	**System Response**
	1. Entering Username and Password. 3. Select Attendance Menu. 5. Mark Attendance and submit.	2. Direct to the Dashboard 4. Select Attendance Form. 6. System Check Required Inputs. 7. The System will save the Attendance Data by the Actors.
Alternative fields	Alt- step 3. If the actor leaves any of the required input fields the system will alert an error message and if the actor clicks the cancel button it will return to the Attendance Form.	
Conclusion	Actor successfully handle attendance	
Postcondition	Attendance has been assigned.	

4.7.2 Activity Diagram

Here is some activity diagrams that are formed from the use case diagram previously discussed.

1. Activity Diagram of the Use Case "Login".

The Login activity diagram illustrates the actors who will login first to access the system.

First to access the system. The actors will need to login, with the system login menu. Then the actor enters their username and password that was created before. When their username and password are incorrect the system will provide them with a warning alert/message shown that the actor’s login credentials are incorrect. However, if the username and password are correct, the system will display the main page of the AIS which is the Dashboard of the specific actor.

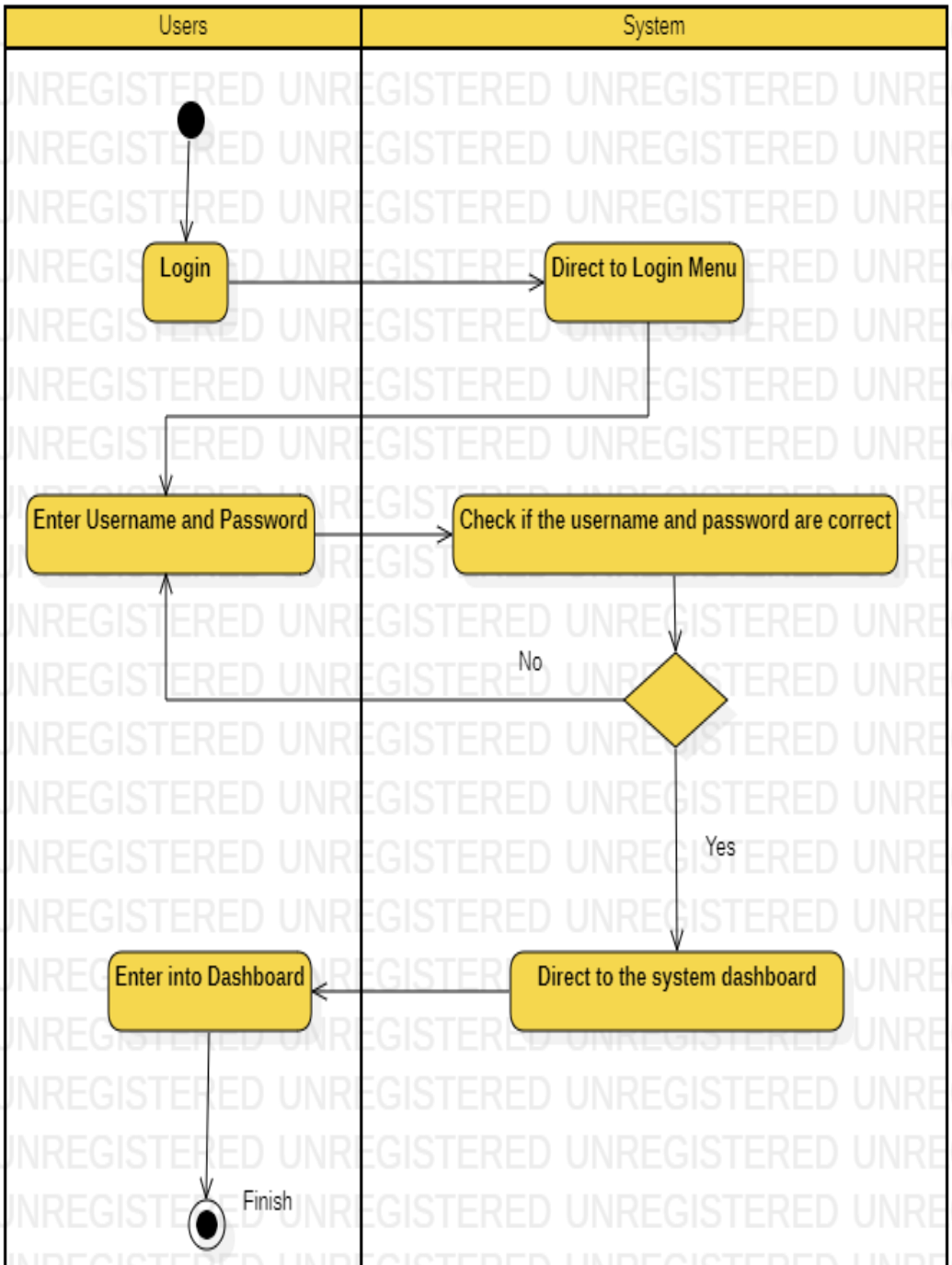


Figure 4. 2 Activity Diagram "Login"

2. Activity Diagram of the Use Case "Manage Student"

The activity diagram of the management students below illustrates an actor who needs to login first and has the privilege of doing activities such as add, edit, view and deletes students. Then the settings made by the actor will be saved and the results will be displayed on the system page.

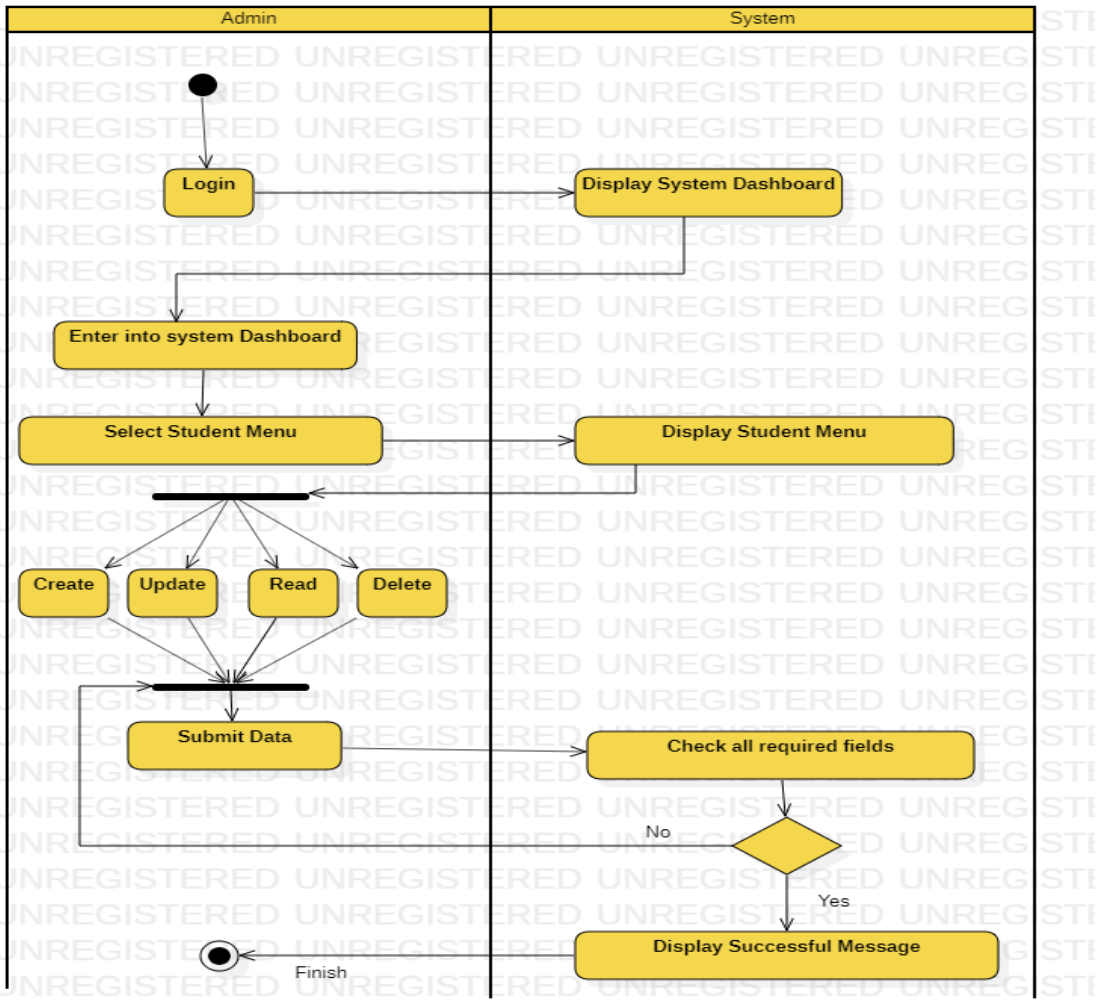


Figure 4. 3 Activity Diagram "Manage Student"

3. Activity Diagram of the Use Case "Manage Teacher"

The activity diagram of the management students below illustrates an actor who needs to login first and has the privilege of doing activities such as add, edit, view and deletes students. Then the settings made by the actor will be saved and the results will be displayed on the system page.

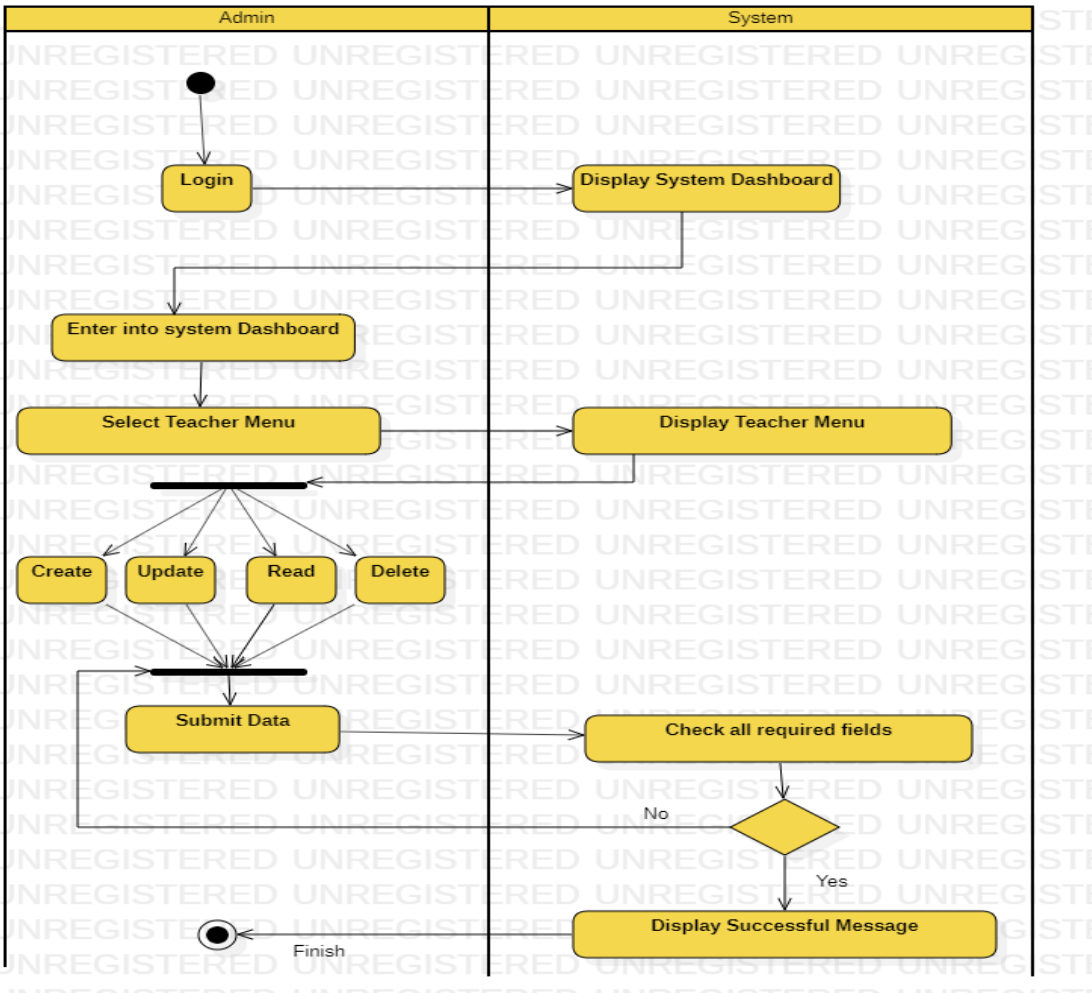


Figure 4. 4 Activity Diagram "Manage Teacher"

4. Activity Diagram of the Use Case "Manage Subject"

The activity diagram of the management students below illustrates an actor who needs to login first and has the privilege of doing activities such as add, edit, view and deletes students. Then the settings made by the actor will be saved and the results will be displayed on the system page.

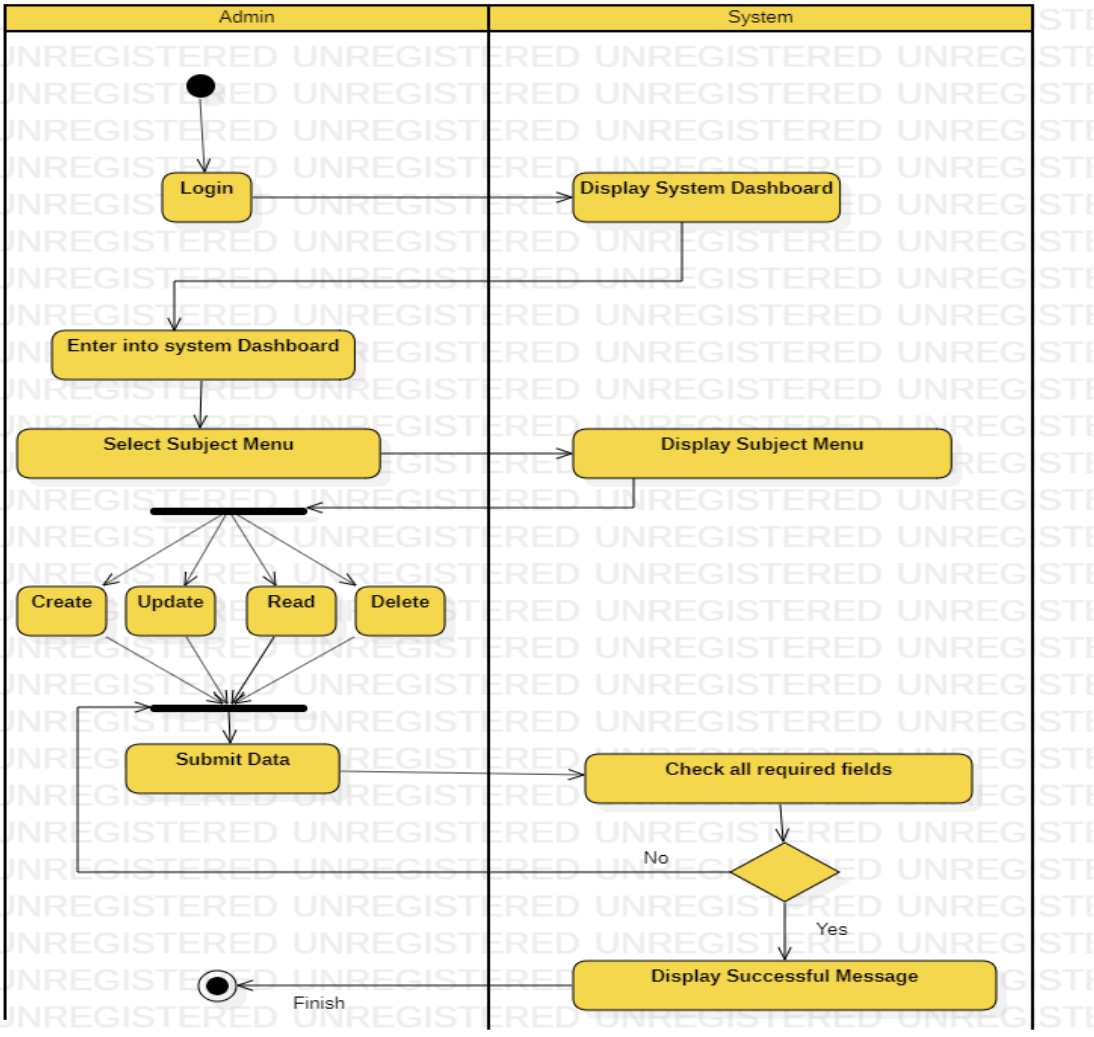


Figure 4. 5 Activity Diagram "Manage Subject"

5. Activity Diagram of the Use Case “Manage Class”

The activity diagram of the manage students below illustrates an actor who needs to login first and has the privilege of doing activities such as add, edit, view and deletes students. Then the settings made by the actor will be saved and the results will be displayed on the system page.

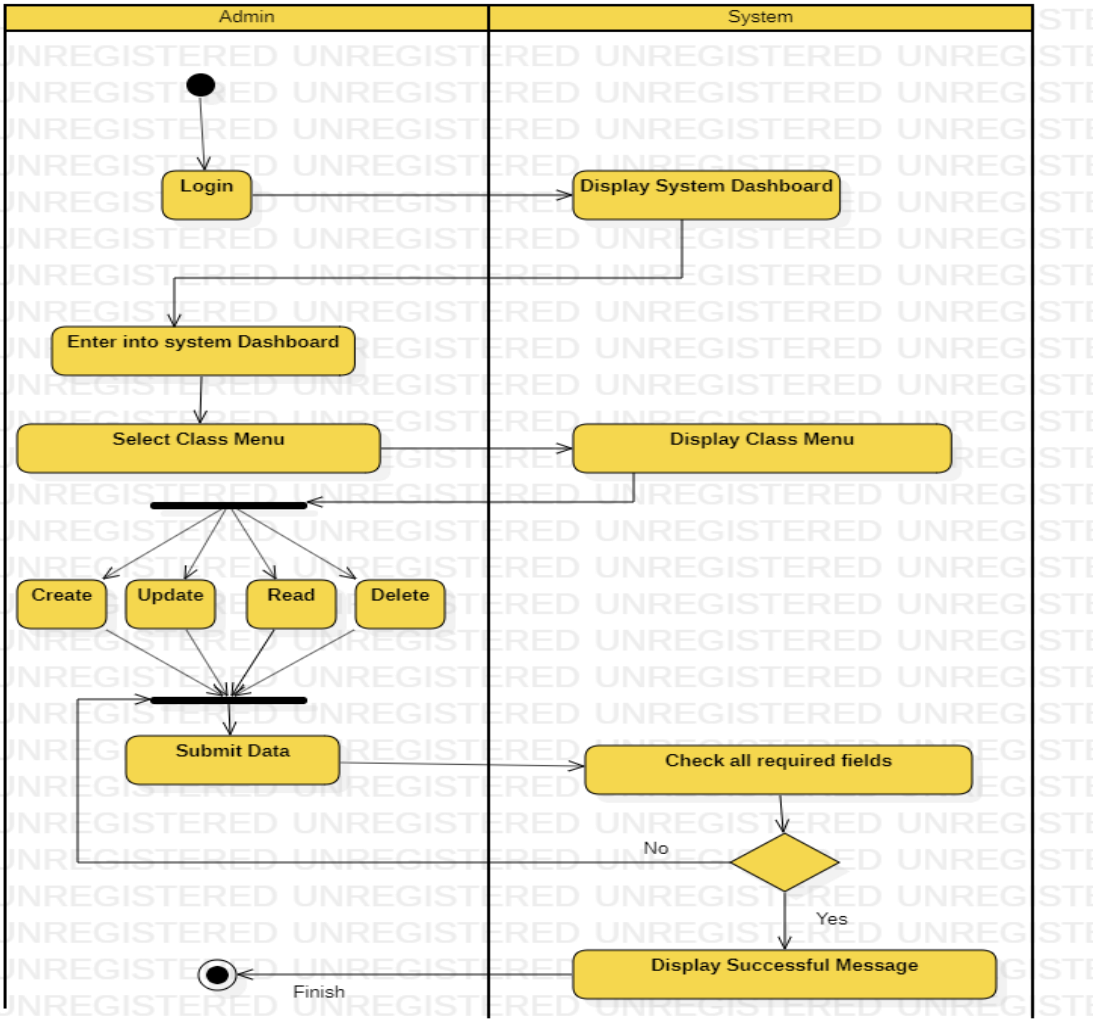


Figure 4. 6 Activity Diagram “Manage Class”

6. Activity Diagram of the Use Case "Manage Semester"

The activity diagram of the manage students below illustrates an actor who needs to login first and has the privilege of doing activities such as add, edit, view and delete students. Then the settings made by the actor will be saved and the results will be displayed on the system page.

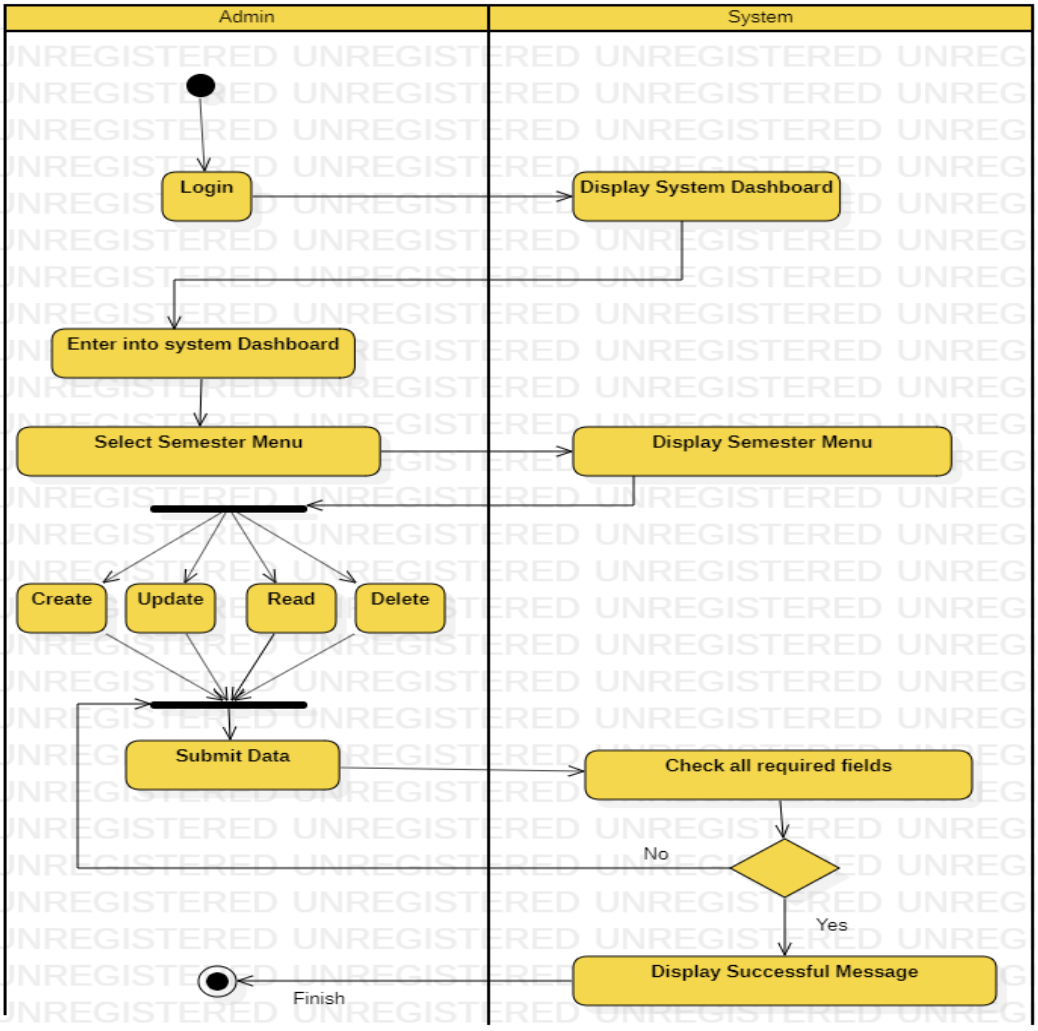


Figure 4. 7 Activity Diagram "Manage Semester"

7. Activity Diagram of the Use Case “Manage Batch”

The activity diagram of the manage students below illustrates an actor who needs to login first and has the privilege of doing activities such as add, edit, view and delete students. Then the settings made by the actor will be saved and the results will be displayed on the system page.

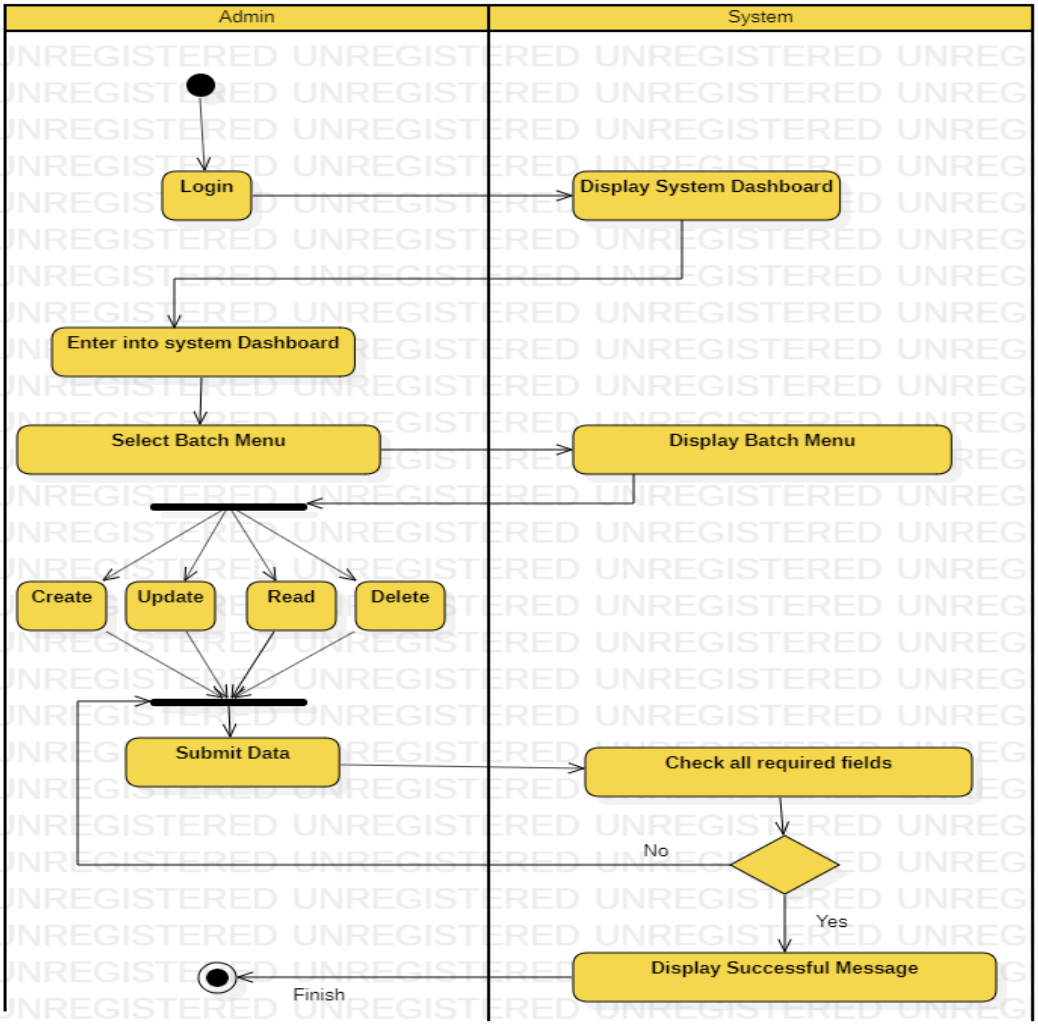


Figure 4. 8 Activity Diagram “Manage Batch”

8. Activity Diagram of the Use Case “Manage Faculty”

The activity diagram of the manage students below illustrates an actor who needs to login first and has the privilege of doing activities such as add, edit, view and delete students. Then the settings made by the actor will be saved and the results will be displayed on the system page.

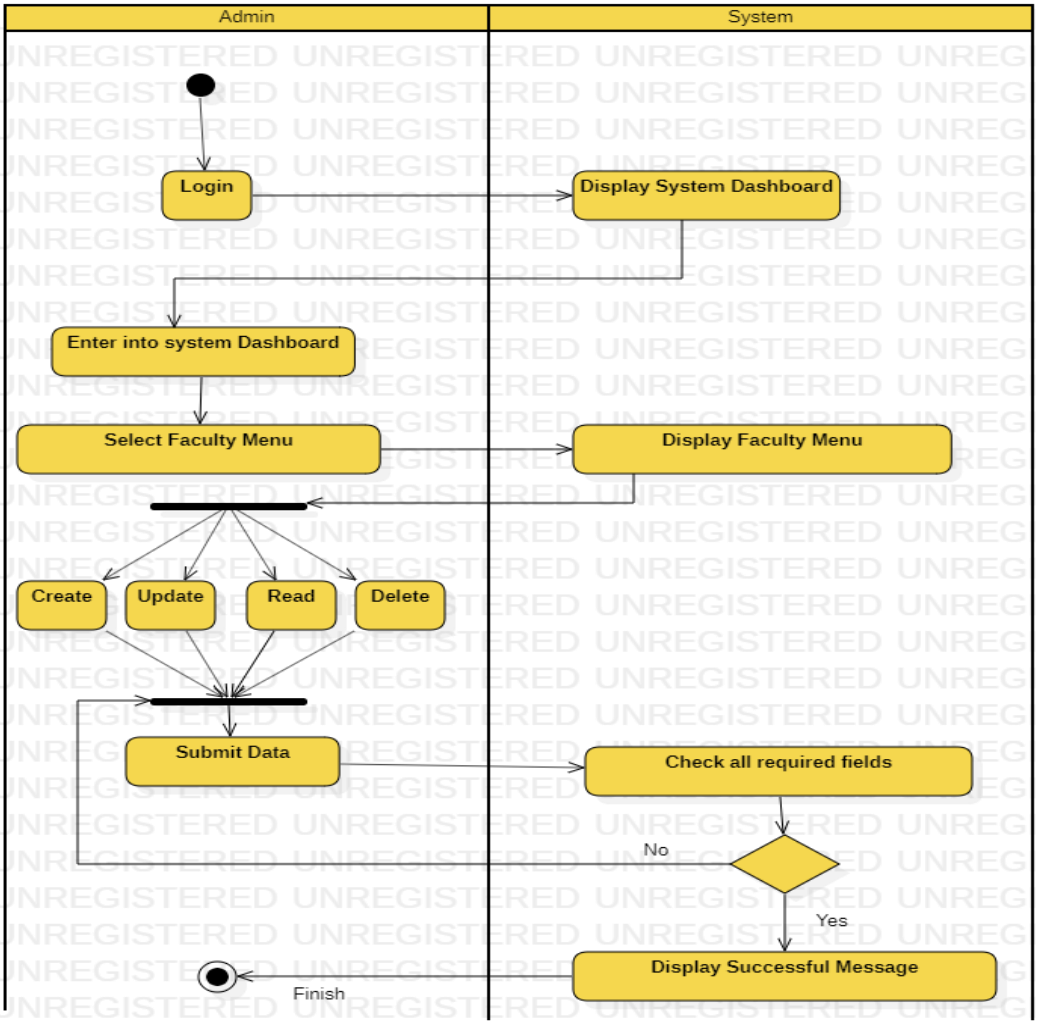


Figure 4. 9 Activity Diagram “Manage Faculty”

9. Activity Diagram of the Use Case "Manage Department"

The activity diagram of the manage students below illustrates an actor who needs to login first and has the privilege of doing activities such as add, edit, view and delete students. Then the settings made by the actor will be saved and the results will be displayed on the system page.

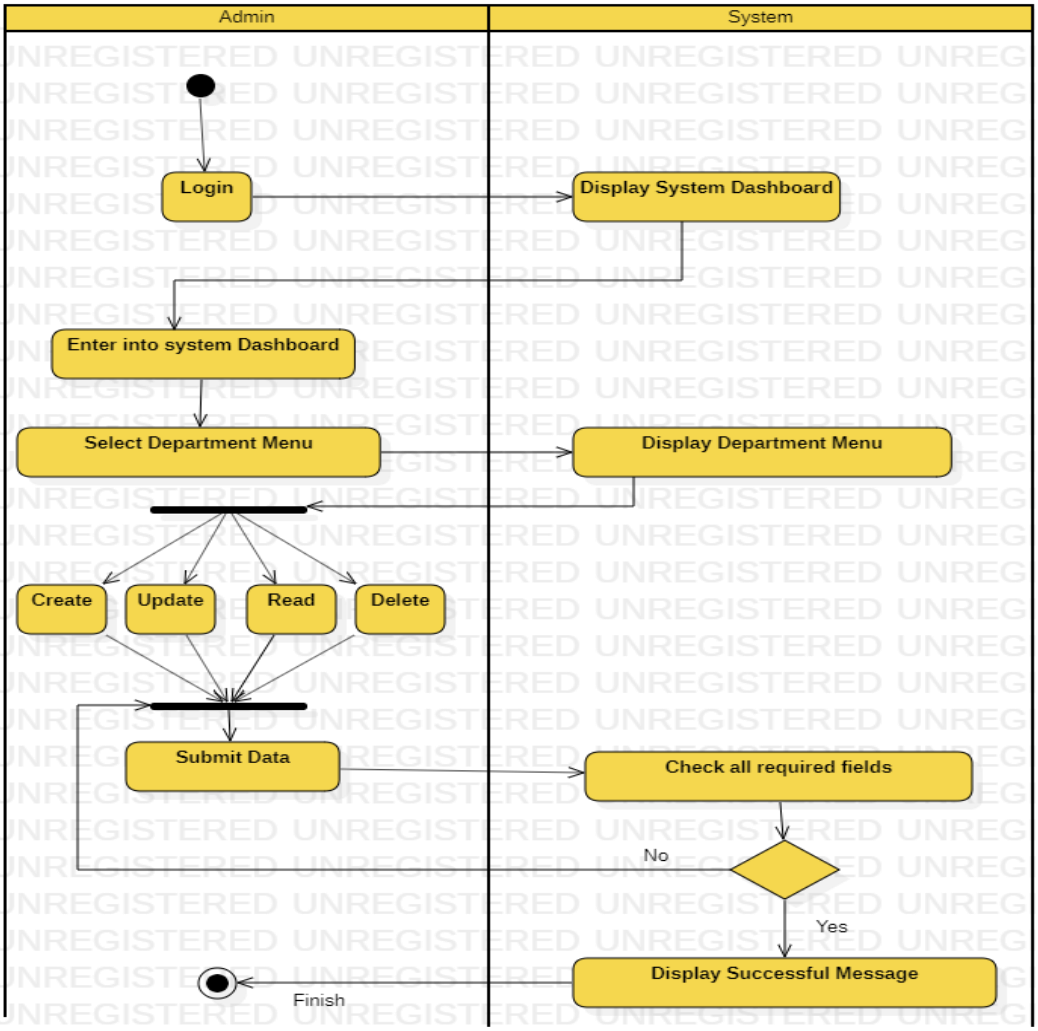


Figure 4. 10 Activity Diagram "Manage Department"

10. Activity Diagram of the Use Case "Manage Timetable"

The activity diagram of the manage students below illustrates an actor who needs to login first and has the privilege of doing activities such as add, edit, view and delete students. Then the settings made by the actor will be saved and the results will be displayed on the system page.

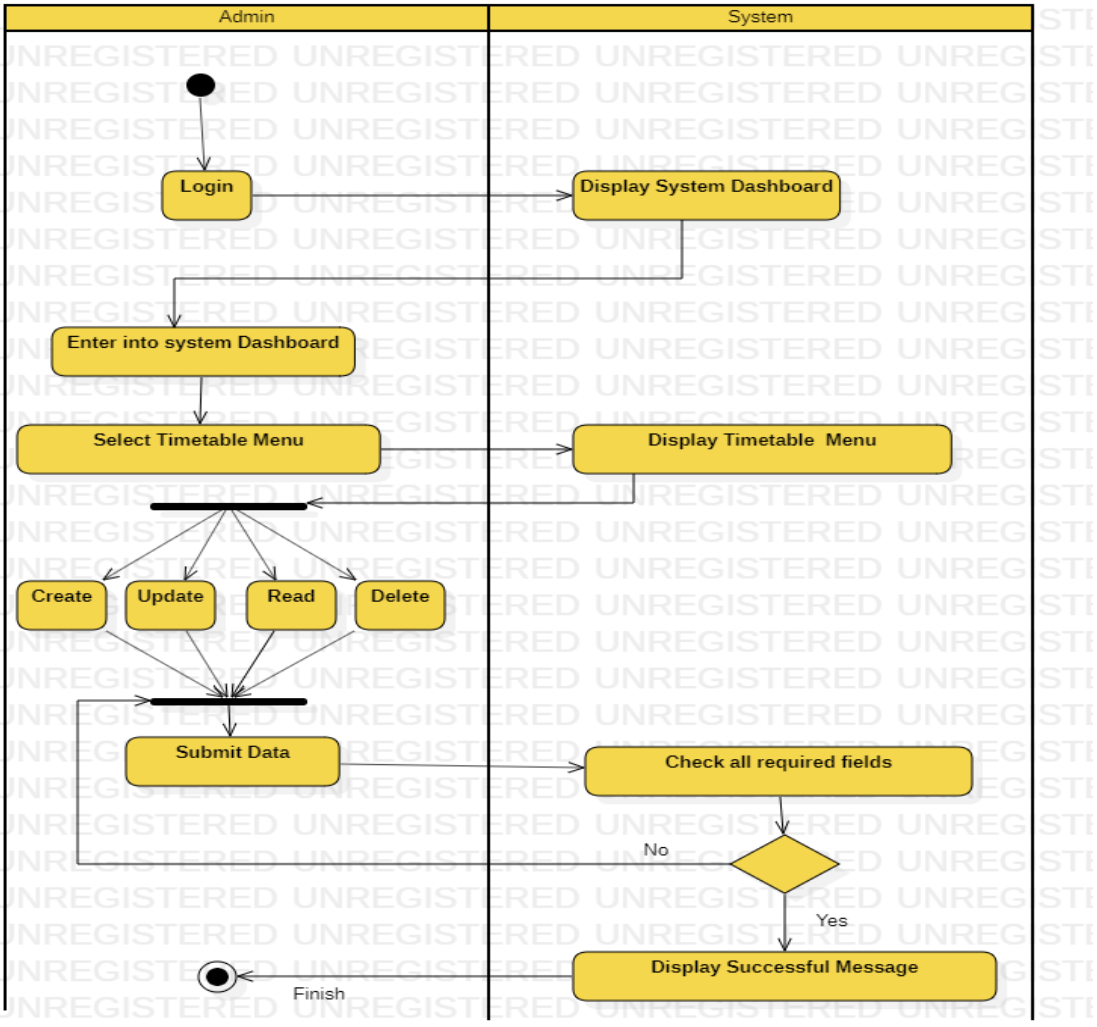


Figure 4. 11 Activity Diagram "Manage Timetable"

11. Activity Diagram of the Use Case “Manage Exams”

The activity diagram of the manage students below illustrates an actor who needs to login first and has the privilege of doing activities such as add, edit, view and delete exam. Then the settings made by the actor will be saved and the results will be displayed on the system page.

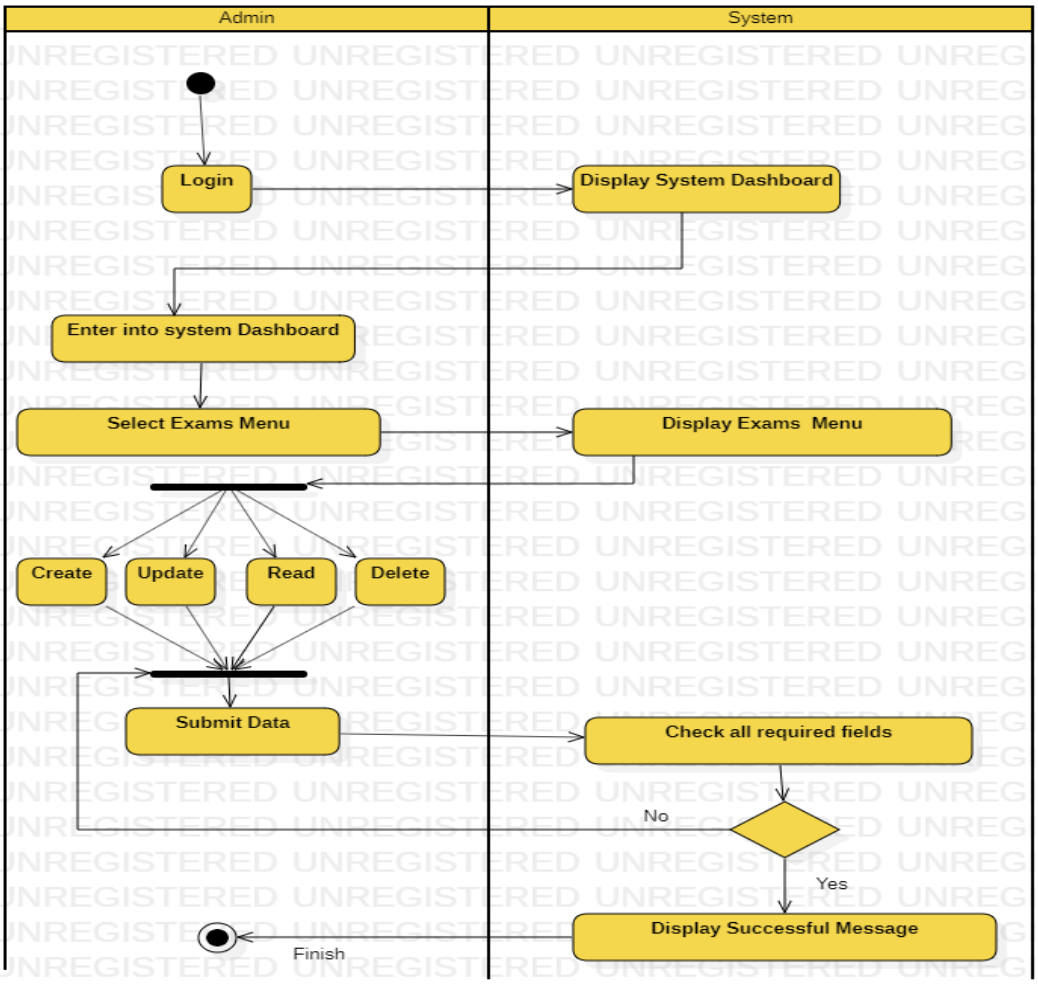


Figure 4. 12 Activity Diagram “Manage Exams”

12. Activity Diagram of the Use Case “Manage Shifts”

The activity diagram of the manage students below illustrates an actor who needs to login first and has the privilege of doing activities such as add, edit, view and delete students. Then the settings made by the actor will be saved and the results will be displayed on the system page.

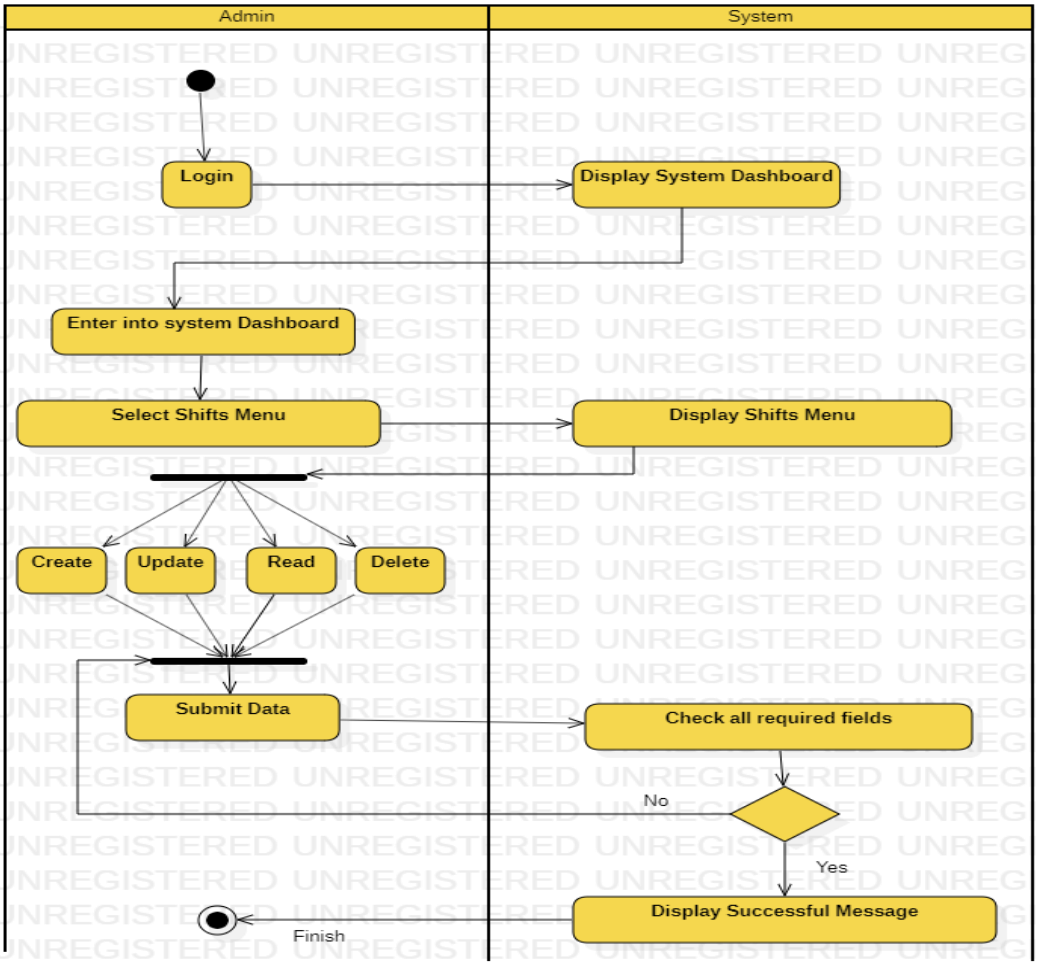


Figure 4. 13 Activity Diagram “Manage Shifts”

13. Activity Diagram of the Use Case “Manage Degree”

The activity diagram of the manage students below illustrates an actor who needs to login first and has the privilege of doing activities such as add, edit, view and delete students. Then the settings made by the actor will be saved and the results will be displayed on the system page.

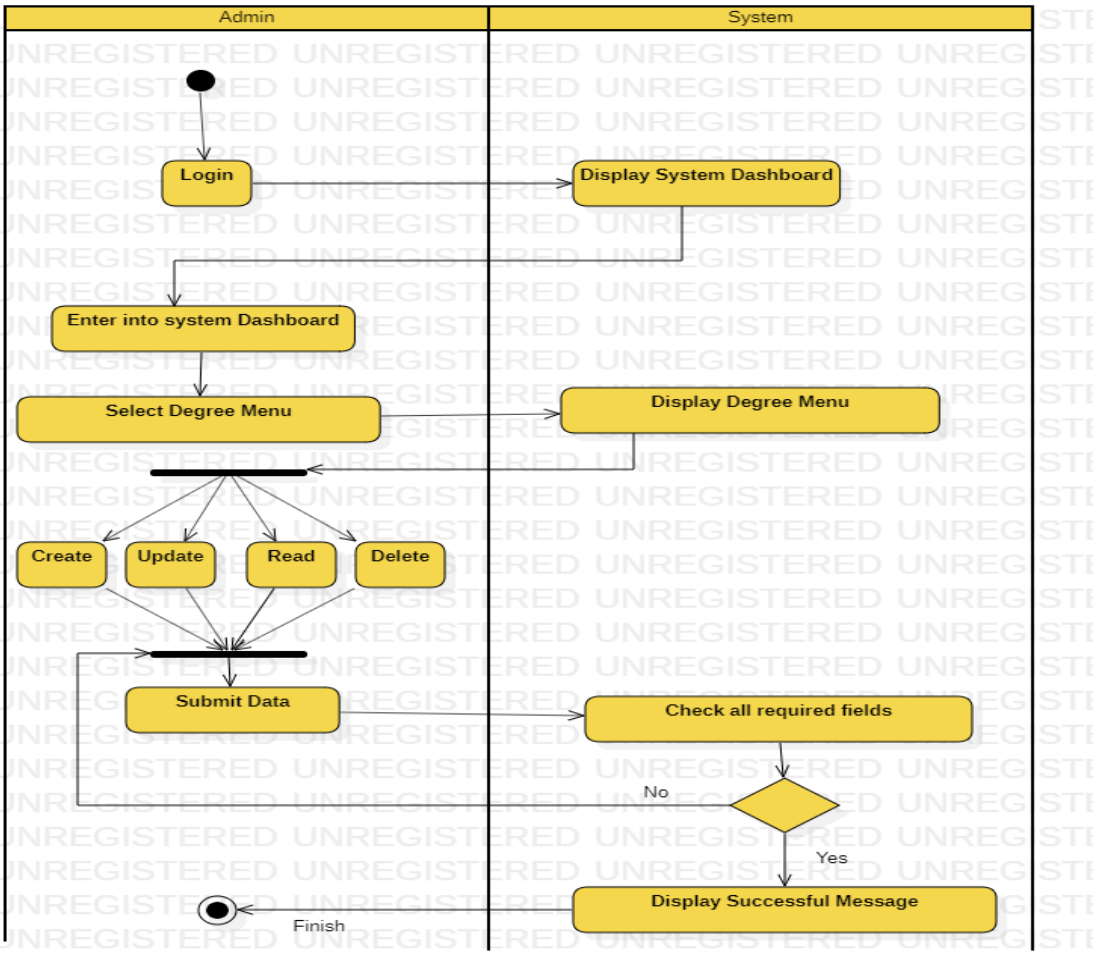


Figure 4. 14 Activity Diagram “Manage Degree”

14. Activity Diagram of the Use Case “Manage Fees”

The activity diagram of the manage students below illustrates an actor who needs to login first and has the privilege of doing activities such as add, edit, view and delete students. Then the settings made by the actor will be saved and the results will be displayed on the system page.

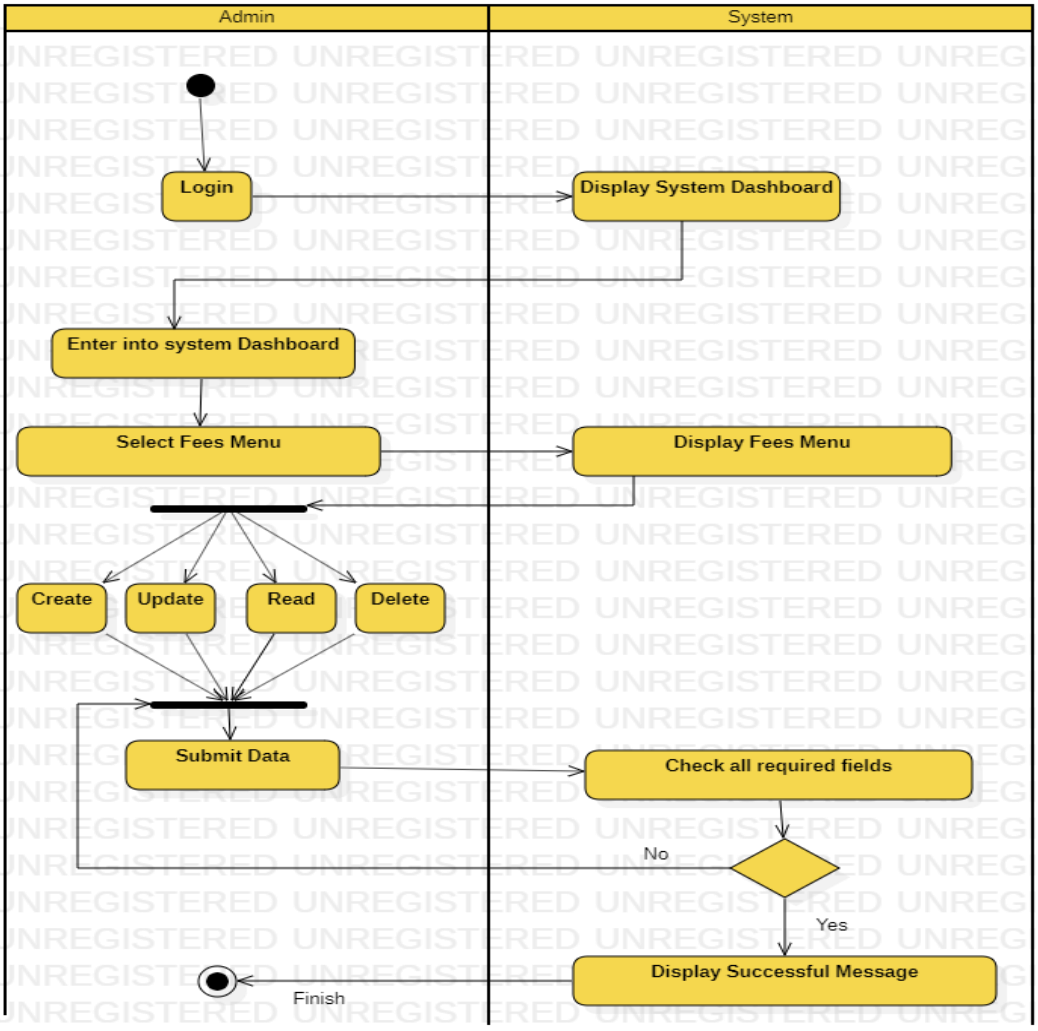


Figure 4. 15 Activity Diagram “Manage Fees”

15. Activity Diagram of the Use Case "Manage Class Schedule"

The activity diagram of the manage students below illustrates an actor who needs to login first and has the privilege of doing activities such as add, edit, view and delete students. Then the settings made by the actor will be saved and the results will be displayed on the system page.

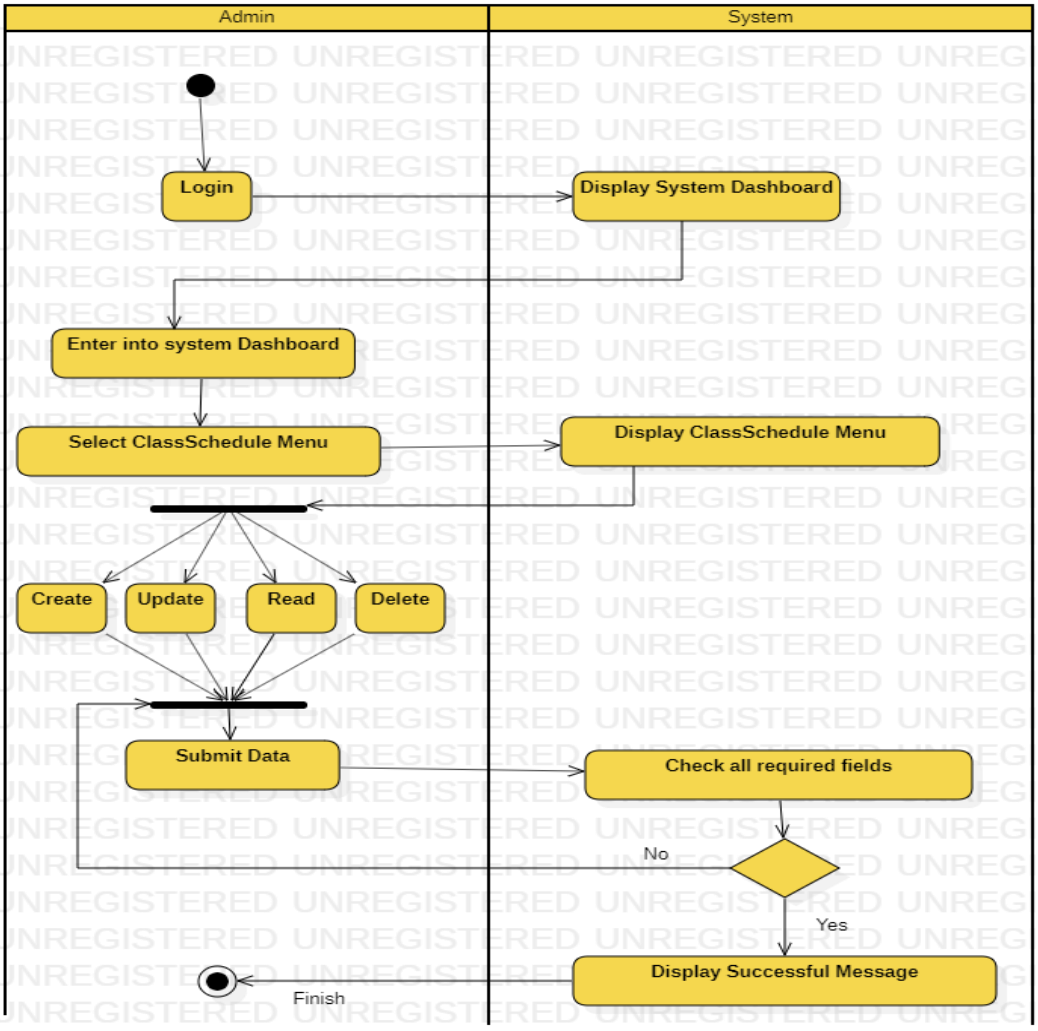


Figure 4. 16 Activity Diagram "Manage Class Schedule"

4.8 Database Structure

4.8.1 Class Diagram Design

At this stage, the database needs to be designed using class diagram tools. A class diagram is a graphic about the static object structure of a system. The first step that needs to be done to create a class diagram is to identify nouns that might describe potential objects. The identification phase of potential objects can be seen in the table below:

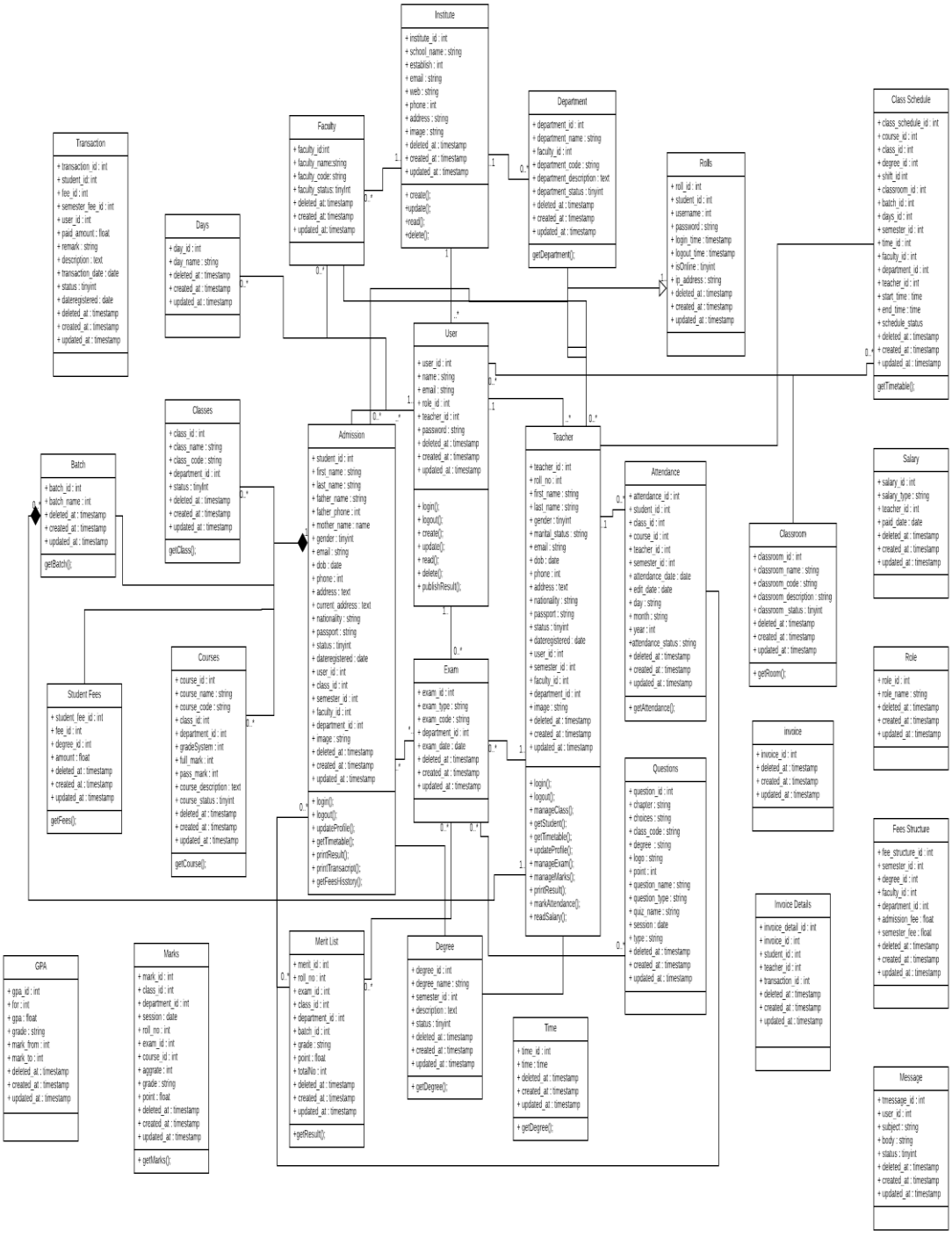


Figure 4. 17 UML "Class Diagram"

4.8.2 Entity Relationship Diagram (ERD)

An entity-relationship diagram (**ERD**) shows the relationships of entity sets stored in a database. It illustrates the logical structure of the database. From the results of the Class Diagram above, show the objects which will be used for system requirements in the form of tables. The tables are linked through foreign keys to perform relationship.

.

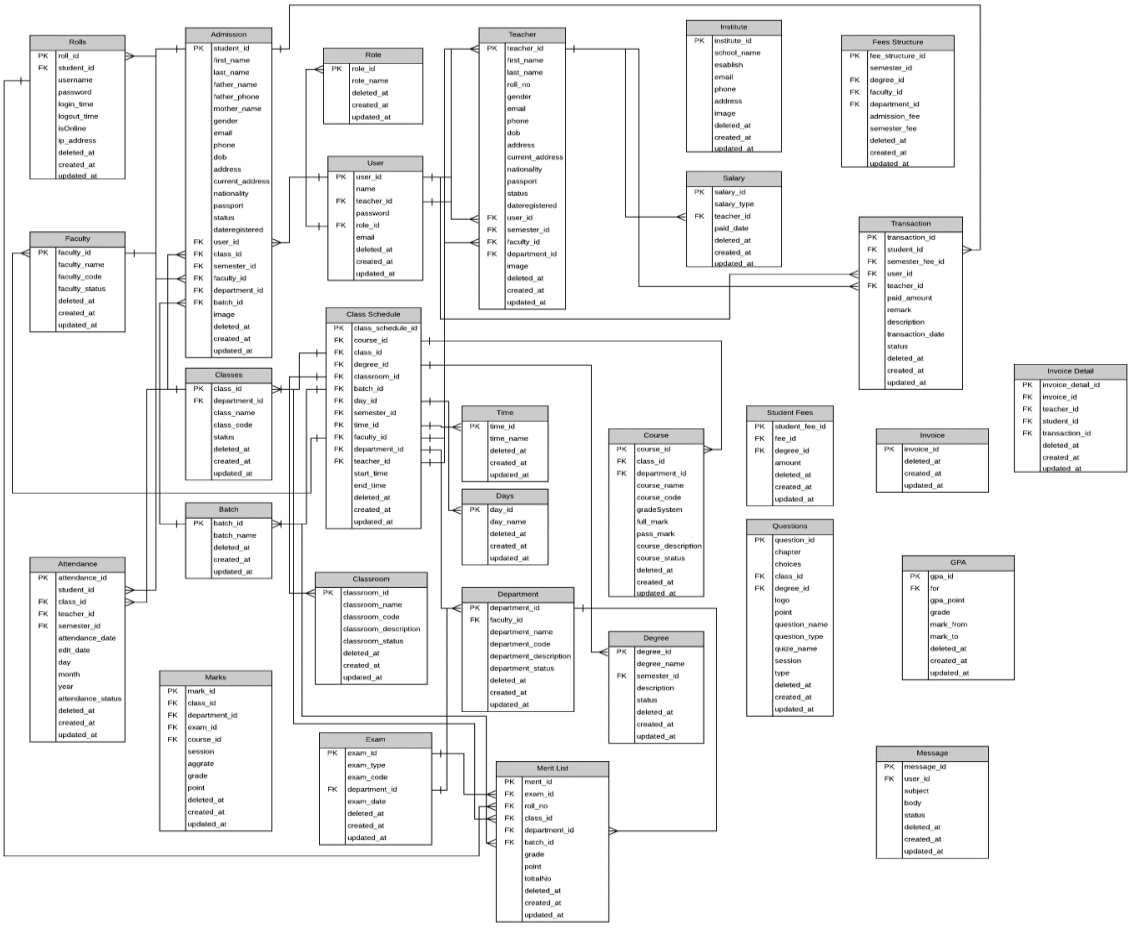


Figure 4. 18 ERD "Entity Relational Diagram"

4.8.3 Sequence Diagram Design

In this stage, the researcher discusses how to make sequential diagrams seen from the use case diagram and class AIS diagram. The sequence diagram for this academic information system (AIS) is as follows:

1. Login sequence diagram

The sequence diagram that has been used for this system illustrates the process of how users will have access to login into the system.

The first thing the user needs to enter the username and password. Then the system will validate the username and password. If the combination is correct, the system will display the main page. Else system will return the user to the login form to enter a correct validate credential.

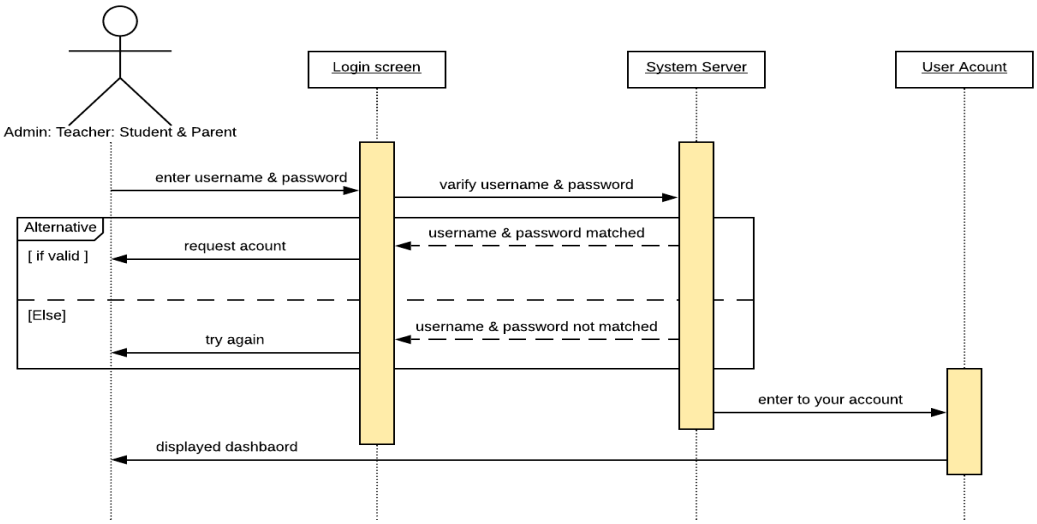


Figure 4. 19 Login Sequence Diagram

2. Manage Student Sequence Diagram

The sequence diagram below shows the actor who will manage student the actor can add student, edit student, view student, and delete student.

Therefore the actor can select add new student form. Then the system will display a form that the actor will enter the student details and if the student details are incomplete the system will through an error to tell the actor there are some required information that is not complete, else the system will save the new student information inside the database and return to student list table. Then the actor can select edit, delete or view to change the student details.

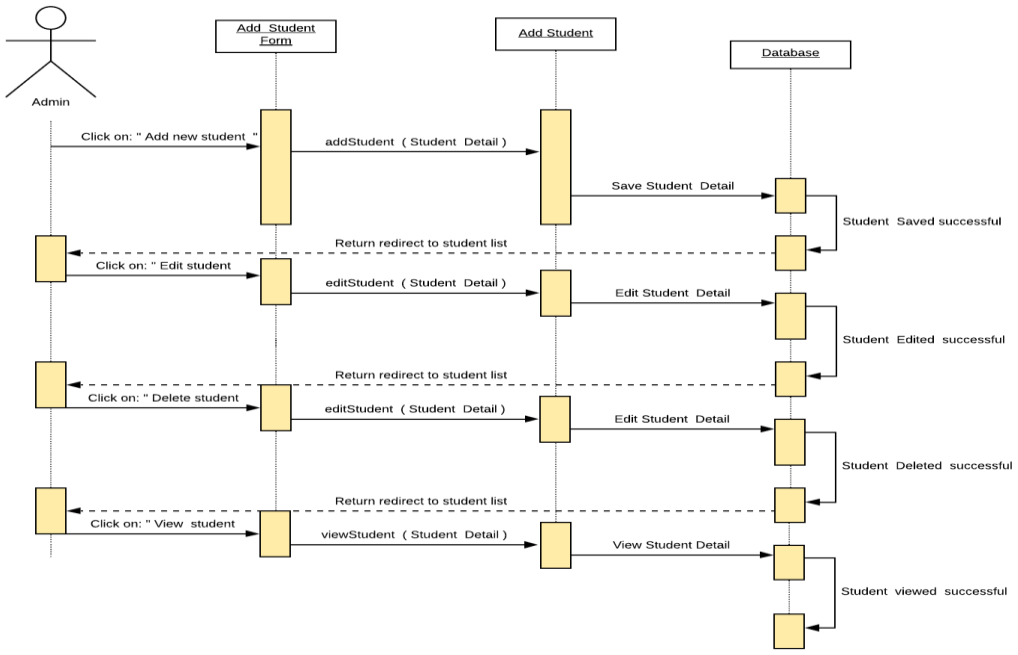


Figure 4. 20 Manage Student Sequence Diagram

3. Manage Teacher Sequence Diagram

The sequence diagram below shows the actor who will manage the teacher. The actor can add teacher, edit teacher, view teacher, and delete teacher. Therefore the actor can select add new teacher form. Then the system will display a form that the actor will enter the teacher details and if the teacher details are incomplete the system will through an error to tell the actor there are some required information that is not complete, else the system will save the new teacher information inside the database and return to teacher list table. Then the actor can select edit, delete or view to change the teacher details.

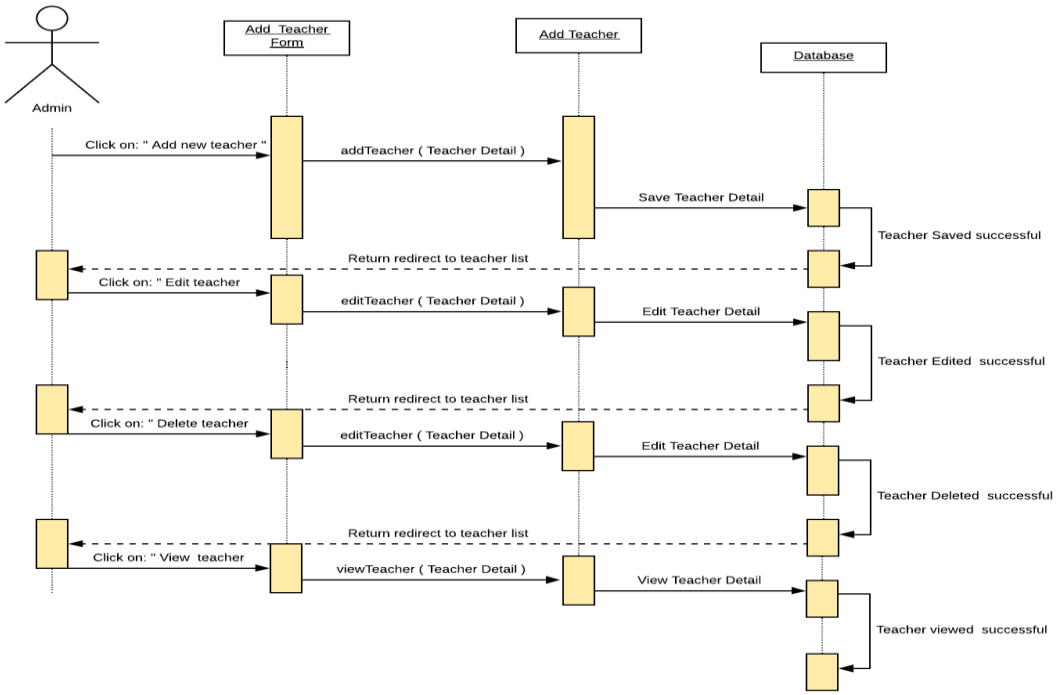


Figure 4. 21 Manage Teacher Sequence Diagram

4. Manage Faculty Sequence Diagram

The sequence diagram below shows the actor who will manage the faculty. The actor can add faculty, edit faculty, view faculty, and delete faculty. Therefore the actor can select add new faculty form. Then the system will display a form that the actor will enter the faculty details and if the faculty details are incomplete the system will through an error to tell the actor there are some required information that is not complete, else the system will save the new faculty information inside the database and return to faculty list table. Then the actor can select edit, delete or view to change the faculty details.

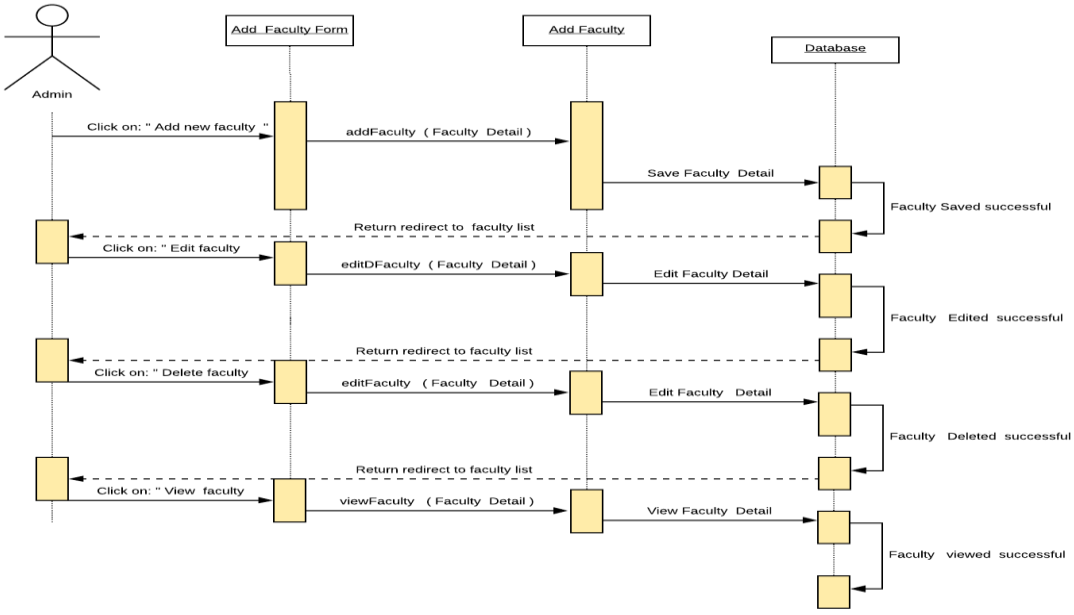


Figure 4. 22 Manage Faculty Sequence Diagram

5. Manage Department Sequence Diagram

The sequence diagram below shows the actor who will manage the department. The actor can add a department, edit department, view department, and delete department. Therefore the actor can select add new department form. Then the system will display a form that the actor will enter the department details and if the department details are incomplete the system will through an error to tell the actor there are some required information that is not complete, else the system will save the new department information inside the database and return to department list table. Then the actor can select edit, delete or view to change the department details.

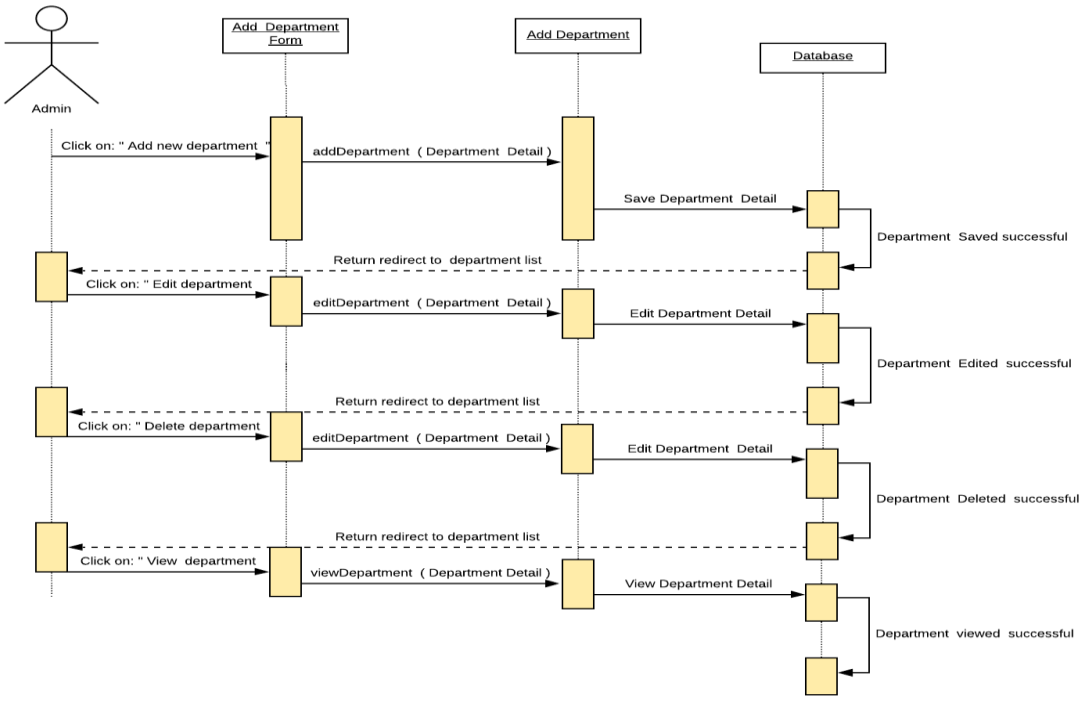


Figure 4. 23 Manage Department Sequence Diagram

6. Manage Class Sequence Diagram

The sequence diagram below shows the actor who will manage the class. The actor can add class, edit class, view class, and delete the class. Therefore the actor can select add new class form. Then the system will display a form that the actor will enter the class details and if the class details are incomplete the system will through an error to tell the actor there are some required information that is not complete, else the system will save the new class information inside the database and return to class list table. Then the actor can select edit, delete or view to change the class details.

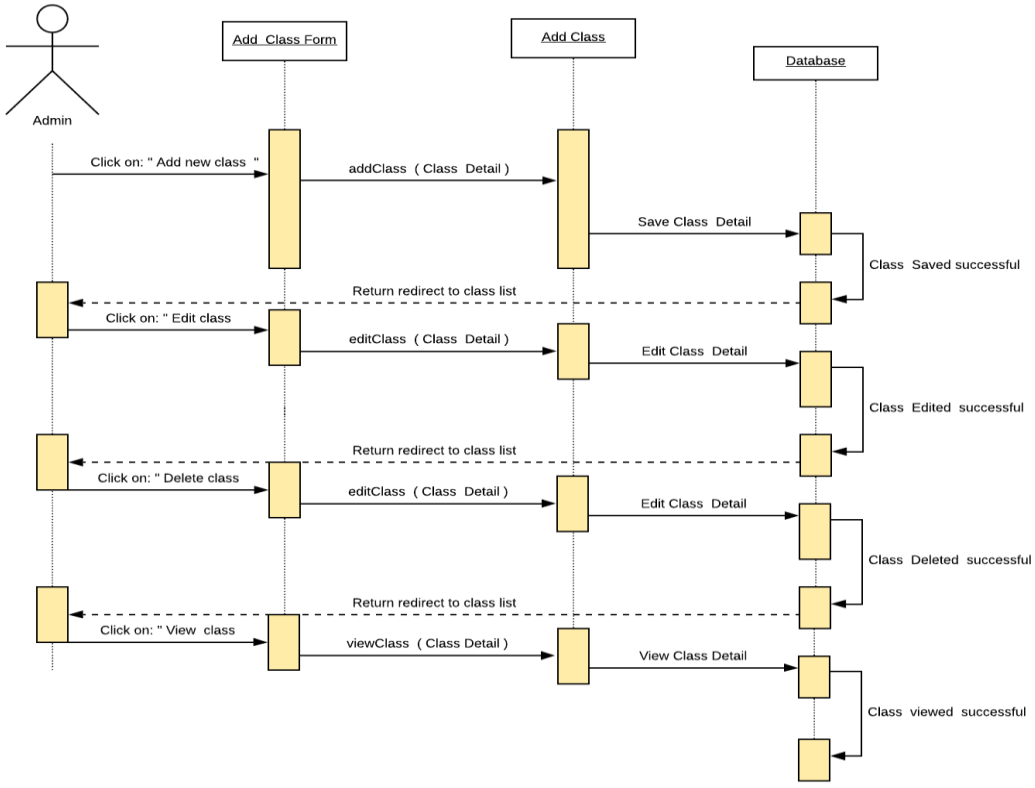


Figure: Manage Class Sequence Diagram

7. Manage Semester Sequence Diagram

The sequence diagram below shows the actor who will manage the semester. The actor can add semester, edit semester, view semester, and delete semester. Therefore the actor can select add new semester form. Then the system will display a form that the actor will enter the semester details and if the semester details are incomplete the system will through an error to tell the actor there are some required information that is not complete, else the system will save the new semester information inside the database and return to class list table. Then the actor can select edit, delete or view to change the semester details.

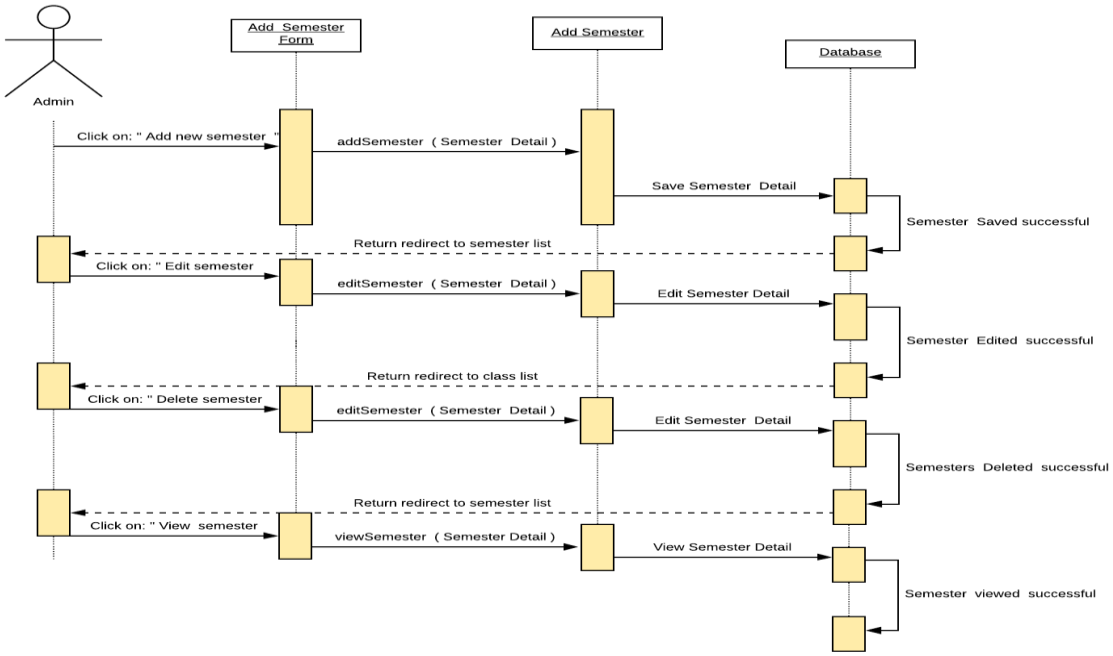


Figure 4. 24 Manage Semester Sequence Diagram

8. Manage Course Sequence Diagram

The sequence diagram below shows the actor who will manage the course. The actor can add a course, edit course, view course, and delete a course. Therefore the actor can select add new course form. Then the system will display a form that the actor will enter the course details and if the course details are incomplete the system will through an error to tell the actor there are some required information that is not complete, else the system will save the new course information inside the database and return to course list table. Then the actor can select edit, delete or view to change the course details.

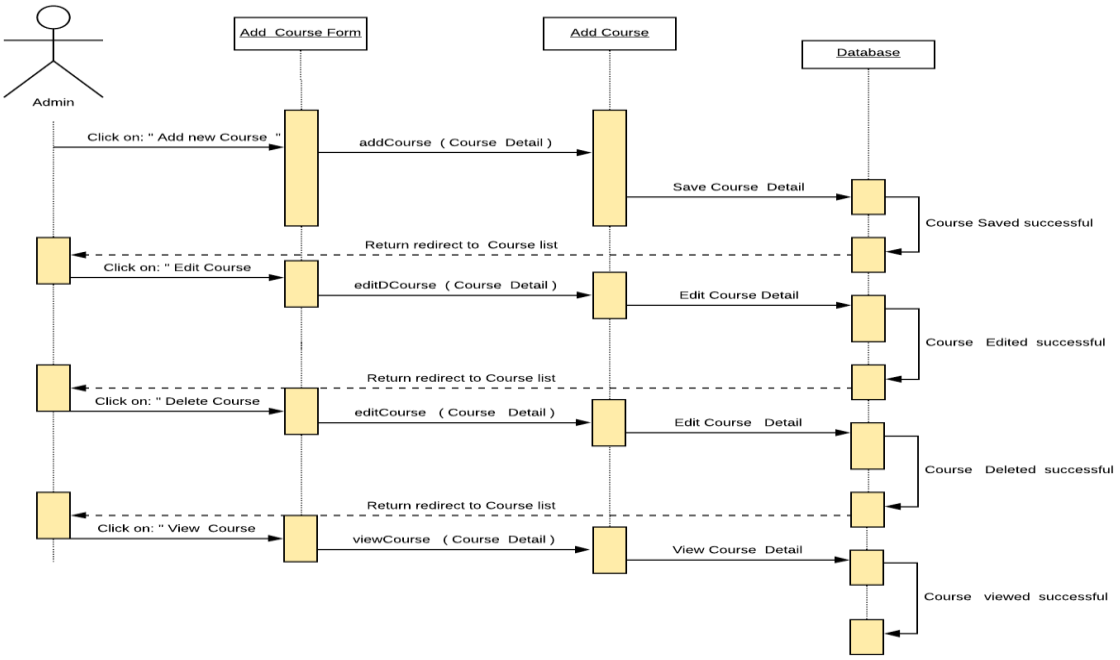


Figure 4. 25 Manage Course Sequence Diagram

9. Manage Degree Sequence Diagram

The sequence diagram below shows the actor who will manage a degree. The actor can add a degree, edit degree, view degree, and delete a degree. Therefore the actor can select add new degree form. Then the system will display a form that the actor will enter the degree details and if the course details are incomplete the system will through an error to tell the actor there are some required information that is not complete, else the system will save the new degree information inside the database and return to degree list table. Then the actor can select edit, delete or view to change the degree details.

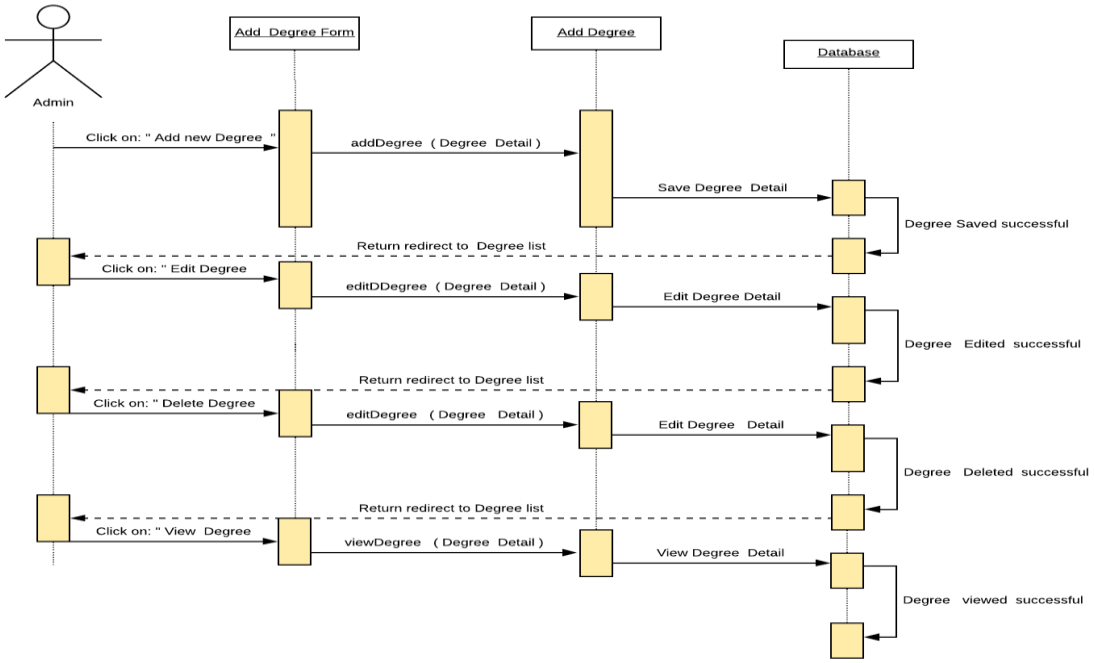


Figure 4. 26 Manage Degree Sequence Diagram

10. Manage Batch Sequence Diagram

The sequence diagram below shows the actor who will manage a batch. The actor can add batch, edit batch, view batch, and delete a batch. Therefore the actor can select add new batch form. Then the system will display a form that the actor will enter the batch details and if the batch details are incomplete the system will through an error to tell the actor there are some required information that is not complete, else the system will save the new batch information inside the database and return to batch list table. Then the actor can select edit, delete or view to change the batch details.

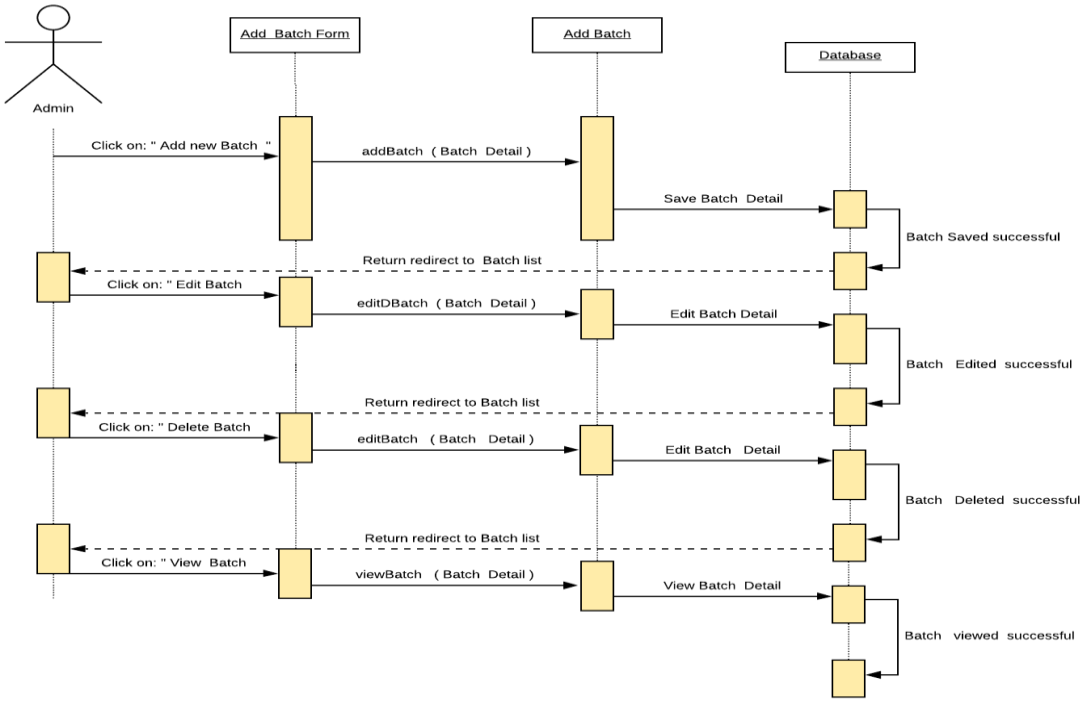


Figure 4. 27 Manage Batch Sequence Diagram

11. Manage Time Sequence Diagram

The sequence diagram below shows the actor who will manage time. The actor can add time, edit time, view time, and delete time. Therefore the actor can select add new time form. Then the system will display a form that the actor will enter the time details and if the time details are incomplete the system will through an error to tell the actor there are some required information that is not complete, else the system will save the new time information inside the database and return to time list table. Then the actor can select edit, delete or view to change the time details

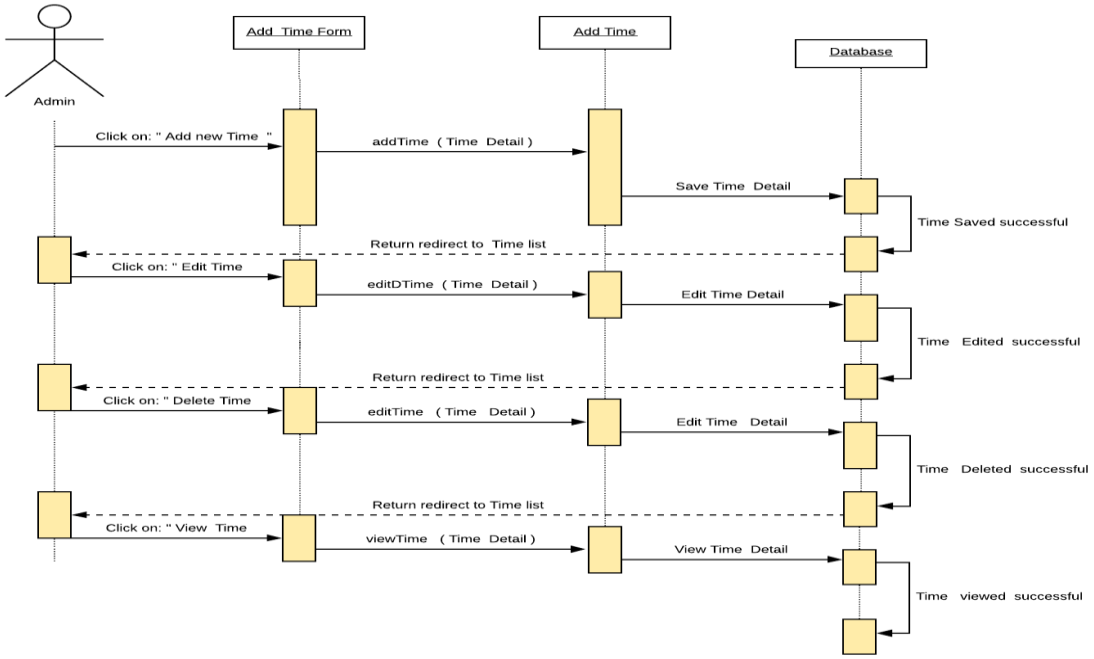


Figure 4. 28 Manage Time Sequence Diagram

12. Manage Day Sequence Diagram

The sequence diagram below shows the actor who will manage the day. The actor can add a day, edit day, view day, and delete day. Therefore the actor can select add new day form. Then the system will display a form that the actor will enter the day details and if the day details are incomplete the system will through an error to tell the actor there are some required information that is not complete, else the system will save the new day information inside the database and return to the day list table. Then the actor can select edit, delete or view to change the day details

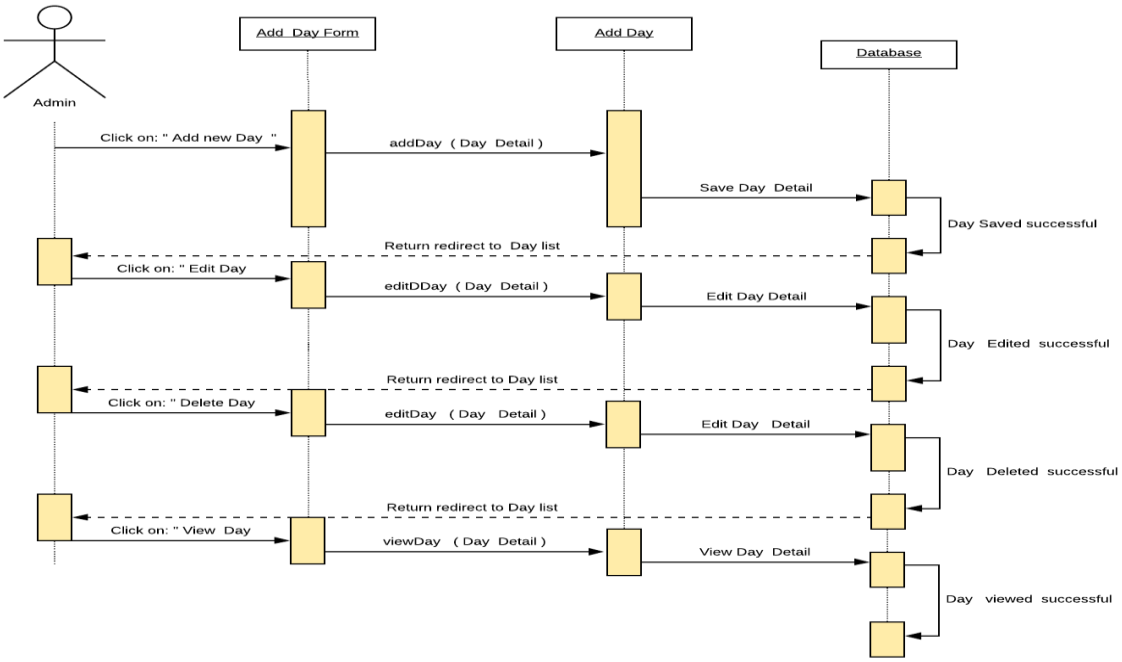


Figure 4. 29 Manage Day Sequence Diagram

13. Manage Shift Sequence Diagram

The sequence diagram below shows the actor who will manage a shift. The actor can add a shift, edit shift, view shift, and delete shift. Therefore the actor can select add new shift form. Then the system will display a form that the actor will enter the shift details and if the shift details are incomplete the system will through an error to tell the actor there are some required information that is not complete, else the system will save the new shift information inside the database and return to shift list table. Then the actor can select edit, delete or view to change the shift details

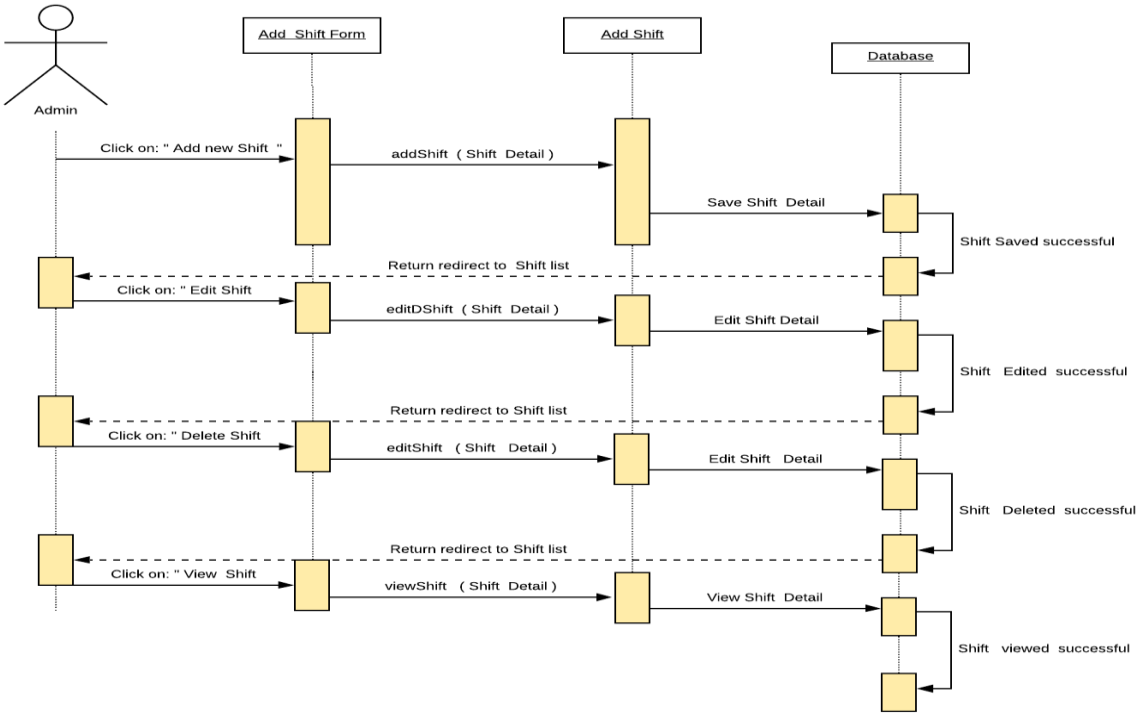


Figure 4. 30 Manage Shift Sequence Diagram

14. Manage Role Sequence Diagram

The sequence diagram below shows the actor who will manage the role. The actor can add a role, edit a role, view role, and delete the role. Therefore the actor can select add new role form. Then the system will display a form that the actor will enter the role details and if the role details are incomplete the system will through an error to tell the actor there are some required information that is not complete, else the system will save the new role information inside the database and return to role list table. Then the actor can select edit, delete or view to change the role details

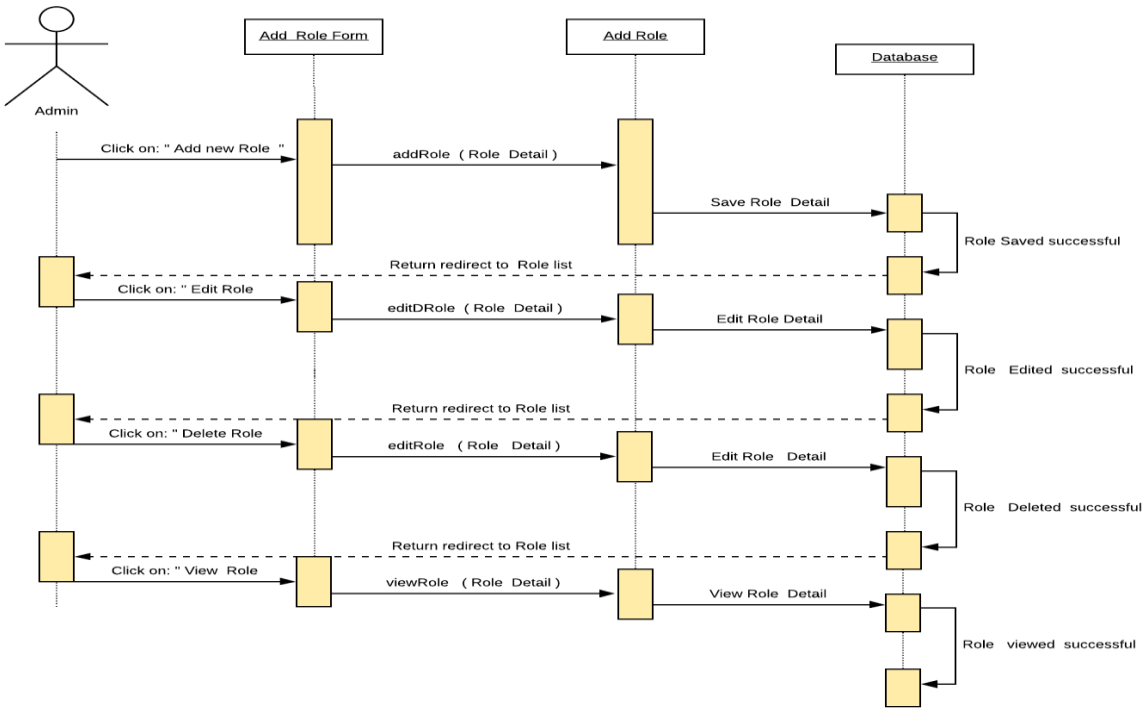


Figure 4. 31 Manage Role Sequence Diagram

15. Manage Fee Structure Sequence Diagram

The sequence diagram below shows the actor who will manage the fee structure. The actor can add fee structure, edit fee structure, view fee structure, and delete fee structure. Therefore the actor can select add new fee structure form. Then the system will display a form that the actor will enter the fee structure details and if the fee structure details are incomplete the system will through an error to tell the actor there are some required information that is not complete, else the system will save the new fee structure information inside the database and return to fee structure list table. Then the actor can select edit, delete or view to change the fee structure details

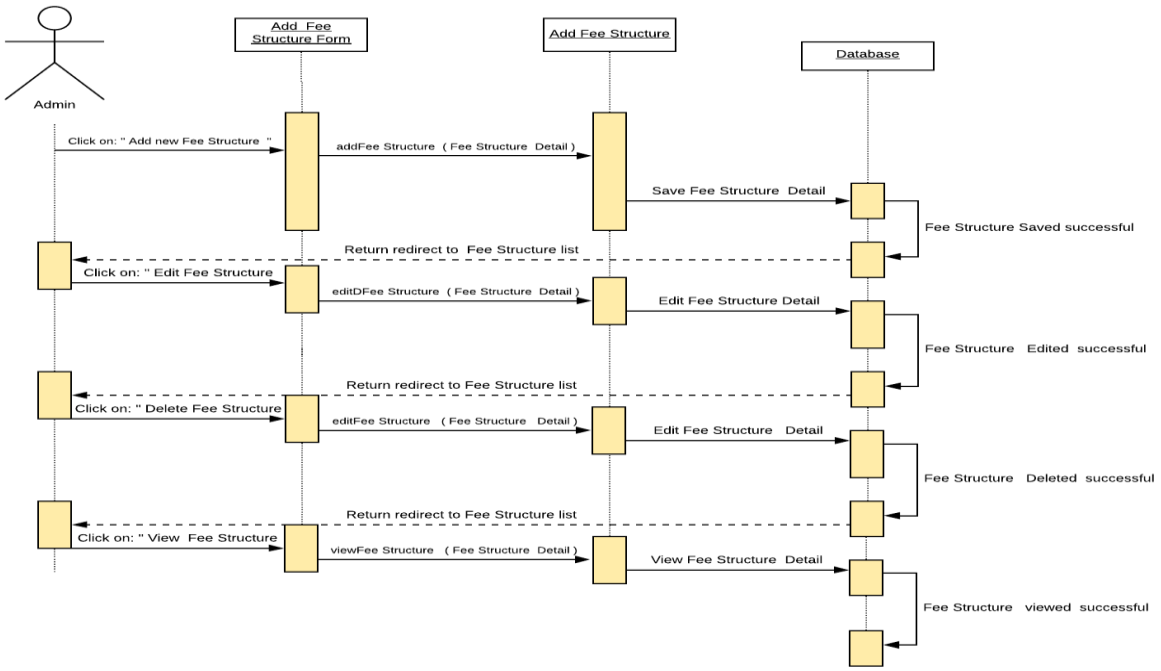


Figure 4. 32 Manage Fee Structure Sequence Diagram

16. Manage Questions Sequence Diagram

The sequence diagram below shows the actor who will manage the question. The actor can create a question, edit a question, view question, and delete a question. Therefore the actor can select add new question form. Then the system will display a form that the actor will enter the question details and if the question details are incomplete the system will through an error to tell the actor there are some required information that is not complete, else the system will save the new question information inside the database and return to question list table. Then the actor can select edit, delete or view to change the question details

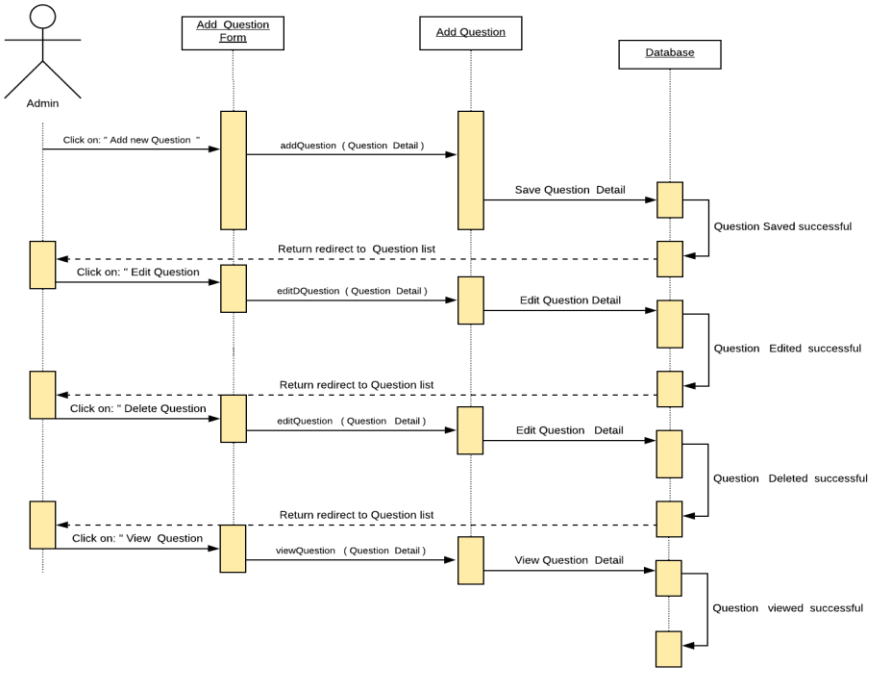


Figure 4. 33 Manage Question Sequence Diagram

17. Manage Exam Sequence Diagram

The sequence diagram below shows the actor who will manage the exam. The actor can create an exam, edit exam, view exam, and delete an exam. Therefore the actor can select add new exam form. Then the system will display a form that the actor will enter the exam details and if the exam details are incomplete the system will through an error to tell the actor there are some required information that is not complete, else the system will save the new exam information inside the database and return to exam list table. Then the actor can select edit, delete or view to change the exam details

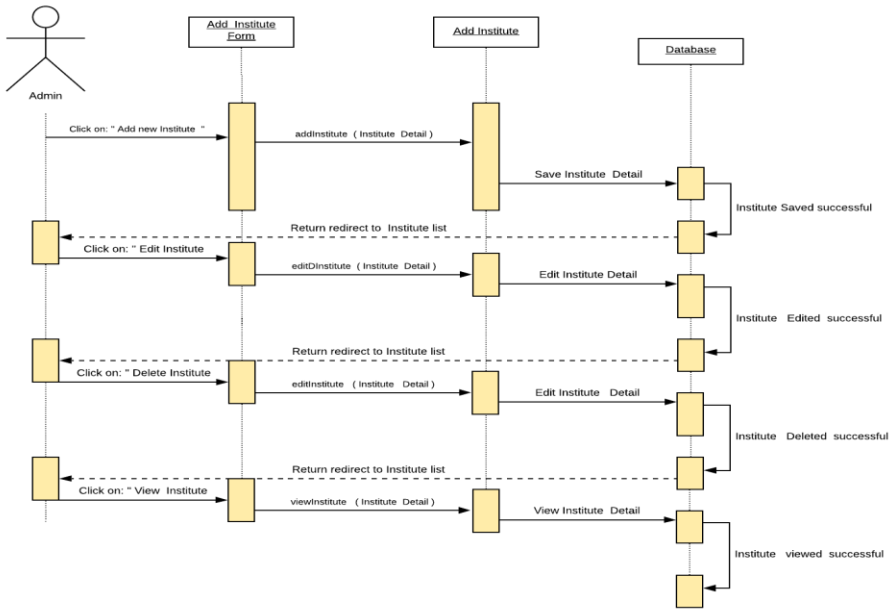


Figure 4. 34 Manage Institute Sequence Diagram

18. Manage Institute Sequence Diagram

The sequence diagram below shows the actor who will manage the institute. The actor can add institute detail, edit institute, view institute, and delete institute. Therefore the actor can select add new institute detail form. Then the system will display a form that the actor will enter the institute details and if the institute details are incomplete the system will through an error to tell the actor there are some required information that is not complete, else the system will save the new institute information inside the database and return to institute list table. Then the actor can select edit, delete or view to change the institute details

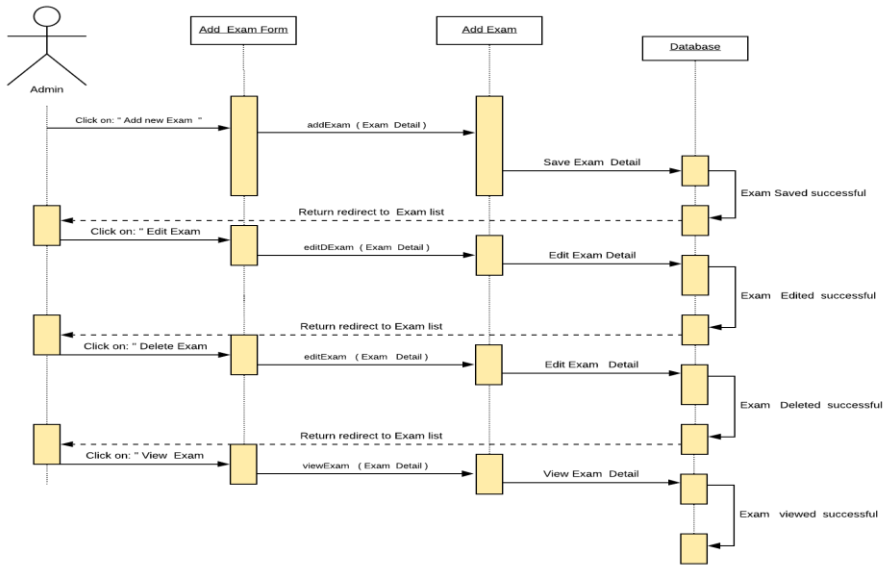


Figure 4. 35 Manage Exam Sequence Diagram

19. Manage GPA Sequence Diagram

The sequence diagram below shows the actor who will manage GPA. The actor can add GPA, edit GPA, view GPA and delete GPA. Therefore the actor can select add new GPA form. Then the system will display a form that the actor will enter the gpa details and if the GPA details are incomplete the system will through an error to tell the actor there are some required information that is not complete, else the system will save the new GPA information inside the database and return to GPA list table. Then the actor can select edit, delete or view to change the GPA details

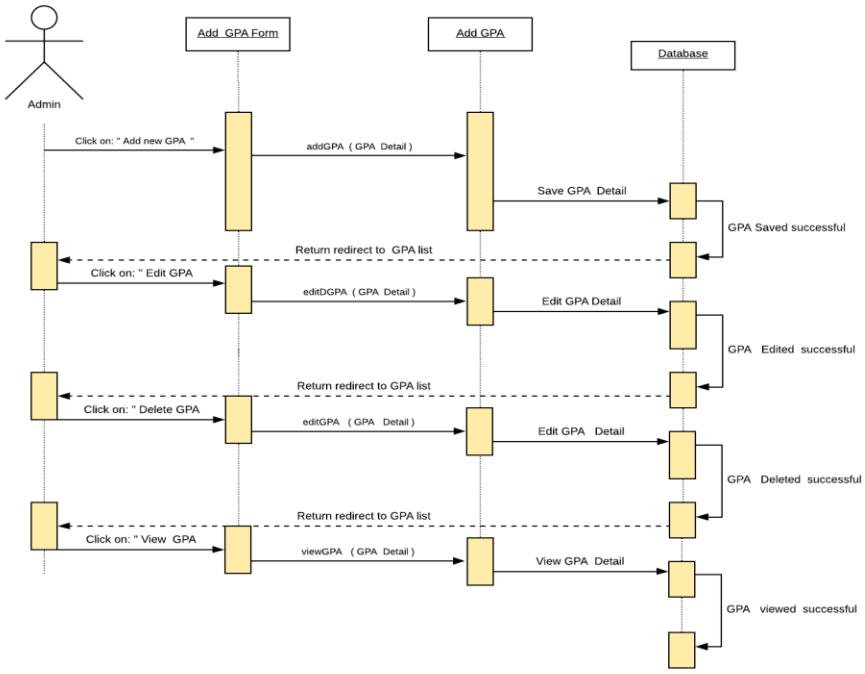


Figure 4. 36 Manage GPA Sequence Diagram

20. Manage Marks Sequence Diagram

The sequence diagram below shows the actor who will manage marks. The actor can add marks, edit marks, view marks, and delete marks. Therefore the actor can select add new marks form. Then the system will display a form that the actor will enter the marks details and if the marks details are incomplete the system will through an error to tell the actor there are some required information that is not complete, else the system will save the new marks information inside the database and return to marks list table. Then the actor can select edit, delete or view to change the marks details

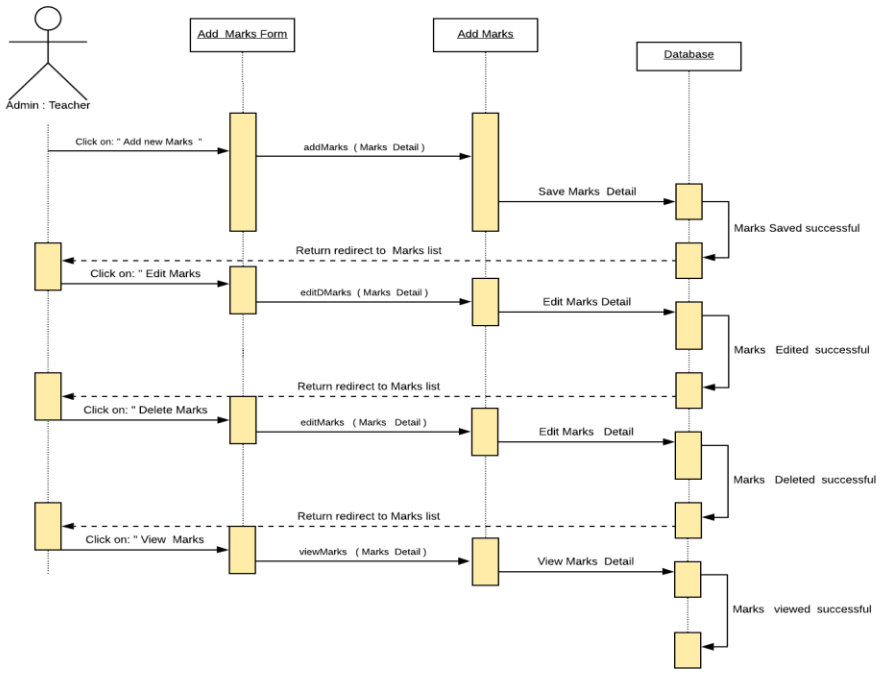


Figure 4. 37 Manage Mark Exam Sequence Diagram

21. Attendance Sequence Diagram

The sequence diagram below shows the actor who will manage marks. The actor can add marks, edit marks, view marks, and delete marks. Therefore the actor can select add new marks form. Then the system will display a form that the actor will enter the marks details and if the marks details are incomplete the system will through an error to tell the actor there are some required information that is not complete, else the system will save the new marks information inside the database and return to marks list table. Then the actor can select edit, delete or view to change the marks details

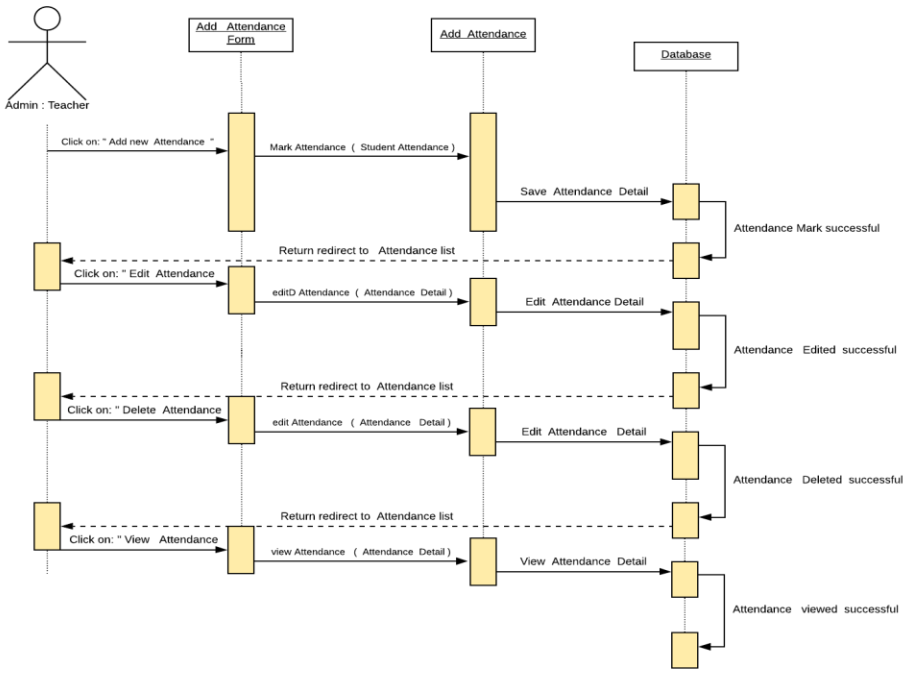


Figure 4. 38 Manage Attendance Sequence Diagram

22. Manage Timetable Sequence Diagram

The sequence diagram below shows the actor who will manage the timetable. The actor can create a timetable, edit timetable, view timetable, and delete timetable. Therefore the actor can select create a new timetable form. Then the system will display a form that the actor will enter the timetable details and if the timetable details are incomplete the system will through an error to tell the actor there are some required information that is not complete, else the system will save the new timetable information inside the database and return to timetable list table. Then the actor can select edit, delete or view to change the timetable details

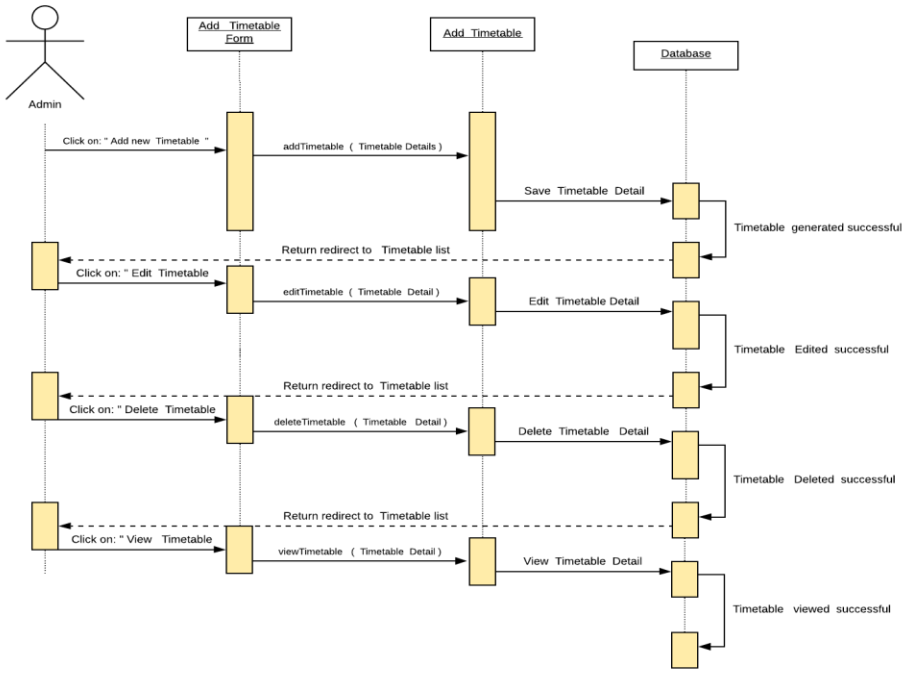


Figure 4. 39 Manage Timetable Sequence Diagram

23. Change Password Sequence Diagram

The sequence diagram below shows the actor who can change the password. The actor can change his/her password. Therefore the actor can click the change password form and the system will display the change password form and the actor can enter his/her current password and the system will check if the password is valid or not, if the password is valid then the system will update the password else the system will through an error password invalid return back.

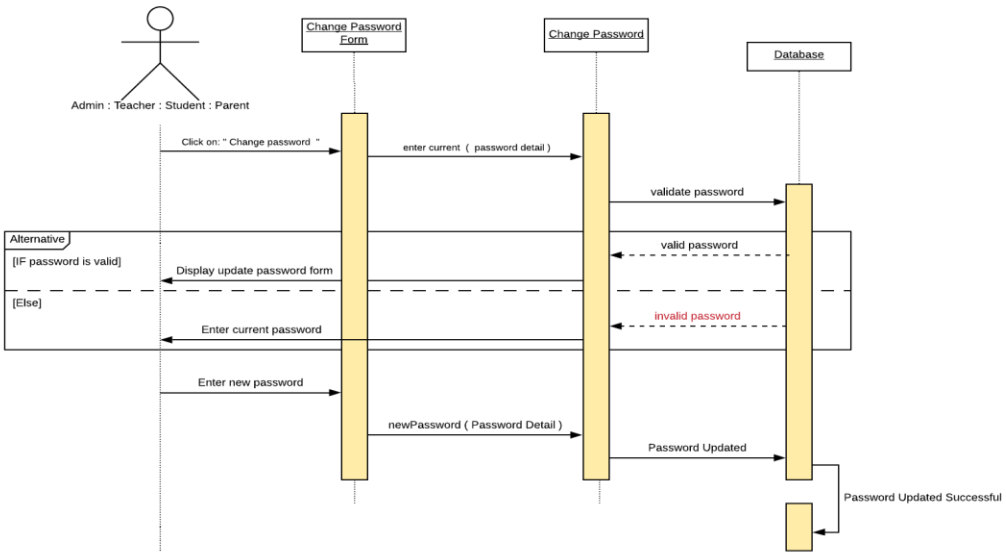


Figure 4. 40 Change Password Sequence Diagram

24. Profile Sequence Diagram

The sequence diagram below shows the actor who can view the profile. The actor can view his/her profile by click the profile menu form and the system will display the profile page and the actor can see his/her details

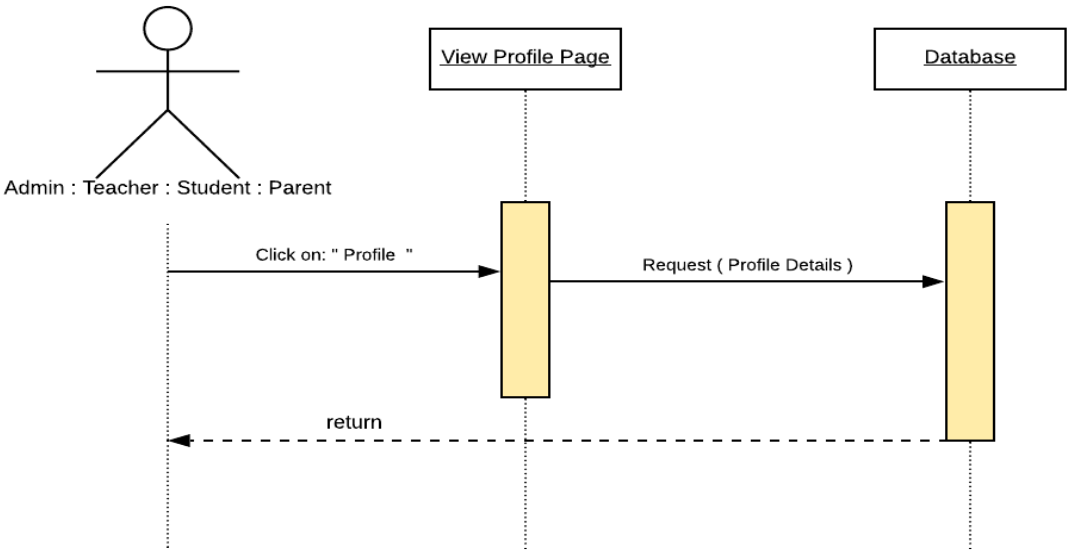


Figure 4. 41 Profile Sequence Diagram

25. Teacher Sequence Diagram

The sequence diagram below shows the actor teacher who can able to enter student exam marks, view his/her timetable, create an exam for students, generate exam results for students, and create exam question paper, print documents and handle his/her classes. The teacher actor will select any of his/her forms and the system will display that form for the actor teacher to able to proceed with his/her work. And the system will check if the actor teacher is working with the system accordingly or not if yes then the system will display all else the system will display an error message as a warning.

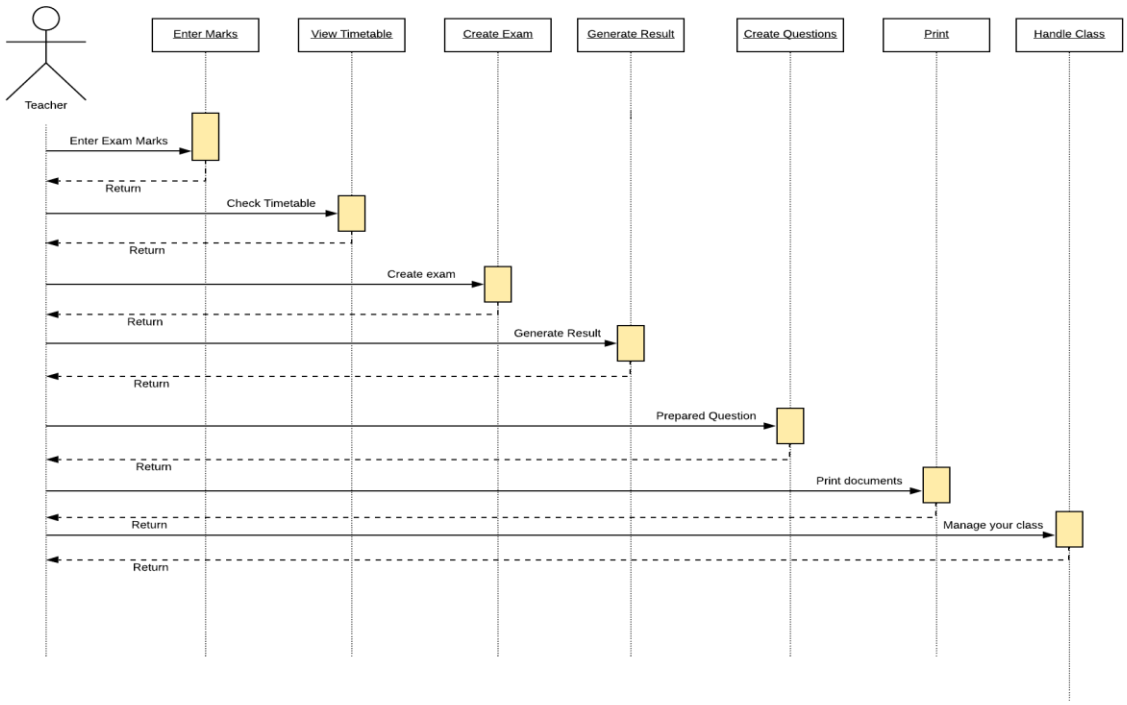


Figure 4. 42 Teacher Sequence Diagram

26. Student Sequence Diagram

The sequence diagram below shows the actor student who can able to view exam marks, view his/her timetable, view exam, view exam result, and view school transcript, print documents. The student actor will select any of his/her forms and the system will display that form for the actor student to able to proceed with his/her work. And the system will check if the actor student is working with the system accordingly or not if yes then the system will display all else the system will display an error message as a warning.

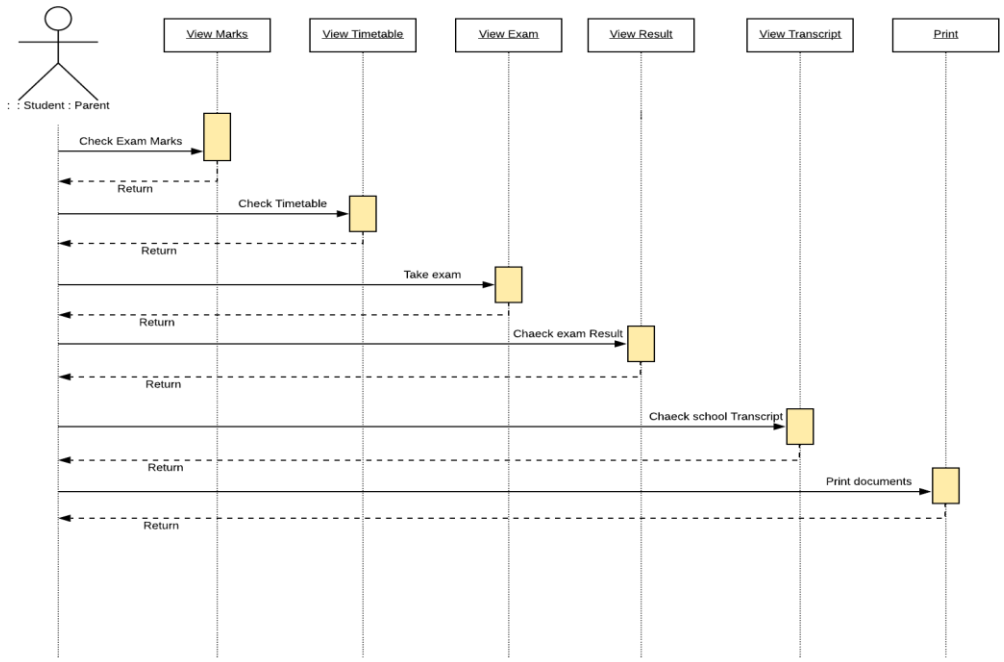


Figure 4. 43 Student Sequence Diagram

27. Logout Sequence Diagram

In the Sequence diagram below, it explains the process of Logout. In this activity the user selects the logout menu to exit the system, and then the system will display the login page again

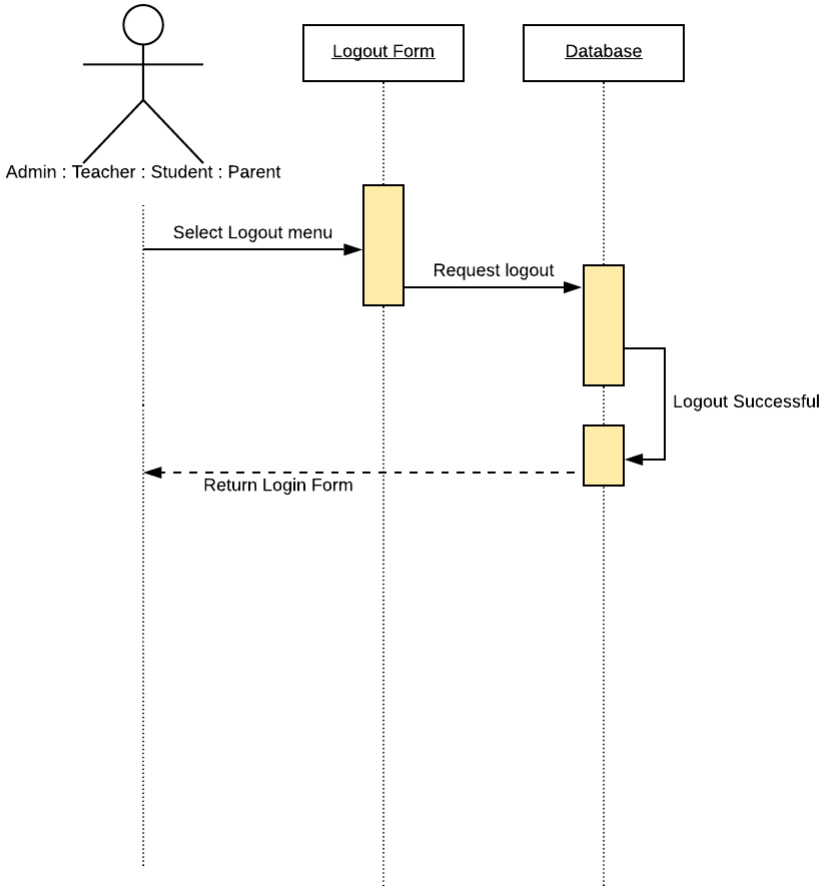


Figure 4. 44 Logout Sequence Diagram

4.8.4 Database Dictionaries

In a database management system file that defines the basic organization of a database. A data specification contains a list of all databases, the number of records in each file, and the names, types of each field. Most database management systems keep the data hidden from users to prevent them from accidentally destroying its contents.

Data dictionaries do not contain any actual data from the database, only bookkeeping information for managing it. Without a data dictionary, however, a database management system cannot access data from the database.

Following are the database tables of the Academic information system project:

Table 4. 27 Database Tables

No	Table name	Description
1	Admission	The admission table is the table for registering new students
2	Teachers	The teacher table is the table to register new teachers
3	Semester	The semester table is the table for register new semester
4	Course	The course table is the table for register new course
5	Degree	The degree table is the table for register new degree
6	Times	The times table is the table for register new times
7	Batches	The batches table is the table for register new batch
8	Attendance	The attendance table is the table for mark student and teacher attendance
9	Marks	The marks table is the table for entering exam marks
10	Classes	The classes table is the table for registering new class
11	Class room	The classroom table is the table for registering new classroom
12	Class schedule	The class schedule is the table for making class routine and timetable for students and teachers
13	Day	The day table is the table for registering new days
14	Faculty	The faculty table is the table for registering new faculty

Table 4. 28 Database Tables (Continued)		
15	Department	The department table is the table to registering a new department
16	Exam	The exam table is the table to registering new exams for student to take
17	Fees	The fees table is the table to registering student or semester fee
18	Fee structure	The fee structure table is the table for structuring student, semester and admission fee
19	Gpa	The gpa table is the table to enter the value of the current or existing gpa for student.
20	Message	The message table is the table to communicate between admin, teacher, student and parent
21	Institute	The institute table is the table for registering the school detail
22	Invoice	The invoice table is the table for holding new invoice from the transaction
23	Invoice details	The invoice detail table is the table that holds the entire details of the invoice transaction
24	Merit list	The merit list is the table for publication of student results
25	Question	The question table is the table for generating exam question papers
26	Role	The role table is the table for giving access to users with their specific roles
27	Rolls	The rolls table is the table that generate student roll numbers and username/ password of the specific student
28	Salary	The salary table is the table for paying teacher monthly salary
29	Shift	The shift is the table for adding new shift in shift table
30	Student fee	The student fee is the table that hold the student payment of fees
31	Transaction	The transaction table is the that stored al the transaction from the teacher and student
32	User	The user table is the table for specific users that can able to access the system and whole.

4.8.5 Data Specifications

In this step, the researcher will create a database design.

The following are the database tables with field names, data types, length, key field, and description.

1. Admission table

The information related to students is stored in this table. This gives various information while students interact with its account.

Table 4. 29 Admission Table

Field Name	Data Types	Constrains		Description
	Length	Null	Key	
Student_id	Integer (11)	Not null	PK	This is the id of the student and its primary key
First_name	String(50)	Not null		This is the first name of the student
Last_name	String (50)	Not null		This is last name of the student
Father_name	String (100)	Not null		This is the father name of the student
Mother_name	String (100)	Not null		This is the mother name of the student
Gender	Tiny integer (1)	Not null		This is the gender of the student
Dob	Date	Not null		This is the birth date of the student
Father_phone	Number (15)	Not null		This is the father phone of the student
Email	String (50)	Not null	Unique	This is the email of the student
Phone	Number (15)	Not null		This is the phone number of the student
Permanent_address	Text	Not null		This is the permanent address of the student
Current_address	Text	Not null		This is the current address of the student
Passport	String (30)	Not null	Unique	This is the passport number of the student
Nationality	String (50)	Not null		This is the nationality of the student
Status	Tiny integer	Not null		This is the status of the student

Table 4. 30 Admission Table (Continued **)**				
Dateregistered	Date	Not null		This is the date register of the student
User_id	Integer (11)	Not null	FK	This is the user id from the user table as the foreign key in the admission table
Semester_id	Integer (11)	Not null	FK	This is the semester id from the semester table as the foreign key in the admission table
Degree_id	Integer (11)	Not null	FK	This is the degree id from the degree table as foreign key in admission table
Class_code	String (10)	Not null	FK	This is the class code from the classes table as foreign key in admission table
Batch_id	Integer (11)	Not null	FK	This is the batch id from the batch table as foreign key in admission table
Faculty_id	Integer (11)	Not null	FK	This is the faculty id from the faculty table as foreign key in admission table
Department_id	Integer (11)	Not null	FK	This is the department id from the department table as foreign key in admission table
Image	String (30)	Not null		This is the image of the student
Deleted_at	Timestamp	Null		This is the deleted at timestamp
Created_at	Timestamp	Null		This is the created at timestamp
Updated_at	Timestamp	Null		This is the deleted at timestamp

2. Teacher table

The information related to the teacher is stored in this table. This gives various information while the teacher interacts with its account.

Table 4. 31 Teacher Table

Field Name	Data Types	Constrains		Description
	Length	Null	Key	
Teacher_id	Integer (11)	Not null	PK	This is the id of the teacher and its primary key
First_name	String(50)	Not null		This is the first name of the teacher
Last_name	String (50)	Not null		This is last name of the teacher
Gender	Tiny integer (1)	Not null		This is the gender of the teacher
Dob	Date	Not null		This is the birth date of the teacher
Email	String (50)	Not null	Unique	This is the email of the teacher
Phone	Number (15)	Not null		This is the phone number of the teacher
Permanent_address	Text	Not null		This is the permanent address of the teacher
Current_address	Text	Not null		This is the current address of the teacher
Passport	String (30)	Not null	Unique	This is the passport number of the teacher
Nationality	String (50)	Not null		This is the nationality of the teacher
Status	Tiny integer	Not null		This is the status of the teacher
Dateregistered	Date	Not null		This is the date register of the teacher
User_id	Integer (11)	Not null	FK	This is the user id from the user table as foreign key in teacher table
Semester_id	Integer (11)	Not null	FK	This is the semester id from the semester table as foreign key in teacher table
Faculty_id	Integer (11)	Not null	FK	This is the faculty id from the faculty table as foreign key in teacher table
Department_id	Integer (11)	Not null	FK	This is the department id from the department table as foreign key in teacher table
Image	String (30)	Not null		This is the image of the student
Deleted_at	Timestamp	Null		This is the deleted at timestamp
Created_at	Timestamp	Null		This is the created at timestamp
Updated_at	Timestamp	Null		This is the deleted at timestamp

3. Attendance table

The information related to attendance is stored in this table. This gives various information while the teacher interacts with its account.

Table 4. 32 Attendance Table

Attribute	Data Types	Constrains		Description
	Length	Null	Key	
Attendance_id	Integer (11)	Not null	PK	This is the attendance id and its primary key
Student_id	Integer (11)	Not null	FK	This is the student id from the admission table foreign key in attendance table
Class_id	Integer (11)	Not null	FK	This is the class id from the class table foreign key in attendance table
Course_id	Integer (11)	Not null	FK	This is the course id from the course table foreign key in attendance table
Teacher_id	Integer (11)	Not null	FK	This is the teacher id from the teacher table foreign key in attendance table
Semester_id	Integer (11)	Not null	FK	This is the semester id from the semester table foreign key in attendance table
Attendance_date	date	Not null	Unique	This is the attendance date
Edit_date	Date	Null		This is the attendance edit date
Day	String (9)	Not null		This is the day in attendance table
Month	String (10)	Not null		This is the month in attendance table
Year	integer (10)	Not null		This is the year in attendance table
Attendance_status	Tiny integer (1)	Not null		This is the attendance status
Deleted_at	Timestamp	Null		This is the deleted at timestamp

Created_at	Timestamp	Null		This is the created at timestamp
Updated_at	Timestamp	Null		This is the deleted at timestamp

4. Classes table

The information related to classes is stored in this table.

Table 4. 33 Classes Table

Field Name	Data Types	Constrains		Description
	Length	Null	Key	
Class_id	Integer (11)	Not null	PK	This is the class id and its primary key
Class_name	string (10)	Not null		This is the class name in class table
Class_code	string (15)	Not null	Unique	This is the class code in class table
Department_id	Integer (11)	Not null	FK	This is the department id from the department table foreign key in classes table
status	Tiny integer (1)	Not null		This is the classes status
Deleted_at	Timestamp	Null		This is the deleted at timestamp
Created_at	Timestamp	Null		This is the created at timestamp
Updated_at	Timestamp	Null		This is the deleted at timestamp

5. Class Room table

The information related to a classroom is stored in this table.

Table 4. 34 Classroom Table

Field Name	Data Types	Constrains		Description
	Length	Null	Key	
Classroom_id	Integer (11)	Not null	PK	This is the classroom id and its primary key
Classroom_name	string (10)	Not null		This is the classroom name in class table
Classroom_code	string (15)	Not null	Unique	This is the classroom code in class table
Classroom_description	string(150)	Not null		This is the classroom description
Classroom_status	Tiny integer (1)	Not null		This is the classes status
Deleted_at	Timestamp	Null		This is the deleted at timestamp
Created_at	Timestamp	Null		This is the created at timestamp
Updated_at	Timestamp	Null		This is the deleted at timestamp

6. Class Schedule table

The information related to the class schedule is stored in this table.

Table 4. 35 Class Schedule Table

Field Name	Data Types	Constrains		Description
	Length	Null	Key	
Class_schedule_id	Integer (11)	Not null	PK	This is the class schedule id and its primary key
Course_id	Integer (11)	Not null	FK	This is the course id in course table foreign key in class schedule table
Class_id	Integer (11)	Not null	FK	This is the class id in course table foreign key in class schedule table
Level_id	Integer (11)	Not null	FK	This is the level id in level table foreign key in class schedule table

Shift_id	Integer (11)	Not null	FK	This is the shift id in shift table foreign key in class schedule table
Classroom_id	Integer (11)	Not null	FK	This is the classroom id in classroom table foreign key in class schedule table
Batch_id	Integer (11)	Not null	FK	This is the batch id in batch table foreign key in class schedule table
Day_id	Integer (11)	Not null	FK	This is the day id in day table foreign key in class schedule table
Time_id	Integer (11)	Not null	FK	This is the time id in time table foreign key in class schedule table
Semester_id	Integer (11)	Not null	FK	This is the schedule id in schedule table foreign key in class schedule table
Degree_id	Integer (11)	Not null	FK	This is the degree id in degree table foreign key in class schedule table
Faculty_id	Integer (11)	Not null	FK	This is the faculty id in faculty table foreign key in class schedule table
Department_id	Integer (11)	Not null	FK	This is the department id in shift table foreign key in class schedule table
Teacher_id	Integer (11)	Not null	FK	This is the teacher id in teacher table foreign key in class schedule table
Start_date	Date	Not null		This is the start date of class schedule
End_date	Date	Not null		This is the end date of class schedule
Schedule_status	Tiny integer (1)	Not null		This is the status of class schedule
Deleted_at	Timestamp	Null		This is the deleted at timestamp
Created_at	Timestamp	Null		This is the created at timestamp
Updated_at	Timestamp	Null		This is the deleted at timestamp

7. Course table

It stored the different types of courses along with other related informations such as class id, semester id, and department id.

Table 4. 36 Course Table

Field Name	Data Types	Constrains		Description
	Length	Null	Key	
Course_id	Integer (11)	Not null	PK	This is the course id and its primary key
Course_code	string (11)	Not null	Unique	This is the course code
Course_name	string (30)	Not null		This is the course name
Class_id	Integer (11)	Not null	FK	This is the class id in class table foreign key in course table
Department_id	Integer (11)	Not null	FK	This is the department id in shift table foreign key in department table
Semester_id	Integer (11)	Not null	FK	This is the semester id in semester table foreign key in course table
Gradesystem	Integer (11)	Not null	FK	This is the grade system of course
Total_mark	Integer (11)	Not null	FK	This is the full marks
Total_full	Integer (11)	Not null	FK	This is the pass marks
Description	String (190)	Null		This is the description
Course_status	Tiny integer (1)	Not null		This is the course status of course table
Deleted_at	Timestamp	Null		This is the deleted at timestamp
Created_at	Timestamp	Null		This is the created at timestamp
Updated_at	Timestamp	Null		This is the deleted at timestamp

8. Day table

The information related to the day is stored in this table.

Table 4. 37 Day table

Field Name	Data Types	Constrains		Description
	Length	Null	Key	
Day_id	Integer (11)	Not null	PK	This is the day id and its primary key
Day_name	string (10)	Not null	Unique	This is the day name in day table
Deleted_at	Timestamp	Null		This is the deleted at timestamp
Created_at	Timestamp	Null		This is the created at timestamp
Updated_at	Timestamp	Null		This is the deleted at timestamp

9. Degree table

The information related to the degree is stored in this table.

Table 4. 38 Degree table

Field Name	Data Types	Constrains		Description
	Length	Null	Key	
Degree_id	Integer (11)	Not null	PK	This is the day id and its primary key
Degree_name	string (10)	Not null	Unique	This is the degree name in degree table
Semester_id	String (10)	Not null	FK	This is the semester id from semester table foreign key in degree table
Description	Text	Null		This is the description in degree table
Status	Tiny integer (1)	Not null		This is the status in degree table
Deleted_at	Timestamp	Null		This is the deleted at timestamp

Created_at	Timestamp	Null		This is the created at timestamp
Updated_at	Timestamp	Null		This is the deleted at timestamp

10. Exam table

The information related to an exam is stored in this table.

Table 4. 39 Exam table

Field Name	Data Types	Constrains		Description
	Length	Null	Key	
Exam_id	Integer (11)	Not null	PK	This is the exam id and its primary key
Exam_code	string (10)	Not null	Unique	This is the exam code in exam table
Class_id	Integer (11)	Not null	FK	This is the class id from class table foreign key in exam table
Department_id	Integer (11)	Not null	FK	This is the department id from department foreign key in exam table
Type	String (11)	Null		This is the type in exam table
Exam_date	Date	Not null		This is the exam date in exam table
Deleted_at	Timestamp	Null		This is the deleted at timestamp
Created_at	Timestamp	Null		This is the created at timestamp
Updated_at	Timestamp	Null		This is the deleted at timestamp

11. Faculty table

The information related to faculty is stored in this table.

Table 4. 40 Faculty table

Field Name	Data Types	Constrains		Description
	Length	Null	Key	
Faculty_id	Integer (11)	Not null	PK	This is the faculty id and its primary key
Faculty_name	string (30)	Not null		This is the faculty name in faculty table
Faculty_code	String (11)	Not null	Unique	This is the faculty code in faculty table
Faculty_status	Tiny integer (1)	Not null		This is the faculty status in faculty table
Deleted_at	Timestamp	Null		This is the deleted at timestamp
Created_at	Timestamp	Null		This is the created at timestamp
Updated_at	Timestamp	Null		This is the deleted at timestamp

12. Department table

The information related to the department is stored in this table.

Table 4. 41 Department table

Field Name	Data Types	Constrains		Description
	Length	Null	Key	
Department_id	Integer (11)	Not null	PK	This is the department id and its primary key
Department_name	string (10)	Not null		This is the department name in department table
Faculty_id	Integer (10)	Not null	FK	This is the faculty id from faculty table foreign key in department table
Department_code	Integer (11)	Not null	Unique	This is the department code
Description	Text	Null		This is the description in department table
Status	Tiny integer (1)	Not null		This is the status in degree table
Deleted_at	Timestamp	Null		This is the deleted at timestamp
Created_at	Timestamp	Null		This is the created at timestamp

Updated_at	Timestamp	Null		This is the deleted at timestamp

13. Fee structure table

The information related to fee structure is stored in this table.

Table 4. 42 Fee structure table

Field Name	Data Types	Constrains		Description
	Length	Null	Key	
Fee_structure_id	Integer (11)	Not null	PK	This is the fee structure id and its primary key
Semester_id	Integer (11)	Not null	FK	This is the faculty name in faculty table
Degree_id	Integer (11)	Not null	FK	This is the faculty code in faculty table
Faculty_id	Integer (11)	Not null	FK	This is the faculty status in faculty table
Department_id	Integer (11)	Not null	FK	This is the department id from department table foreign key in fee structure
AdmissionFee	Float (8,2)	Not null		This is the admission fee
SemesterFee	Float (8,2)	Not null		This is the semester fee
Deleted_at	Timestamp	Null		This is the deleted at timestamp
Created_at	Timestamp	Null		This is the created at timestamp
Updated_at	Timestamp	Null		This is the deleted at timestamp

14. GPA table

The information related to GPA is stored in this table.

Table 4. 43 GPA table

Field Name	Data Types	Constrains		Description
	Length	Null	Key	
Gpa_id	Integer (11)	Not null	PK	This is the fee structure id and its primary key
For	Integer (10)	Not null		This is the for in table gpa
Gpa	Float (8,2)	Not null		This is the gpa in gpa table
Grade	String (11)	Not null		This is the grade in gpa table
Mark_from	Integer (11)	Not null		This is the mark from in gpa table
Mark_to	Integer (11)	Not null		This is the mark to in gpa table
Deleted_at	Timestamp	Null		This is the deleted at timestamp
Created_at	Timestamp	Null		This is the created at timestamp
Updated_at	Timestamp	Null		This is the deleted at timestamp

15. Message table

The information related to a message is stored in this table.

Table 4. 44 Message table

Field Name	Data Types	Constrains		Description
	Length	Null	Key	
Message_id	Integer (11)	Not null	PK	This is the message id and its primary key
User_id	Integer (10)	Not null	FK	This is the user id from table user foreign key in message table
Subject	String (255)	Not null		This is the subject in message table
Body	Text	Not null		This is the body in message table
Status	Tiny integer (1)	Not null		This is the mark from in gpa table
Deleted_at	Timestamp	Null		This is the deleted at timestamp

Created_at	Timestamp	Null		This is the created at timestamp
Updated_at	Timestamp	Null		This is the deleted at timestamp

16. Institute table

The information related to the institute is stored in this table.

Table 4. 45 Institute table

Field Name	Data Types	Constrains		Description
	Length	Null	Key	
Institute_id	Integer (11)	Not null	PK	This is the institute id and its primary key
Institute_name	String (100)	Not null		This is the institute name
Establish	Date	Not null		This is the establish year from institute table
Email	String (100)	Not null		This is the email from institute table
Web	Text	Null		This is the web from the institute table
Phone no.	Number (15)	Not null		This is the phone number in institute table
Address	Text	Not null		This is the address from institute table
Image	String (40)	Not null		This is the logo of the institute from institute table
Deleted_at	Timestamp	Null		This is the deleted at timestamp
Created_at	Timestamp	Null		This is the created at timestamp
Updated_at	Timestamp	Null		This is the deleted at timestamp

17. Invoice table

The information related to the invoice is stored in this table.

Table 4. 46 invoice table

Field Name	Data Types	Constrains		Description
	Length	Null	Key	
Invoice_id	Integer (11)	Not null	PK	This is the invoice id and its primary key
Deleted_at	Timestamp	Null		This is the deleted at timestamp
Created_at	Timestamp	Null		This is the created at timestamp
Updated_at	Timestamp	Null		This is the deleted at timestamp

18. Invoice Detail table

The information related to invoice detail is stored in this table.

Table 4. 47 invoice detail table

Field Name	Data Types	Constrains		Description
	Length	Null	Key	
Invoice_detail_id	Integer (11)	Not null	PK	This is the invoice detail id and its primary key
Invoice_id	Integer (11)	Not null	FK	This is the invoice id from invoice table foreign key in invoice detail table
Student_id	Integer (11)	Not null	FK	This is the student id from admission table foreign key in invoice detail table
Transaction_id	Integer (11)	Not null	FK	This is the transaction id from transaction table foreign key in invoice detail table
Deleted_at	Timestamp	Null		This is the deleted at timestamp
Created_at	Timestamp	Null		This is the created at timestamp
Updated_at	Timestamp	Null		This is the deleted at timestamp

19. Marks table

The information related to marks is stored in this table.

Table 4. 48 Marks table

Attribute	Data Types	Constrains		Description
	Length	Null	Key	
Mark_id	Integer (11)	Not null	PK	This is the mark id and its primary key
Class_id	Integer (11)	Not null	FK	This is the class id from class table foreign key in marks table
Department_id	Integer (11)	Not null	FK	This is the department id from department table foreign key in marks table
Shift_id	Integer (11)	Not null	FK	This is the shift id from shift table foreign key in marks table
Roll_no	Integer (20)	Not null	FK	This is the roll number from rolls table foreign key in marks table
Exam_id	Integer (11)	Not null	FK	This is the exam id from exam table foreign key in marks table
Course_id	Integer (11)	Not null	FK	This is the course id from course table foreign key in marks table
Total	Integer (11)	Not null		This is the total from marks table
Grade	String (2)	Not null		This is the grade from marks table
Point	Float (8,2)	Not null		This is the point from marks table
Absent	Tiny integer (1)	Not null		This is the absent from marks table
Deleted_at	Timestamp	Null		This is the deleted at timestamp
Created_at	Timestamp	Null		This is the created at timestamp
Updated_at	Timestamp	Null		This is the deleted at timestamp

20. Merit list table

The information related to the merit list is stored in this table.

Table 4. 49 Merit list table

Field Name	Data Types	Constrains		Description
	Length	Null	Key	
Merit_id	Integer (11)	Not null	PK	This is the merit id and its primary key
Roll_no	Integer (20)	Not null	FK	This is the roll no from rolls table foreign key in merit list table
Exam_id	Integer (11)	Not null	FK	This is the exam id from exam table foreign key in merit list table
Class_id	Integer (11)	Not null	FK	This is the class id from class table foreign key in merit list table
Department_id	Integer (11)	Not null	FK	This is the department id from department table foreign key in merit list table
Batch_id	Integer (11)	Not null	FK	This is the batch id from merit list table foreign key in marks table
TotalNo	Integer (11)	Not null		This is the total from merit list table
Grade	String (2)	Not null		This is the grade from marks table
Point	Float (8,2)	Not null		This is the point from marks table
Deleted_at	Timestamp	Null		This is the deleted at timestamp
Created_at	Timestamp	Null		This is the created at timestamp
Updated_at	Timestamp	Null		This is the deleted at timestamp

21. Question table

This table will store all exam questions paper and generated it for students to able to take an exam during examination time.

Table 4. 50 Question table

Field Name	Data Types	Constrains		Description
	Length	Null	Key	
Question_id	Integer (11)	Not null	PK	This is the question id and its primary key

Class_id	Integer (11)	Not null	FK	This is the class id from class table foreign key in question table
Course_id	Integer (11)	Not null	FK	This is the course id from course table foreign key in question table
Session	Date	Not null	FK	This is the session from question table
Chapter	String (255)	Not null		This is the chapter from question table
Question_name	String (255)	Not null		This is the question name from question table
Question_type	Integer (11)	Not null		This is the question type from question table
Choices	String (255)	Not null		This is the choice from question table
Answer	String (255)	Not null		This is the answer from question table
Type	String (22)	Not null		This is the type from question table
Level	String (150)	Not null		This is the level from question table
Logo	String (150)	Not null		This is the logo of the institute
Quize_name	String (190)	Not null		This is the quiz name from question table
Point	Float (8,2)	Not null		This is the point from marks table
Deleted_at	Timestamp	Null		This is the deleted at timestamp
Created_at	Timestamp	Null		This is the created at timestamp
Updated_at	Timestamp	Null		This is the deleted at timestamp

22. Role table

The information related to the role is stored in this table.

Table 4. 51 Role table

Field Name	Data Types	Constrains		Description
	Length	Null	Key	
Role_id	Integer (11)	Not null	PK	This is the role id and its primary key
Roll_name	String (20)	Not null	Unique	This is the role name in role table
Deleted_at	Timestamp	Null		This is the deleted at timestamp
Created_at	Timestamp	Null		This is the created at timestamp
Updated_at	Timestamp	Null		This is the deleted at timestamp

23. Roll table

The information related to the roll is stored in this table.

Table 4. 52 Roll table

Field Name	Data Types	Constrains		Description
	Length	Null	Key	
Roll_id	Integer (11)	Not null	PK	This is the roll id and its primary key
Student_id	Integer (11)	Not null	FK	This is the student id from admission table foreign key in roll table
Semester_id	Integer (11)	Not null	FK	This is the semester id from semester table foreign key in roll table
Username	Integer (20)	Not null	Unique	This is the username in roll table
password	Integer (20)	Not null		This is the password in roll table
Login_time	Timestamp	Not null		This is the login time in roll table
Logout_time	Timestamp	Not null		This is the logout time in roll table
isOnline	Tiny integer (2)	Not null		This is the isonline in roll table

Ip_address	String (50)	Not null		This is the ip address in roll table
Deleted_at	Timestamp	Null		This is the deleted at timestamp
Created_at	Timestamp	Null		This is the created at timestamp
Updated_at	Timestamp	Null		This is the deleted at timestamp

24. Salary table

The information related to salary is stored in this table.

Table 4. 53 Salary table

Field Name	Data Types	Constrains		Description
	Length	Null	Key	
Salary_id	Integer (11)	Not null	PK	This is the salary id and its primary key
Salary_type	float (8,2)	Not null		This is the salary type
Teacher_id	Integer (11)	Not null	FK	This is the teacher id from teacher table foreign key in roll table
Paid_date	Date	Not null		This is the paid date
Deleted_at	Timestamp	Null		This is the deleted at timestamp
Created_at	Timestamp	Null		This is the created at timestamp
Updated_at	Timestamp	Null		This is the deleted at timestamp

25. Semester table

The information related to the semester is stored in this table.

Table 4. 54 Semester table

Field Name	Data Types	Constrains		Description
	Length	Null	Key	
Semester_id	Integer (11)	Not null	PK	This is the semester id and its primary key
Semester_name	String (20)	Not null		This is the semester name from semester table
Semester_code	Integer (11)	Not null	Unique	This is the semester code from semester table
Semester_duration	String (20)	Not null		This is the semester duration in semester table
Semester_description	Text	Null		This is the semester description in table
Deleted_at	Timestamp	Null		This is the deleted at timestamp
Created_at	Timestamp	Null		This is the created at timestamp
Updated_at	Timestamp	Null		This is the deleted at timestamp

26. Shift table

The information related to the shift is stored in this table.

Table 4. 55 Shift table

Field Name	Data Types	Constrains		Description
	Length	Null	Key	
Shift_id	Integer (11)	Not null	PK	This is the shift id and its primary key
Shift_name	String (10)	Not null		This is the shift name in shift table
Deleted_at	Timestamp	Null		This is the deleted at timestamp
Created_at	Timestamp	Null		This is the created at timestamp
Updated_at	Timestamp	Null		This is the deleted at timestamp

27. Student fees table

The information related to student fees is stored in this table.

Table 4. 56 Student fee table

Field Name	Data Types	Constrains		Description
	Length	Null	Key	
Student_fee_id	Integer (11)	Not null	PK	This is the student fee id and its primary key
Student_id	Integer (11)	Not null	FK	This is the student id from admission table foreign key in student fees table
Fee_id	Integer (11)	Not null	FK	This is the fee id from fees table foreign key in student fees table
Degree_id	Integer (11)	Not null	FK	This is the degree id from degree table foreign key in student fees table
Amount	Float (8,2)	Not null		This is the amount in student fee table
Deleted_at	Timestamp	Null		This is the deleted at timestamp
Created_at	Timestamp	Null		This is the created at timestamp
Updated_at	Timestamp	Null		This is the deleted at timestamp

28. Times table

The information related to time is stored in this table.

Table 4. 57 Times table

Field Name	Data Types	Constrains		Description
	Length	Null	Key	
Time_id	Integer (11)	Not null	PK	This is the times id and its primary key
Time_name	String (20)	Not null		This is the time name from

				time table
Shift_id	Integer (11)	Not null	FK	This is the shift id from shift table foreign key in time table
Deleted_at	Timestamp	Null		This is the deleted at timestamp
Created_at	Timestamp	Null		This is the created at timestamp
Updated_at	Timestamp	Null		This is the deleted at timestamp

29. Transaction table

The information related to the transaction is stored in this table.

Table 4. 58 Transaction table

Field Name	Data Types	Constrains		Description
	Length	Null	Key	
Transaction_id	Integer (11)	Not null	PK	This is the transaction id and its primary key
Student_id	Integer (11)	Not null	FK	This is the student id from admission table foreign key in transaction table
Fee_id	Integer (11)	Not null	FK	This is the fee id from fees table foreign key in transaction table
Semester_id	Integer (11)	Not null	FK	This is the semester id from semester table foreign key in transaction table
User_id	Integer (11)	Not null	FK	This is the user id from user table foreign key in transaction table
Paid_amount	Float (8,2)	Not null		This is the paid amount in transaction table
Description	Text	Not null		This is the description in transaction table
Transaction_date	Date	Not null		This is the transaction date in transaction table
Deleted_at	Timestamp	Null		This is the deleted at timestamp

Created_at	Timestamp	Null		This is the created at timestamp
Updated_at	Timestamp	Null		This is the deleted at timestamp

30. User table

The information related to the user is stored in this table.

Table 4. 59 User table

Field Name	Data Types	Constrains		Description
	Length	Null	Key	
User_id	Integer (11)	Not null	PK	This is the user id and its primary key
Name	String (200)	Not null		This is the user name in user table
Email	String (100)	Not null	Unique	This is the email in user table
Role_id	Integer (11)	Not null	FK	This is the role id from role table foreign key in user table
Teacher_id	Integer (11)	Not null	FK	This is the teacher id from teacher table foreign key in user table
Password	String (200)	Not null		This is the password in user table
Deleted_at	Timestamp	Null		This is the deleted at timestamp
Created_at	Timestamp	Null		This is the created at timestamp
Updated_at	Timestamp	Null		This is the deleted at timestamp

4.9 Implementation phase & System Testing

In this stage, the developer consists of two crucial stages which help any other developer to use the layers of programming language that will implement to coding

stage, and the system testing stage will be after completing everything from the programming stage, use the black box testing stage before deploying the system.

The following are the steps carried out in the implementation phase along with system testing.

4.9.1 Coding

At this stage, the coding of the system design is carried out predefined. The coding is done by using the PHP framework Laravel, JavaScript, and phpMyAdmin MySQL as the database. The web-based software following the designs above.

4.9.2 System Testing

System testing is a critical aspect of Software Quality Assurance (SQA) and represents the ultimate review of specification, design, and coding. Testing is a process of executing a program with the intent of finding an error. A good system test is one that has a probability of finding an as-yet-undiscovered error. The purpose of testing is to identify and correct bugs in the developed system. Nothing is complete without testing. Testing is vital to the success of the system.

The testing phase used by the researcher is black-box testing by running the units, and then observe whether the results of the units under the desired process and to enhance performance, the automated system developed were tested at four levels, namely:

Unit Level	Module Level	Integration	Regression

Table 4. 60 "System Testing"

Level	Description
Unit	In unit testing, each module developed was tested to affirm it is error-free.
Module	Module Testing is done using the test cases prepared earlier. The module is defined during the time of design
Integration & System	This is used to verify the combining of the software modules and their functionality. It addresses the issues associated with the dual problems of verification and program construction. System testing is used to verify, whether the developed system meets the requirements.
Regression	Each modification in software impacts unmodified areas, which results in serious injuries to that software. So, the process of re-testing for rectification of errors due to modification is known as regression testing. Apart from these tests, there are some special tests conducted which are given below:
Peak Load Tests	This determines whether the new system will handle the volume of activities when the system is at the peak of its processing demand. The test has revealed that the new system developed is capable of handling the demands at peak time.
Storage Testing	This determines the capacity of the new system to store data on a disk or other files. The proposed system has the required storage space available because it had been spelled out in system requirements.
Performance Time	This test determines the length of the time used by the system to process data.

1. Testing the Administrators Actor System

Table 4. 61 Administrator

No	Process Design	Expected results	Results
1	Login as Administrator	Can able to access his/her dashboard	Pass/Ok
2	Add student	Can able to add student individually or bulk	Pass/Ok
3	Edit student	Can able to edit new or existing student	Pass/Ok
4	Update student	Can able to update existing student	Pass/Ok
5	Delete student	Can able to delete existing student	Pass/Ok
6	Search student	Can able to search existing student	Pass/Ok
7	Collect student fee	Can able to collect student fees	Pass/Ok
8	View student profile	Can able to view student profile	Pass/Ok
9	Assign student	Can able to assign a student to a class	Pass/Ok
10	Add teacher	Can able to add teacher individually or bulk	Pass/Ok
11	Edit teacher	Can able to edit new or existing teacher	Pass/Ok
12	Update teacher	Can able to update new or existing teacher	Pass/Ok
13	Delete teacher	Can able to delete existing teacher	Pass/Ok
14	Search teacher	Can able to search existing teacher	Pass/Ok
15	Pay salary	Can able to pay teacher salary	Pass/Ok
16	View teacher profile	Can able to view teacher profile	Pass/Ok
17	Assign teacher	Can able to assign a teacher to a class	Pass/Ok
18	Add classes	Can able to add a new class	Pass/Ok
19	Edit classes	Can able to edit an existing class	Pass/Ok
20	Update classes	Can able to update existing class	Pass/Ok

21	Delete classes	Can able to delete an existing class	Pass/Ok
22	Search classes	Can able to search existing class	Pass/Ok
23	Add course	Can able to add a course	Pass/Ok
24	Edit course	Can able to edit an existing course	Pass/Ok
25	Update course	Can able to update existing course	Pass/Ok
26	Delete course	Can able to delete existing course	Pass/Ok
27	Search course	Can able to search existing course	Pass/Ok
28	Add classroom	Can able to add classroom	Pass/Ok
29	Edit classroom	Can able to edit existing classroom	Pass/Ok
30	Update classroom	Can able to update existing classroom	Pass/Ok
31	Delete classroom	Can able to delete existing classroom	Pass/Ok
32	Search classroom	Can able to search existing classroom	Pass/Ok
33	Add degree	Can able to add a degree	Pass/Ok
33	Edit degree	Can able to edit existing degree	Pass/Ok
33	Update degree	Can able to update the existing degree	Pass/Ok
33	Delete degree	Can able to delete existing degree	Pass/Ok
33	Search degree	Can able to search existing degree	Pass/Ok
34	Add semester	Can able to add a semester	Pass/Ok
35	Edit semester	Can able to edit existing semester	Pass/Ok
36	Update semester	Can able to update existing semester	Pass/Ok
37	Delete semester	Can able to delete existing semester	Pass/Ok

38	Search semester	Can able to search existing semester	Pass/Ok
39	Add batch	Can able to add batch	Pass/Ok
40	Edit batch	Can able to edit an existing batch	Pass/Ok
41	Update batch	Can able to update existing batch	Pass/Ok
42	Delete batch	Can able to delete existing batch	Pass/Ok
43	Search batch	Can able to search existing batch	Pass/Ok
44	Add time	Can able to add new time	Pass/Ok
45	Edit time	Can able to edit existing time	Pass/Ok
46	Update time	Can able to update existing time	Pass/Ok
47	Delete time	Can able to delete existing time	Pass/Ok
48	Search time	Can able to search existing time	Pass/Ok
49	Add department	Can able to add department	Pass/Ok
50	Edit department	Can able to edit existing department	Pass/Ok
51	Update department	Can able to update existing department	Pass/Ok
52	Delete department	Can able to delete existing department	Pass/ok
53	Search department	Can able to search existing department	Pass/ok
54	Add faculty	Can able to add faculty	Pass/Ok
55	Edit faculty	Can able to edit existing faculty	Pass/Ok
56	Update faculty	Can able to update existing faculty	Pass/Ok
57	Delete faculty	Can able to delete existing faculty	Pass/Ok
58	Search faculty	Can able to search existing faculty	Pass/Ok

59	Add day	Can able to add days	Pass/Ok
60	Edit day	Can able to edit existing day	Pass/Ok
61	Update day	Can able to update existing day	Pass/Ok
62	Delete day	Can able to delete existing day	Pass/Ok
63	Search day	Can able to search existing day	Pass/Ok
64	Mark attendance	Can able to mark new attendance	Pass/Ok
65	Edit attendance	Can able to edit existing attendance	Pass/Ok
66	Update attendance	Can able to update existing attendance	Pass/Ok
67	Delete attendance	Can able to delete existing attendance	Pass/Ok
68	Search attendance	Can able to search existing attendance	Pass/Ok
69	Print report	Can able to print new attendance or existing attendance report.	Pass/Ok
70	Send attendance	Can able to send attendance reports to parents, students, etc.	Pass/Ok
71	Add shift	Can able to add shift	Pass/Ok
72	Edit shift	Can able to edit existing shift	Pass/Ok
73	Update shift	Can able to update existing shift	Pass/Ok
74	Delete shift	Can able to delete existing shift	Pass/Ok
75	Search shift	Can able to search existing shift	Pass/Ok
76	Add role	Can able to add a role	Pass/ok
77	Edit role	Can able edit an existing role	Pass/Ok
78	Update role	Can able to update the existing role	Pass/ok
79	Search role	Can able to search existing role	Pass/Ok

80	Assign role	Can able to assign a role to users	Pass/Ok
81	Delete role	Can able to delete an existing role	Pass/Ok
82	Assign roll	Can able to assign roll to students	Pass/Ok
83	Make invoice	Can able to make invoice	Pass/Ok
84	Add fee structure	Can able to add fee structure for admission and semester	Pass/Ok
85	Edit fee structure	Can able to edit existing fee structure	Pass/Ok
86	Update fee structure	Can able to update the existing fee structure	Pass/Ok
87	Delete fee structure	Can bale to delete existing fee structure	Pass/Ok
88	Search fee structure	Can able to search existing fee structure	Pass/Ok
89	Add institute info	Can able to add institute details	Pass/Ok
90	Edit institute	Can able to edit existing institute detail	Pass/Ok
91	Update institute	Can able to update existing institute	Pass/Ok
92	Delete institute	Can able to delete existing institute	Pass/Ok
93	Search institute	Can able to search institute details	Pass/Ok
94	Generate timetable	Can able to generate student and teacher timetable	Pass/Ok
95	Edit timetable	Can able to edit timetable	Pass/Ok
96	Update timetable	Can able to update existing timetable	Pass/Ok
97	Delete timetable	Can able to delete existing timetable	Pass/Ok
98	Search timetable	Can able to search existing timetable	Pass/Ok
99	Print timetable	Can able to print, in four different formats: pdf, print, word, and excel	Pass/Ok

100	View Transaction	Can able to view transaction history	Pass/Ok
101	Add exam	Can able to add exam	Pass/Ok
102	Edit exam	Can able to edit the existing exam	Pass/Ok
103	Update exam	Can able to update existing exam	Pass/Ok
104	Delete exam	Can able to delete the existing exam	Pass/Ok
105	Search exam	Can able to search existing exam	Pass/Ok
106	Set exam	Can able to set an exam for students	Pass/Ok
107	Create a question	Can able to create question paper for exams	Pass/Ok
108	Edit question	Can able to edit existing question paper	Pass/Ok
109	Update question	Can able to update existing question paper	Pass/Ok
110	Delete question	Can able to delete existing question paper	Pass/Ok
111	Search question	Can able to search question papers	Pass/Ok
112	Print question	Can able to print question paper, download	Pass/Ok
113	Enter marks	Can able to enter exam marks for students	Pass/Ok
114	Edit marks	Can able to edit existing exam marks	Pass/Ok
115	Update marks	Can able to update existing exam marks	Pass/Ok
116	Delete marks	Can able to delete existing exam marks	Pass/Ok
117	Search marks	Can able to search existing student exam marks	Pass/Ok
118	Print mark sheet	Can able to print exam mark sheet of the student	Pass/Ok
119	Add GPA	Can able to add new GPA value	Pass/Ok
120	Edit GPA	Can able to edit existing GPA value	Pass/Ok

121	Update GPA	Can able to update existing GPA value	Pass/Ok
122	Delete GPA	Can able to delete existing GPA value	Pass/Ok
123	Search GPA	Can able to search existing GPA value	Pass/Ok
124	Publish result	Can able to publish exam result	Pass/Ok
125	Generate result	Can able to generate student result	Pass/Ok
126	Print result	Can able to print result in four formats, word, pdf, excel and print	Pass/Ok
127	Send result	Can able to send the result to students, parents, and guidance	Pass/Ok
128	Message	Can able to send a message to teachers, a students, and parents	Pass/Ok
129	Receive message	Can able to receive a message from students, teachers, and parents	Pass/Ok

2. Testing the Teacher Actor System

Table 4. 62 Teacher module system testing

No	Process Design	Expected results	Results
1	Login as Teacher	Can able to access to his/her dashboard	Pass/Ok
2	View profile	Can able to view his/her profile	Pass/Ok
3	Change password	Can able to change his/her password	Pass/Ok
4	Request forgotten pass	Can able to request his/her forgotten password	Pass/OK
5	Handle class	Can able to handle his/her classes	Pass/Ok
6	View class assign	Can able to view his/her assign class	Pass/Ok
7	Timetable	Can able to view his/her timetable	Pass/Ok
8	Print timetable	Can able to print his/her timetable	Pass/Ok

9	View salary	Can able to view his/her salary history	Pass/Ok
10	Mark attendance	Can able to mark new attendance	Pass/Ok
11	Edit attendance	Can able to edit existing attendance	Pass/Ok
12	Update attendance	Can able to update existing attendance	Pass/Ok
13	Search attendance	Can able to search attendance	Pass/Ok
14	Print attendance	Can able to print attendance report	Pass/Ok
15	Send attendance report	Can able to send attendance report	Pass/Ok
16	Create question	Can able to create question paper for exams	Pass/Ok
17	Edit question	Can able to edit existing question paper	Pass/Ok
18	Update question	Can able to update existing question paper	Pass/Ok
19	Search question	Can able to search question papers	Pass/Ok
20	Print question	Can able to print question paper, download	Pass/Ok
21	Enter marks	Can able to enter exam marks for students	Pass/Ok
22	Edit marks	Can able to edit existing exam marks	Pass/Ok
23	Update marks	Can able to update existing exam marks	Pass/Ok
24	Search marks	Can able to search existing student exam marks	Pass/Ok
25	Print mark sheet	Can able to print exam mark sheet of the student	Pass/Ok
26	Generate result	Can able to generate student result	Pass/Ok
27	Print result	Can able to print result in four formats, word, pdf, excel and print	Pass/Ok
28	Send result	Can able to send the result to students, parents and guidance	Pass/Ok

29	Message	Can able to send a message to teachers, students, and parents	Pass/Ok
30	Receive message	Can able to receive a message from students, teachers, and parents	Pass/Ok

3. Testing the Student Actor System

Table 4. 63 Student **module system testing**

No	Process Design	Expected results	Results
1	Login as Student	Can able to access to his/her dashboard	Pass/Ok
2	View profile	Can able to view his/her profile	Pass/Ok
3	Change password	Can able to change his/her password	Pass/Ok
4	Request forgotten pass	Can able to request his/her forgotten password	Pass/OK
5	View exam marks	Can able to view his/her exam marks	Pass/Ok
6	View result	Can able to view his/her exam result	Pass/Ok
7	View timetable	Can able to view his/her timetable	Pass/Ok
8	Print timetable	Can able to print his/her timetable	Pass/Ok
9	View fees	Can able to view his/her fee history	Pass/Ok
10	Print result	Can able to print his/her result	Pass/Ok
11	Print transcript	Can able to print his/her transcript	Pass/Ok
12	Send message	Can able to send a message and receive	Pass/Ok

4. Testing the Parent System Module

Table 4. 64 Parent module system testing

No	Process Design	Expected results	Results
1	Login as Parent	Can able to access to his/her dashboard	Pass/Ok
2	View profile	Can able to view his/her profile	Pass/Ok
3	Change password	Can able to change his/her password	Pass/Ok
4	Request forgotten pass	Can able to request his/her forgotten password	Pass/OK
5	View exam marks	Can able to view student exam marks	Pass/Ok
6	View result	Can able to view student exam result	Pass/Ok
7	View timetable	Can able to view student timetable	Pass/Ok
8	Print timetable	Can able to print student timetable	Pass/Ok
9	View fees	Can able to view student fee history	Pass/Ok
10	Print result	Can able to print student result	Pass/Ok
11	Print transcript	Can able to print student transcript	Pass/Ok
12	Send message	Can able to send a message and receive	Pass/Ok
13	Attendance	Can able to see student attendance report	Pass/Ok

4.10 System Specification

4.10.1 Hardware Specification

Processor	Intel Pentium IV and above
RAM	2 GB above
Hard Disk	80 GB HDD above

Table 4. 65 Hardware specification

4.10.2 Software Specification

Technology Implemented	Apache Server
Language Used	Php 7.3
Framework used	Laravel 6.7
Database	MySQL 5.5
User Interface Design	HTML, CSS, JAVASCRIPT
Web Browser	Mozilla, Google Chrome, Torch, Opera, IE8
Server	Wamp64 server

Table 4. 66 Software specification

4.11 Academic Information System (screenshots)

CHAPTER V
CONCLUSION

5.1 Conclusion

The system has developed with much care that it is free of errors and at the same time, it is efficient and less time-consuming. The important thing is that the system is robust and all events happened in real-time. Avoid malfunction from outsiders. It goes through all phases of the software development cycle. So, the information is accurate. Also, provision is provided for future developments in the system.

REFERENCES

Agnes, M., Jola, L., & Gaspersz, S. (2018). Academic Information System for Student (Case Study: Victory University of Sorong). *International Journal of Computer Applications*, *180*(43), 26–33. https://doi.org/10.5120/ijca2018917134

Aliyu, M. M. (2015). *Design and Implementation of Web – Based Academic Information System (Webais) for Nigerian Universities*. *April*.

Asrar-ul-haq, M., & Anwar, S. (2016). A systematic review of knowledge management and knowledge sharing : Trends , issues , and challenges A systematic review of knowledge management and knowledge sharing : Trends , issues , and challenges. *Cogent Business & Management*, *14*(1). https://doi.org/10.1080/23311975.2015.1127744

Bartosz Porebski, Karol Przystalski, L. N. (2011). *Building PHP Applications with Symfony™, CakePHP, and Zend® Framework*. Wiley Publishing. www.wiley.com

Dalkir, K. (2005). *Knowledge Management in Theory and Practice*. A catalogue record for this book is available from the British Library. https://erl.ucc.edu.gh/dspace/bitstream/123456789/2994/1/%5BDalkir_Kimiz%5D_Knowledge_Management_in_Theory_and_%28BookZZ.org%29.pdf

Darmalaksana, W., Ali Ramdhani, M., Cahyana, R., & Syakur Amin, A. (2018). Strategic Design of Information System Implementation at University.

International Journal of Engineering & Technology, *7*(2.29), 787. https://doi.org/10.14419/ijet.v7i2.29.14257

Dennis, A., Wixom, H. B., & Tegarden, D. (2015). Systems Analysis Design with UML Version 2.5: An Object-Oriented Approach. In *John Wiley & Sons*. www.visible.com

Eddy Tukamushaba Kurobuza, & Godfrey Ahumuza Ikaanga. (2008). *Introduction to Computer Software Applications*. 343.

Edition, F. (n.d.). *ETHICS IN INFORMATION*.

Hassan, S., & Gil-Garcia, J. R. (2011). Institutional Theory and E-Government Research. In *Handbook of Research on Public Information Technology*. https://doi.org/10.4018/978-1-59904-857-4.ch034

Herdiana. (2013). Implementing School Registering System. *Journal of Chemical Information and Modeling*, *53*(9), 1689–1699. https://doi.org/10.1017/CBO9781107415324.004

Hislop, D. (2018). *knowledge management in organizations*.

Is, W. (2011). Information Systems: Definitions and Components What Is an Information System? *Management Information Systems Lecture …*, *3*, 24–32.

Kaehler, B., & Grundei, J. (2019). *HR Governance A Theoretical Introduction*. *2019*(July 2018).

Kendall, K. E., & Kendall, J. E. (2008). *SYSTEMS ANALYSIS and DESIGN EIGHTH EDITION* (B. H. Eric Svendsen (ed.); 8th Editio). Kendall, Kenneth E.

Kendall, K. E., & Kendall, J. E. (2014). *Systems Analys And Design*.

Laudon, K. C., & Laudon, J. P. (2014). *Management Information Systems Managing The digital Firm Thirteen Edition Global Edition (SIB)*.

McCool, S. (2012). *Laravel Starter*. https://books.google.com.br/books?hl=pt-BR&lr=&id=pIBevFHNX3oC&oi=fnd&pg=PT5&dq=laravel&ots=B8qHd2QvW6&sig=qK4NS4F2YQwd-TF_G52ukNugQT4#v=onepage&q=laravel&f=false

Putra, D. A. D., Jasmi, K. A., Basiron, B., Huda, M., Maseleno, A., Shankar, K., & Aminudin, N. (2018). Tactical steps for e-government development. *International Journal of Pure and Applied Mathematics*, *119*(15).

Ramdhani, M. A., & Ainissyifa, H. (2014). Strategy of Technology Utilization in Supporting the Implementation of Knowledge Management in Educational Institution. *International Conference on Islam in Malay World IV*, *September*, 99–110.

Saputra, Pramana Yoga and Siahaan, D. O. (2014). Analisis dan Desain Sistem Informasi Akademik Politeknik Negeri Malang Menggunakan Metode Rapid Applica-tion Development (RAD). *Prosiding Seminar Nasional Manajemen Teknologi XX MMT-ITS, Surabaya, September 2018*.

Soediono, B., Mustofa, A., Informatika, T., & Nuswantoro, U. D. (2014). Analisis Dan Perancangan Sistem Informasi E-Education Berbasis Web Di Sma Pembangunan Mranggen. *Journal of Chemical Information and Modeling*, *53*(2004), 1–6. https://doi.org/10.1017/CBO9781107415324.004

Stair, R., & Reynolds, G. (2010). *Principles of Information Systems: a managerial Approach, 9th Edition*. 10.

Sullivan, L. (2012). Management Information System (MIS). *The SAGE Glossary of the Social and Behavioral Sciences*. https://doi.org/10.4135/9781412972024.n1516

Untuk, D., Salah, M., & Syarat, S. (2017). *E-COMMERCE PADA VARATA STORE Oleh : Albina Chrisnatasari DAN KOMPUTER (STMIK) GICI*.

Wall, L., Extraction, P., Language, R., Os, M., Scripting, P., Shell, U., Point, T., Point, T., & Point, T. (2015). *About the Tutorial Copyright & Disclaimer*. 2. https://doi.org/10.1017/CBO9781107415324.004

Whitten, B. &. (2007). *System Analysis and Design Methods* (J. L. Whitten (ed.)). Whitten, Jeffrey L. www.mhhe.com

APPENDIX

INTERVIEW

Interviewees : Ousman Keita Department : Principal Date : 25th March 2020	
Interviewer Questions	Interviewer Answers
Question 1: What type of system is been use to register the students?	The school is using paper based system.
Question 2: What are the biggest challenge students faces to access academic documents in the school?	Yes they do, specially does want to take transfer paper or trying to get their previous school document, as the school is using manual system it will took time for student's to get what they request from the school.
Question 3: Is there any automated system in the school?	There is no automated system and willing to get one in featured.
Question 4: How did the administrative collect the student information?	The school administrative use book to store student's information's.
Question 5: Where is the school information save is there any database for the school?	All information of the school is saved in a file which is paper based system.
Question 6: How is the interaction between the school and parents?	The school scheduled monthly meeting with the parents of the students to meet at the school and discuss issues that students are facing in both places.
Question 7: What is your expectation towards the feature of Latrikunda upper basic school?	To see the school in the digital world to lead the student and teachers access their materials in a single click.
Question 8: Are there any difficulties while managing the student data special in school	There are difficulties regarding to fee management process, the selection will took several time to able to get the data that they want and secondly the

fees transaction?	student's will not provide the later to their parent on time.
Question 9: Which system will you be prefer using as far as you are the principal of the school?	Well, as the generation moving forward and everything is digitalize which makes data management easier to everyone, I prefer to use automated system to reduce workload.