# Overview

*The System Design Document describes the system requirements, operating environment, system and subsystem architecture, files and database design, input formats, output layouts, human machine interfaces, detailed design, processing logic, and external interfaces.*

TABLE OF CONTENTS

[Overview ii](#_Toc522098467)

[Table of Contents **Error! Bookmark not defined.**](#_Toc522098468)

[INTRODUCTION 4](#_Toc522098469)

[Purpose and scope 4](#_Toc522098470)

[EXECUTIVE SUMMARY 4](#_Toc522098471)

[1.2.1 System overview 4](#_Toc522098472)

[DESIGN CONSTRAINTS 2](#_Toc522098473)

[1.2.2 Future contingences 3](#_Toc522098474)

[1.3 Document organization 3](#_Toc522098475)

[1.4 Project references 4](#_Toc522098476)

[2.0 SYSTEM ARCHITECTURE 4](#_Toc522098477)

[2.1 System hardware architecture 4](#_Toc522098478)

[2.2 System software architecture 4](#_Toc522098479)

[2.3 INTERNAL COMMUNICATION ARCHITECTURE 5](#_Toc522098480)

[3.0 FILE AND DATABASE DESIGN 6](#_Toc522098481)

[3.1 Introduction 6](#_Toc522098482)

[Class Diagram for Admin, Lecturer and Student 6](#_Toc522098483)

[3.2 Database management files 8](#_Toc522098484)

[3.3 Non-database management files 10](#_Toc522098485)

[4.0 HUMAN-MACHINE INTERFACE 11](#_Toc522098486)

[4.1 Introduction 11](#_Toc522098487)

[4.2 Inputs **Error! Bookmark not defined.**](#_Toc522098488)

[4.3 Outputs 11](#_Toc522098489)

[5.0 DETAILED DESIGN 11](#_Toc522098490)

[5.1 Hardware detailed design 11](#_Toc522098491)

[5.2 Software detailed design 12](#_Toc522098492)

[System Users Flow Diagram: 12](#_Toc522098493)

[Sequence Diagram: 14](#_Toc522098494)

[USE CASES: 15](#_Toc522098495)

[5.3 Internal communication detailed design 18](#_Toc522098496)

[6.0 EXTERNAL INTERFACES 19](#_Toc522098497)

[6.1 Interface architecture 19](#_Toc522098498)

[6.2 Interface detailed design 19](#_Toc522098499)

[7.0 SYSTEM INTEGRITY CONTROLS 19](#_Toc522098500)

# 

# INTRODUCTION

This is a Software Design Document has been developed for the Student Attendance System using Fingerprint Recognition. It is in line with the system requirements specification that satisfies all the system requirements, objectives and expectations of the system. It provides documentation which will be used to aid in software development by providing the details for how the software should be built.

Within the Software Design Document are narrative and graphical documentation of the software design for the Student Attendance System. System project including use case models, sequence diagrams, collaboration models, object behavior models and other supporting requirement information.

## Purpose and scope

The purpose of the Software Design Document is to provide a description of the design of a system fully enough to allow for software development to proceed with an understanding of what is to be built and how it is expected to build. The Software Design Document provides information necessary to provide description of the details for the software and system to be built.

The scope of the document is to provide a summary of the models and versions of hardware and software that the system is using. This document targets the project coordinator, supervisor and the developer as the main audience.

# EXECUTIVE SUMMARY

The Student Attendance System using Fingerprint Recognition is system that will help ease the attendance sheets system in the university’s attendance process. It is a desktop-based system that is designed to store and analyze information in the university. Student Attendance System will build an effective management system that is very convenient to use in universities since it can be accessed from anywhere. The system will serve as a more reliable and effective means of undertaking student attendance, remove all forms of delay and stress as well as enable a lecturer to know his or her student’s attendance.

This application’s design is based on the desktop platform though attempts will be later made to make it a Web Application Framework(WAF). This kind of framework enables application developers to design an application that automates the overhead associated activities performed in web development and makes the access more essential.

### 1.2.1 System overview

Student Attendance System is meant to facilitate the process of taking the attendance of students and maintaining their attendance records. It also updates the details of each student and also the details of the main users of the system(lecturers).

The system has two domain actors.

The administrator, this is the actor with the paramount responsibility of registering students and updating their details. The administrator is also responsible for the role of registering lecturers. Other role of the administrator will be to add or delete the lecturers or students. This is also the person with the role of keeping the system up and running. The system administrator will be responsible for ensuring that the database is consistent.

The lecturer, the actor who will take attendance for the respective students in the course. Report for the respective students will be generated by the lecturer or the system.

The third actors will be the student who will check on the attendance details and reports to know his/her place in the respective attendance system.

**Hierarchy Chart:**

STUDENT ATTENDANCE SYSTEM



Administration



Lecturers



Students



Manage



Add



Delete



Save



New



Update



Reports



Attendance

details



Student Details



Lecturers

Details



Profile



Profile



New



Update



View



Add

##### Figure 1: Hierarchical chart

## DESIGN CONSTRAINTS

The system is not a subject to a major constraint. Several constraints are beyond the scope of the Student Attendance System and should be carefully factored into the system design. To date, the following constraints have been identified:

* All the named users of the system, i.e. System administrator, the lecturers or the students will adopt the system.
* The system will be implemented in an environment that will allow efficient and effective access by the users using their computers(Laptops).
* The software should adhere to locally or nationally recognized standards and should follow the National Information Exchange Naming and Design Rules.
* The system will be able to merge with the existing system and will access certain details about the students and lecturers.
* The hardware will be simulated using a simulation software due to lack of a real hardware to operate the system at the time of system production.

# 1.2.2 Future contingences

Except for changes in the number of the current functionalities, there is no major future contingence that is anticipated from the system. The system hardware however shall change to incorporate the newly suggested smart cards by Egerton University Vice Chancellor Professor Munya. In case there is a change in the number of the functionalities currently supported by the system, a workaround would be to incorporate these changes to the system without affecting the current already designed functionalities. Necessary information will be gathered on the preferred interface by the users and make necessary changes to the system.

# 1.3 Document organization

While the software requirement specification (SRS) document is written for a more general audience, this document is intended for individuals directly involved in the development of Student Attendance System application. This includes software developer, project supervisor, and project coordinator. This document need not be read sequentially; users are encouraged to jump to any section they find relevant. Below is a brief overview of each part of the document.

* Part 1 (Introduction) o This section offers a summary of the Student Attendance System application, including goals and objectives, project scope, general system details, and some major constraints associated with the intended platform.
* Part 2 (Architectural and Component-Level Design) o This section describes the Student Attendance System application class by class, including interface details, class hierarchies, performance/design constraints, process details, and algorithmic models.
* Part 3 (Data Design) o Readers interested in how Student Attendance System organizes and handles data should consult this section, which covers data structures and flow patterns utilized by the system.
* Part 4 (User Interface Design) o This section covers all of the details related to the structure of the graphical user interface (GUI), including some preliminary mockups of the Student Attendance System application. Readers can view this section for a tentative glimpse of what the final product will look like.
* Part 5 (Restrictions, Limitations, and Constraints) o This section discusses the general constraints imposed upon the project
* Part 6 (Testing Issues) o Readers interested in the software testing process should consult this section, which offers a list of test cases, expected responses, and other pertinent information.
* Part 7 (Appendices)

## 1.4 Project references

This document references the Student Attendance System proposal document and the Student Attendance System requirement specification document.

# 2.0 SYSTEM ARCHITECTURE

Student Attendance System using Fingerprint Recognition is presentation that is composed of different modules that will be accessed as an application. This application is highly portable and can run of different operating systems.

## 2.1 System hardware architecture

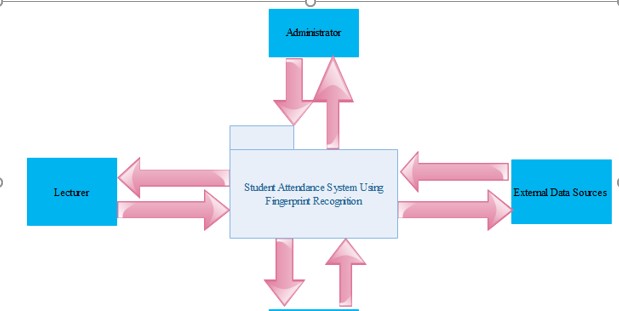
The system is based on two tier architecture with the following components; the database and the application itself. The following hardware components are used in the system:

1. A computer with the following requirements with at least 4 GB of RAM.
2. A fingerprint scanner recommendable Secugen Hamister (Simulation Software to be used instead).
3. Database Server with intel G5 Quad Core, 4GB RAM, Dual 250 GB SATA RAID 1 and Windows Server 2016.

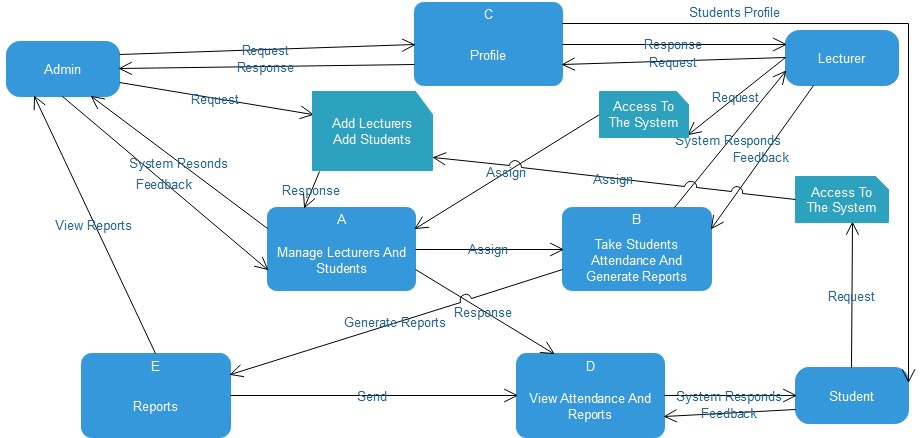
## 2.2 System software architecture

The following software components will be used:

* NetBeans - This is the IDE in which the application will be built.
* Scanner drivers and libraries.
* XAMPP server on which the database server will be hosted.
* MSQL DBMS for data storage and access.
* Below is a context diagram that only shows the top-level communication between the system and external entities. It demonstrates how the Student Attendance System interacts with some of the external sources of data required for the operation of the system.



* **Fig. 2:** *Top level communication between the system and external entities.*
* The Student Attendance System contains four processes, three external entities and two data stores as shown in the figure below.

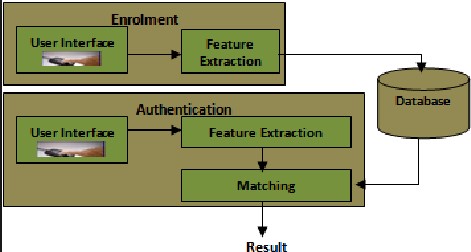


* **Fig. 3:** *A detailed communication between the system and external entities.*

## 2.3 INTERNAL COMMUNICATION ARCHITECTURE

The communication process on the application will be like this:

* There must be database connection to enable users to retrieve data and also to enable the various modules in the system to communicate with one another.
* Connection to the fingerprint reader externally shall be maintained during the data collection period to ensure the activity runs smoothly.



* **Fig. 5:** *Internal**communication between the system, database and the scanner.*

# 3.0 FILE AND DATABASE DESIGN

## 3.1 Introduction

The design of the Student Attendance System using Fingerprint Recognition has made a centralized database to ease the access of information by the various system users. Future references of the data in the system will raise a necessity to keep the data in a persistent format.

This will make sure that information will be available even if the user action has been completed. The data will also persist even when the system is changed or it is updated.

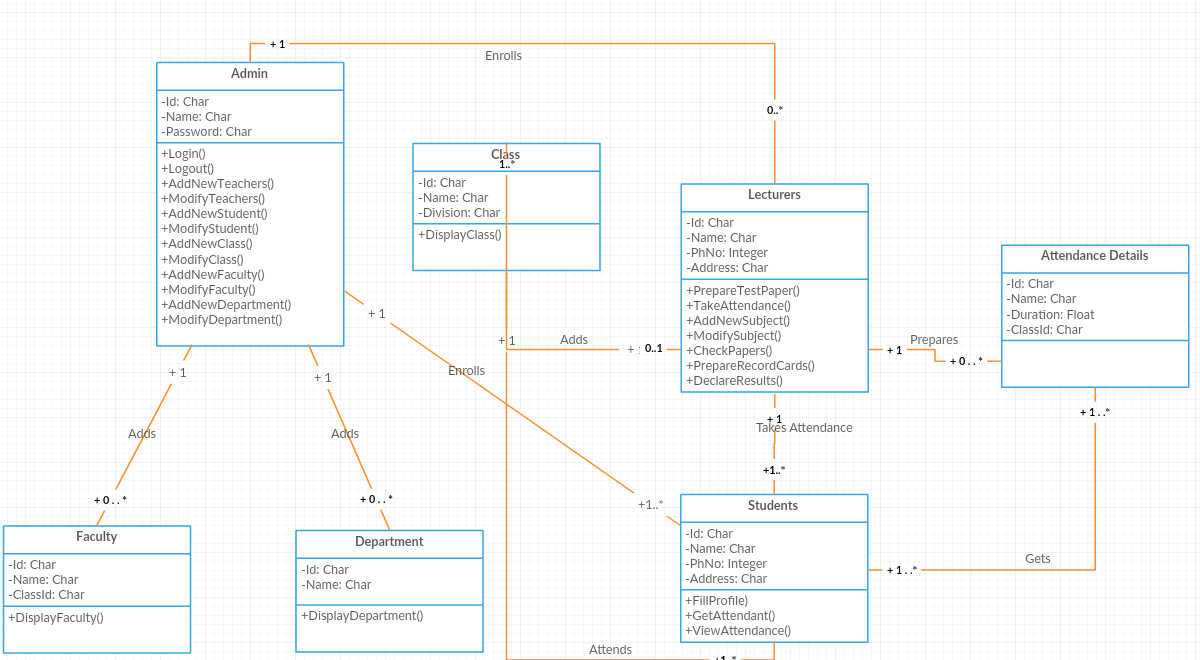
## Class Diagram for Admin, Lecturer and Student

1. Class



1. Interface





*Figure5. Class diagram for student attendance system*

#### ER- Diagrams:

The entity Relationship Diagram (ERD) depicts the relationship between the data objects. The ERD is the notation that is used to conduct the date modeling activity the attributes of each data object noted is the ERD can be described resign a data object description.  The set of primary components that are identified by the ERD are

* Data object  Relationships
* Attributes  Various types of indicators.

The primary purpose of the ERD is to represent data objects and their relationships**.** Rectangle symbol is used to indicate the entities

Oval symbol is used to indicate the attributes of an entity

Diamond symbol is used to indicate the relationship between the entities

Arrow symbol specifies the direction

Login for the admin and lecturer:

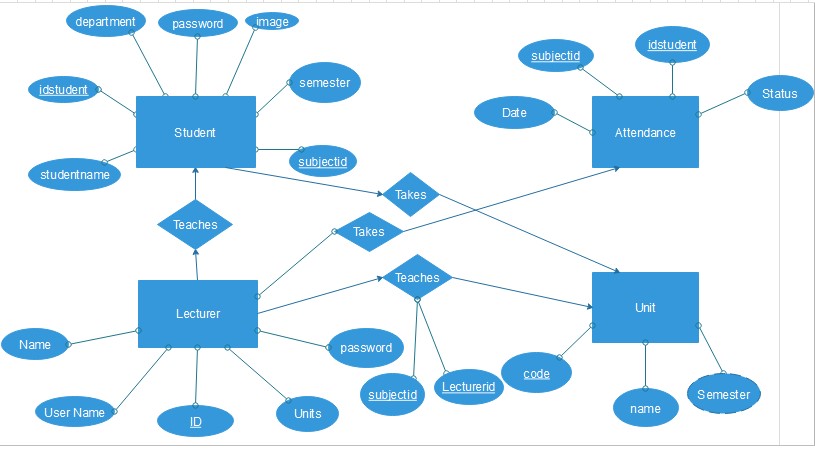
login

username

user type

password

ER Diagram for Students Attendance System Using Fingerprint Recognition



*Figure6. ER diagram for student attendance system*

## 3.2 Database management files

MySQL is used as the primary storage for all the data that is in the application. The reason why the database is used because it has the following qualities:

* It supports Structured Query Language(SQL)
* It is open and free.
* It supports referential integrity and transaction processing

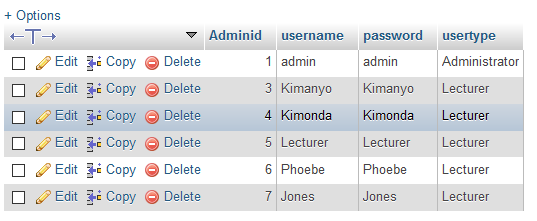
The database also comes with associated rich set of DBMS applications that facilitate query processing, analysis of data and making reports.

To facilitate the integrity, confidentiality and security of the stored data, the database will only be accessible by the system administrator. There will be user interfacing activities on the various modules of the system to further facilitate data access and report generation.

The following entities will form the database schema (These are screenshots from XAMPP MySQL showing the database schema).

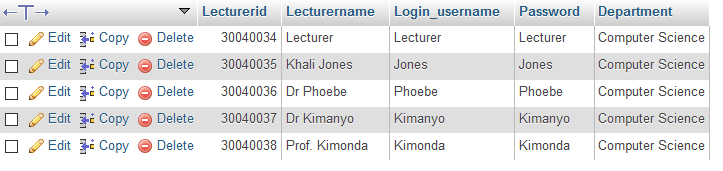
##### Login relation

Handles relation for the admin and lecturer login authorizations.



##### Lecturer relation

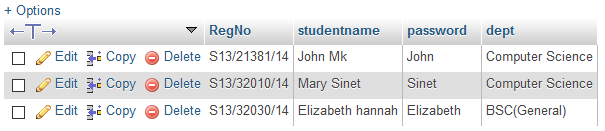
This is a relation that keeps the details of lecturer activities



##### Student relation

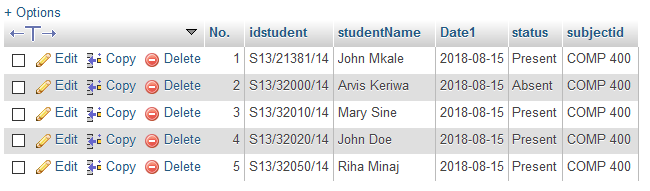
It is user to keep the details of the students in a relation

* Login



##### Attendance relation

Keeps record about the attendance details of the students.



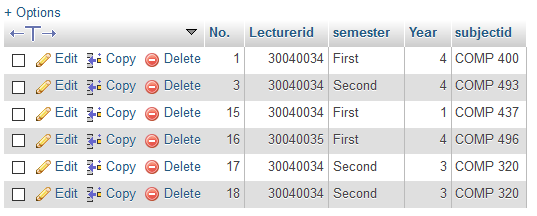
##### Subjects relation

Keeps record about the subjects for a particular semester.



##### Teaches relation

Shows the relationship between subjects and lecturers.



## 3.3 Non-database management files

All data will be stored in relational database to facilitate the process of central access of data by all the system users. The management of data shall be done strictly using a database.

Despite all this, reports generated from the database will be stored in the computer hard drive as excel worksheets. They shall be obtained from the computer hard drive.

# 4.0 HUMAN-MACHINE INTERFACE

## 4.1 Introduction

The system will consist of login module where the administrator or the lecturers will input their username and password. The users will select their domain. After that, the users will then log in to the system and access it using the details already stored in the database.

Users will put input their login credentials and they shall be authenticated. After authentication, the users shall be able to perform their respective functions as stipulated in the Software Requirements Specification. Each user, upon successful login will be directed to their respective modules that will enable them to perform their respective functions.

The students will have a separate login page. This separation is due to security factors associated with complexity of the system.

## Various modules will be implemented in the manual

## 4.3 Outputs

The system will output a range of data based on the user needs. Reports in the system will be generated as plain text, pdf or in an excel worksheet format. Other data will be outputted as either read-only or editable based on the user roles and needs.

# 5.0 DETAILED DESIGN

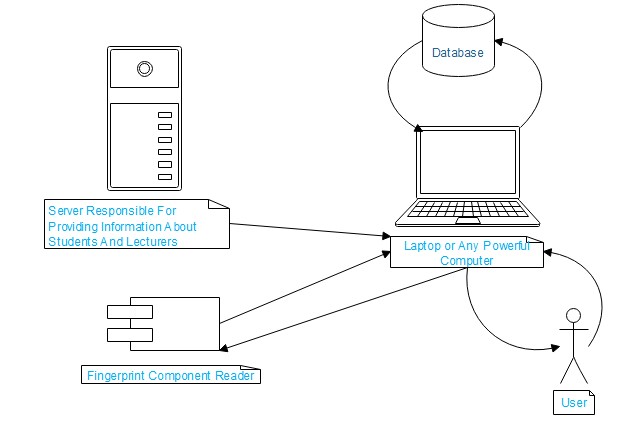
## 5.1 Hardware detailed design

The system will run on a number of hardware components.

* Client PC. With at least 1GB RAM and 2.5 GHZ processor speed.
* Fingerprint scanner.
* USB cable connecter.
* Any other supporting device.

The physical component connection is depicted in the diagram shown below.

**Diagram Showing All Hardware Components**



*Figure7. All hardware components for the student attendance system*

## 5.2 Software detailed design

The system consists of the following modules. Each module is shown by a use case described below having the specific functionalities. The criterion of grouping these modules is based on the user roles. There are two main actors in the system. There are the system administrator and the lecturers. Other actors include the students.

##### Flow Diagrams:

# System Users Flow Diagram:

The set of components are defined as:

1. Process



1. Start/End



1. Decision



1. Data

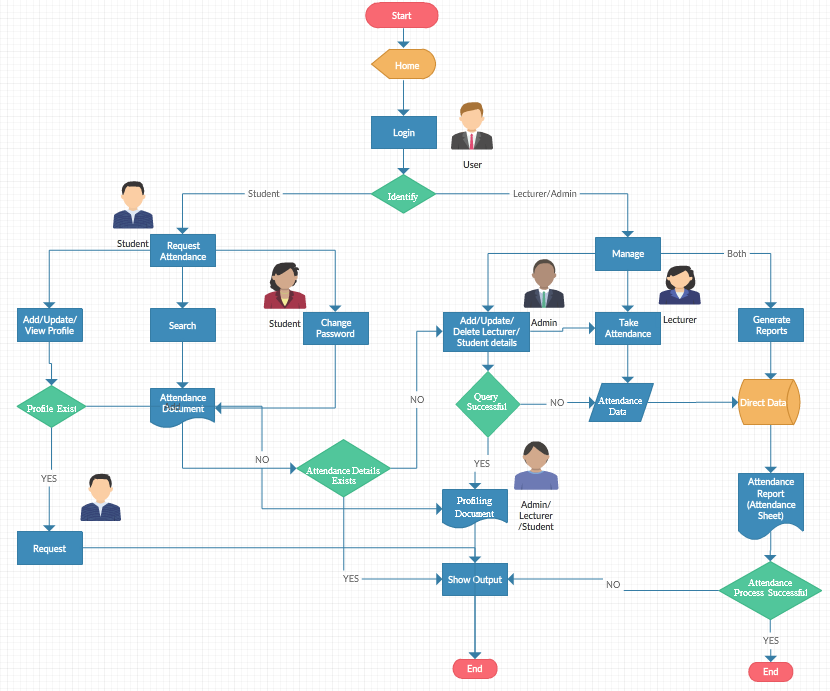


1. Document



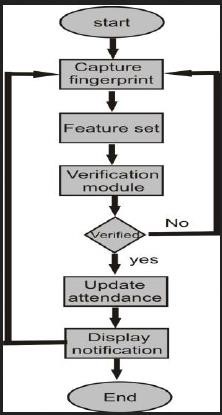
1. Display





*Figure8.Flow diagram for student attendance system*

**Flow Diagram for the Scanner Module System:**



*Figure8. Fingerprint flow diagram for student attendance system*

# Sequence Diagram:

The following components are used:

1. Life Line



1. Activation

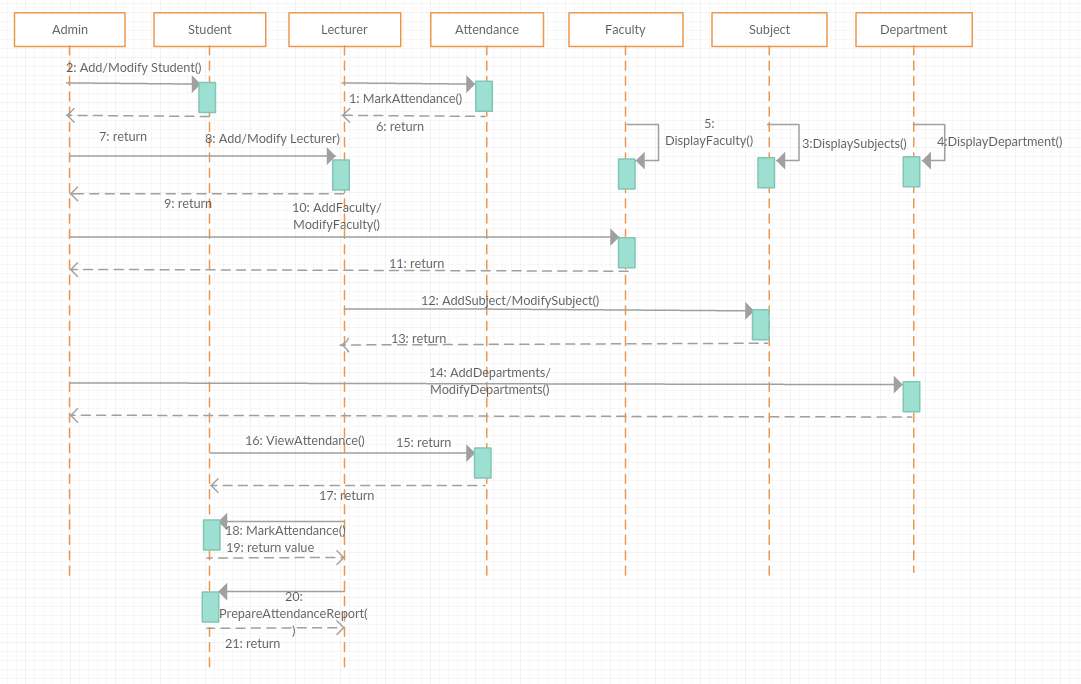


1. Progression Forward Accepted



1. Progression Backward Accepted





*Figure9. Class diagram for student attendance system*

# USE CASES:

* **General Use Case and Relationship**
* **Key**

1. Main Actor



1. Minor Actors

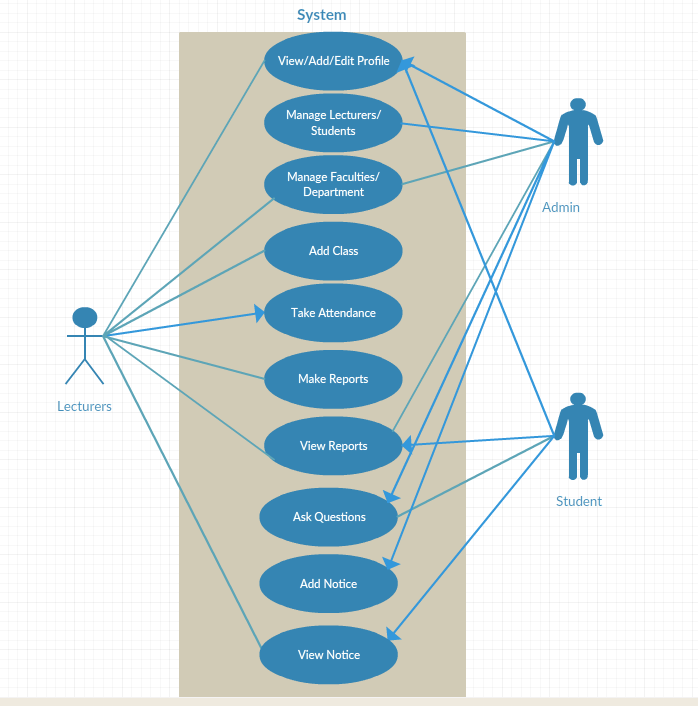


1. System



1. Use case





*Figure10. Fingerprint flow diagram for student attendance system*

1. **System Administrator Use Case**



**View Profile**



**Maintain System**



**Access Reports**



**View Reports**



**Manage Students**



**Manage Lecturers**



**Administrator**

1. **System Lecturer Use Case**



**View Profile**



**Advice Students**



**View**

**Reports**



**Make Reports**



**Take Attendance**



**Add Class**



**Lecturer**

1. **System Student Use Case**



**View Profile**



**Manage Profile**



**View Notice**



**View Reports**



**Ask Questions**



**Request Attendance**



**Student**

**The database:** it is used to store data

## 5.3 Internal communication detailed design

The flow of information from one module to another in the attendance system will make internal flow of information necessary. The internal flow of information will be in form of object communication among various objects of the classes in the system.

The internal data communication will be handled in an agreeable protocol that will relay messages between various objects in the system.

Communication between the system and the database will be facilitated by various user interfaces describes earlier in the human-machine interface section which will ensure that data from the application is successfully stored in the database.

The interface will also ensure that data can easily be fetched from the database, updates and deletions can be done without affecting the consistency of data.

External data communication between the system and the external environment will be realized by use of the physical connection between the system and the external environment through which data will be sent.

# 6.0 EXTERNAL INTERFACES

## 6.1 Interface architecture

External interfacing will facilitate the communication between the application and the external systems. Communication will be through external cabling. A physical USB cable will connect the system to the fingerprint reader.

Specific architectures will be implemented for the application and the database PCs. An active link of communication will be maintained between these components and any other external systems that the application will be exchanging information with.

## 6.2 Interface detailed design

Since communication is through cabling, there is no need to reformat data before it is transmitted or received over the interface between two different communicating modules.

Data exchange over the interface will be purely regulated by the special rules. Errors arising from such exchange will be handled dynamically.

Errors arising will be handled and a brief error description given to the user in exception format.

# 7.0 SYSTEM INTEGRITY CONTROLS

Based on the sensitivity and importance of the data that is passed between the various modules of the system, there is need to put in place integrity checks to ensure correctness of the stored data and accountability on the part of every system user. The following security control measures will be implemented to enforce integrity controls

1. Authorization control. This ensures that users can only access data that belongs to their user groups based on user roles.
2. Access to the data in the database will be restricted to various user groups based on the user roles. Users can only update data that they have supplied. Update of data affecting the whole system will only be restricted to the system administrator. Data deletions will also be limited to certain classes of users based on the user roles.
3. Integrity control. This ensures that data stored in the database cannot be violated or altered by any unauthorized parties in any way whatsoever.
4. Authenticity control. This aspect of security will verify user credentials at the time of login into the system. This ensures that only authorized users are allowed to access the system.
5. Confidentiality control. This is a security technique that prohibits unauthorized access of personal information.
6. There shall be a system audit mechanism that automatically collects data based on user transactions to monitor user activities in the system. The audit will be available to the system administrator in form of log information detailing user activities, user identifications and time so as to enhance accountability.