**EGERTON UNIVERSITY**



**FACULTY OF SCIENCE**

**COMPUTER SCIENCE DEPARTMENT**

**REQUIREMENTS SPECIFICATIONS (SRS)**

**PRESENTED BY:**

**GROUP 1**

**MEMBERS REGISTRATION NUMBER:**

S13/21826/14

SP13/20031/14

S13/21374/14

S13/21379/14

COMP 390 TEAMWORK PROJECT

ASSET MANAGEMENT SYSTEM

**COMPLETION DATE:**

**PRESENTED TO:**

**SUPERVISOR: MR ODIYO**

FOR THE PARTIAL FULFILMENT OF DEGREE IN COMPUTER SCIENCE OF EGERTON UNIVERSITY

Abstract

Over the years there has been misunderstanding between the laboratory technicians and the users. The misunderstanding was a result of some items going missing among the borrowers and the renders tends to refuse the fact that they are responsible for it. Our system will take care of this. It will contain the records of the present equipment and the borrowed. In case an equipment is borrowed the technician updates the database and writes down who has borrowed the equipment. The system will also guide the technician to trace who lost a certain equipment and records that they should pay for it.

Acknowledgement

In our research we acknowledge the Chairman of Physics Department, Dr. Rop for moral support and guidance in feasibility study about the equipment in the physics laboratory in Egerton University.

Conversely, we acknowledge Git Hub for the guidance in coming up with different functions of different classes when coding.

Finally, we acknowledge our lecturer Mr. Odiyo for accepting our proposal for this project. Thanks.

Approval

This is our own original group work not copied anywhere but out of our own brainstorming and coming up with an asset management system.

SCRUM MASTER LECTURER

SIGNATURE\_\_\_\_\_\_\_\_\_\_\_\_ SIGNATURE\_\_\_\_

NAME\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ NAME:\_\_\_\_\_\_\_\_

DATE\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ DATE: \_\_\_\_\_\_\_\_\_

Table of Content

1. Introduction
   1. purpose
   2. Scope
   3. Definitions
   4. Reference
   5. Overview
   6. Product Perspective
   7. Product Function
   8. Constraints
   9. Assumption and Dependence

2.0 Specific Requirements

2.1 Functionality

2.1.1 Add Equipment

2.1.2 Book Equipment

2.1.3 Delete Equipment

2.1.4 Update Equipment

2.2 Reliability

2.2.1 Maintanance

2.2.2 Security Consideration

2.3 Performance Requirement

2.3.1 Response Time

2.3.2 Capacity

2.4 Supportability

2.4.1 Naming Convention

2.4.2 Coding Standard

2.5 Design Constraints

2.5.1 Parent Component

2.5.2 Software Language

2.6 Online User Documentation

2.7 Interfaces

2.7.1 User interface

2.7.2 Hardware interface

2.7.3 Software interface

2.7.4 Communication interface

Logical Database Requirement

3.0 Analysis Model

3.1 Sequence Diagram

3.2 Data Flow Diagram

4.0 Change Management Process

5.0 Support information

5.1.1 Appendix-ER diagram

5.1.2 Appendix-Data Flow Diagram

5.1.3 Appendix-Data Dictionary

Software Requirements Specification

# Introduction

## Purpose

The system is developed so that to main the number of the asset in the laboratory and to be allocated to different lecturers for the students to use it during experimentation. The lab technician enters the username and enters the password which are then matched with the database. If the details match with those that are stored in the database access is denied. The lab technician is the one responsible for any change of equipment quantity.

## Scope

The asset management system will deal with Lecturers login and registration, collection of the quantity of the equipment daily, deletion of the equipment once lost and booking of the equipment by the lecturers and finally deal with the grouping of the students in different distinct groups. Moreover, the system will include adding of the equipment into the system in case of a new equipment has been purchased of been replaced with the lost one.

## Definitions, Acronyms and Abbreviations

Scrum master- The group coordinator and the objectives delegator.

DFD-Data Flow Diagram

OOD-Object Oriented Design

## References

Dennis book for software engineering

## Overview

The rest of the SRS examines the specifications of system including in section 3 it deals with specific requirements. The specific requirements include the functionality of the system such as adding an equipment and the flow of events in adding this functionality. Also includes its reliability, maintenance and security constraints. Performance requirements such as response time and the capacity is also inclusive. The specification of the system in this sections also deals with interfaces which are user interface, hardware interface and software interface. Section 4 deals with the various models for analysis which are DFD and OOD. Section 5 deals with Change Management Process which involves the process of incorporating the change into the institution by implementing the system. Section 6 deals with Supporting information such as Data dictionary.

## Product Perspective

The system will not be web based and will run without any internet connection. The system will only use the database provided, to respond to the request of the users.

## Product Functions

* The system will allow the user, laboratory technician to create a computerized database that will store information about equipment.
* The information entails names of the equipment, booked equipment, lost equipment and the cost of the equipment.
* The system will also give a list of the equipment under maintenance and the list of the equipment which are booked. If one tries to book an equipment which is not available, the system will inform the lecturer trying to book the equipment that it is unavailable.
* The system will also store the dates when an equipment was borrowed.

## Constraints

* The system will work best with Pentium processor.
* A computer with installed memory of 4GB with 64 bit Operating System computer.
* The computer to implement this system must have at least speed of 2.6GHz.

## Assumptions and dependencies

* The developer of the system assumes that the end users are conversant with the network Operation System
* The developer also assumes that the lab technician is conversant with database in MySQL
* The developer can effectively manipulate the database so as to effectively update the database.

# Specific Requirements

## Functionality

### Add Equipment

* The technician logs into system
* Goes to database
* Enters the name of the new equipment as per the data type declared for the name column
* Enters the quantity, in the quantity column which takes an integer
* Date when it was bought is entered and its status whether booked or borrowed is written down.

### Book Equipment

* The lecture logs onto the system
* Request for a display of the available equipment
* The lecturer then clicks the book button for the equipment of interest

### Delete Equipment

* The lab technician logs into the system
* In the list of the equipment, he clicks the delete button to delete the equipment of interest

### Update

* The lab technician logs into the system
* Clicks on the update button to refresh the changes made by the technician

## Reliability

### Maintenance

The system will have the system technician who will be checking on its operation and check whether the response time is constant and there are no delays in response.

Keep on updating the MySQL version so as to keep up to date.

### Security Considerations

The database will only be authorized for the laboratory technician to alter it.

The lab technician is responsible for all the operations on the database such as deletion of the equipment, adding equipment and updating the equipment information such as the date they were borrowed or returned.

## Performance Requirements

### Response time

The maximum amount of time for any operation using this system is 2 seconds.

### Capacity

The database may store up to 100 records about the equipment.

## Supportability

### Naming Convention

All code will written as specified by the policy

### Coding Standards

All code will be written as required by the GNU General Purpose License.

## Design Constraints

### Parent Component:

The system will be part of ERP of the component

### Software Language

All coding will be done in pure language.

## Online User Documentation and Help System Requirements

All documentation will be made in accordance with requirements pertaining to open source software under the GNU General Purpose License. Additionally, on-line user documentation will be in the form of **man** pages accessible through Linux.

## Interfaces

### User Interface-

* The system will have one user interface where each credited user of the system enters his user name and the password which will then be verified with the database if it matches.
* If the username and the password matches, then the username is granted access to the system and can perform various functions assigned to.

.

### Hardware Interface

This one includes

* CPU usage
* Memory usage is about 1GB
* Swap file creation when there are new updates in the system
* Network communication once a request is made about the asset

### Software Interfaces

The system will be interfacing with the operating system

The system will also be interfacing with the MySQL so as to view the database of the asset available in the laboratory

### Communications Interfaces

In this system phone call will be used in case the borrower fails to return the asset in due time. The lab technician does this.

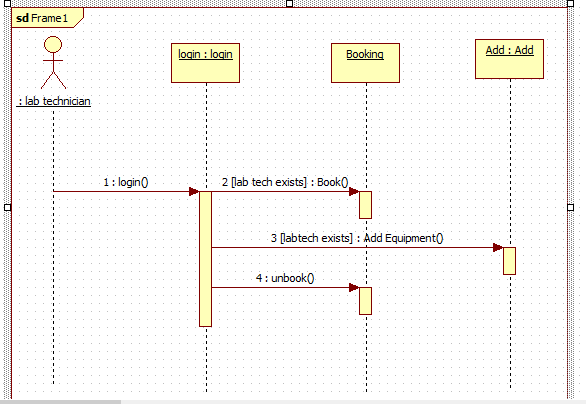
## Logical Database Requirements

In this system database is very crucial. The database includes the records of the assets in the laboratory. So as to enhance the integrity of the database, the lab technician is the only person who has the rights to create, read, update and delete database in the system.

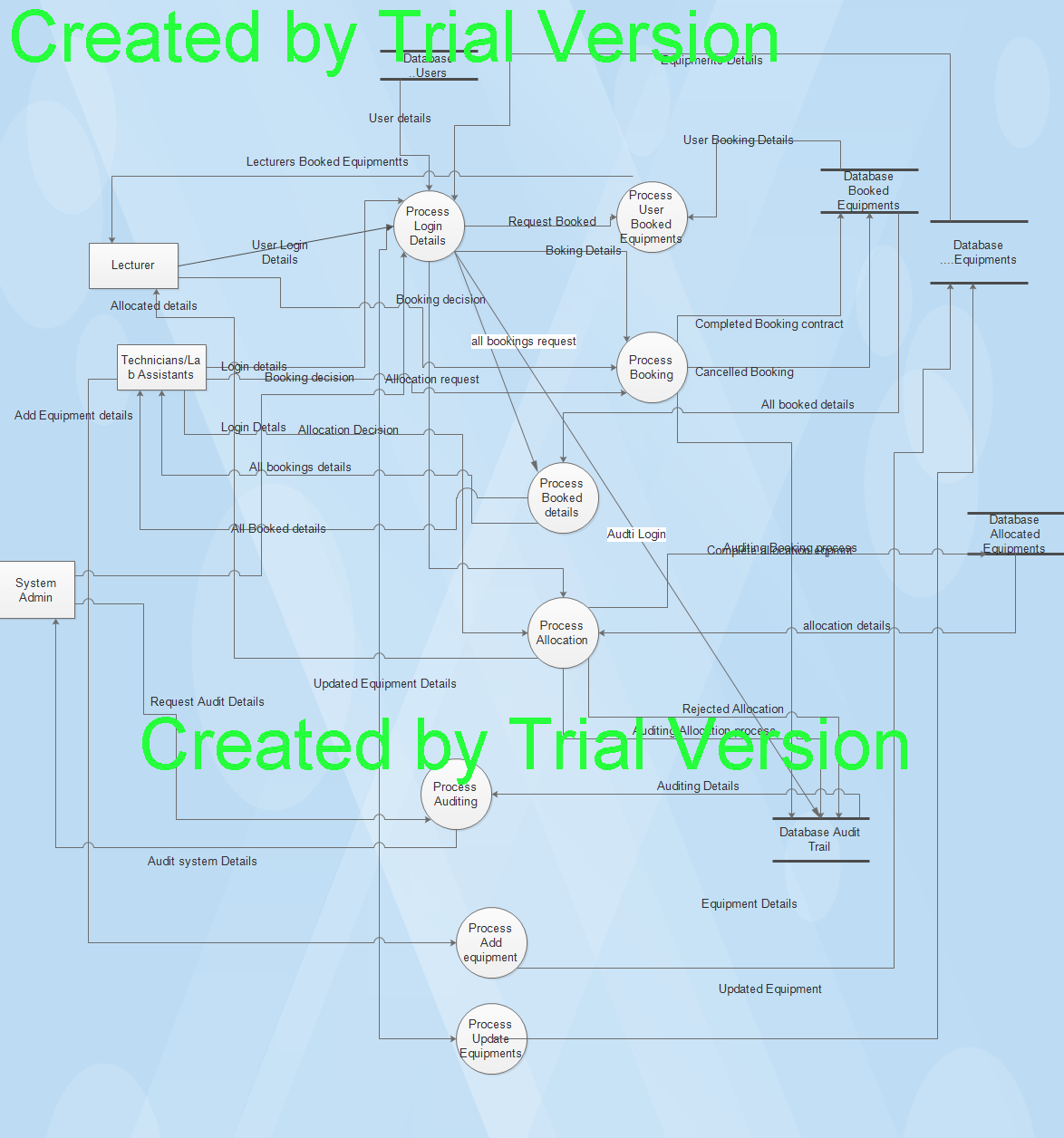
The records of the borrowers database is also used in this system to be on toes who has what

# Analysis Models

## Sequence Diagrams (if used as in O.O design)



## Data Flow Diagrams (DFD) - (structured design)



# Change Management Process

In case a change has occurred in the requirements of the system, it is upon the scrum master to update the SRS.

Notify the System administrator about the change and then proceed to the relevant authorities for the implementation of the change to occur.

The authorities will then prove the changes.

# Supporting Information

### Appendix e.g – Data Flow Diagram

