

**BONGA UNIVERSITY**

**COLLEGE OF ENGINEERING AND TECHNOLOGY**

**DEPARTMENT OF COMPUTER SCIENCE**

**WEB BASED STUDENT CLINIC MANAGEMENT SYSTEM FOR BONGA UNIVERSITY**

The Project submitted to department of computer science, college of engineering and technology, Bonga University, in meeting the preliminary project requirement for partial fulfillment of the award of Bachelor of Science Degree in computer science.

Advisor Name: Memar Zemene

Bonga, Ethiopia

Date/Year: March 4/2023



**BONGA UNIVERSITY**

**COLLEGE OF ENGINEERING AND TECHNOLOGY**

**DEPARTMENT OF COMPUTER SCIENCE**

**WEB BASED STUDENT CLINIC MANAGEMENT SYSTEM FOR BONGA UNIVERSITY**

By

Group member ID

1. Abreham Hailu ------------------------------------------- Aku1106229
2. Adimasu Bashu ------------------------------------------- Aku1106160
3. Beyena Tira ------------------------------------------- Aku1106295
4. Melaku Desie ------------------------------------------- Aku1106329
5. Yalew Fetene ------------------------------------------- Aku1106319

The project submitted to department of computer science, college of engineering and technology, Bonga University, in meeting the preliminary project requirement for partial fulfillment of the award of Bachelor of Science Degree in computer science.

Advisor Name: Memar Zemene

Bonga, Ethiopia

Date/Year: March 4/2023

# Approval sheet

This Project entitled “web based student clinic management system for Bonga University” has been read and approved as meeting the preliminary project requirements of the Department of computer science in partial fulfillment for the award of Bachelor of Science degree in computer science, Bonga University, Bonga, Ethiopia.

**Approved by:**

1. Name of Advisor: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Signature: \_\_\_\_\_\_\_\_\_\_\_Date:\_\_\_\_\_\_\_\_

2. Name of Project Coordinator: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Signature: \_\_\_\_\_\_\_\_\_\_\_Date:\_\_\_\_\_\_\_\_

# Acknowledgement

First of all, we would like to thanks almighty God, for all giving us strength, peace and well health throughout the journeys of this project.

Next, we would like to thank our advisor Memar Zemene. For all admiring advice, support and Passionate guidance which he gave us throughout the project phases. He was very voluntary to guide and assist us from the beginning of the project up to now without being bored and we would like to thank Mr. Kifile Gchilo. The manager of student’s clinic, who is voluntary for interview and shows us some clinic documents.

Finally, thanks to our friends whose support and encouragement goes even beyond the accomplishment of this project.

**Table of Contents**

[Approval sheet i](#_Toc15587)

[Acknowledgement ii](#_Toc19900)

[List of figure v](#_Toc11763)

[List of tables vii](#_Toc27131)

[Acronyms and Abbreviation viii](#_Toc8455)

[Abstract ix](#_Toc30440)

[CHAPTER ONE 1](#_Toc3819)

[1. INTRODUCTION 1](#_Toc6620)

[1.1. Background 1](#_Toc24650)

[1.2. Statements of the Problem 1](#_Toc30382)

[1.3. Objectives 2](#_Toc22570)

[1.3.1. General Objectives 2](#_Toc24745)

[1.3.2. Specific Objectives 2](#_Toc12165)

[1.4. Scope of the project 3](#_Toc31827)

[1.5. System Development Methodology 3](#_Toc21107)

[1.5.1. System development process model 3](#_Toc8298)

[1.5.2. System development approach 3](#_Toc29620)

[1.6. Significant of the project 3](#_Toc30650)

[1.7. Beneficiaries of the project 4](#_Toc4522)

[1.8. Feasibility study (Analysis) 4](#_Toc12990)

[1.8.1. Economic feasibility 4](#_Toc8194)

[1.8.2. Technical Feasibility 4](#_Toc20266)

[1.8.3. Operational feasibility 4](#_Toc25484)

[1.8.4. Legal feasibility 5](#_Toc27846)

[1.8.5. Time feasibility 5](#_Toc14886)

[1.9. Project schedule 5](#_Toc8784)

[1.10. Project budget 6](#_Toc5790)

[CHAPTER TWO 7](#_Toc6718)

[2. REQUIREMENT ANALYSIS 7](#_Toc30962)

[2.1. Current system description 7](#_Toc16008)

[2.1.1. Major function of current system 7](#_Toc11619)

[2.1.2. Problem of existing system 7](#_Toc13293)

[2.2. Requirement Gathering 7](#_Toc12585)

[2.2.1. Data gathering methods 7](#_Toc10328)

[2.2.2. Business Rule 8](#_Toc20946)

[2.3. Proposed system description 8](#_Toc21824)

[2.3.1. Overview 8](#_Toc1170)

[2.3.2. Functional requirement 8](#_Toc24125)

[2.3.3. Nonfunctional requirement 9](#_Toc7431)

[CHAPTER THREE 11](#_Toc13618)

[3. SYSTEM MODEL 11](#_Toc1866)

[3.1. Scenario 11](#_Toc14507)

[3.1.1. Use case model 11](#_Toc1849)

[3.1.2. Use Case Diagram 12](#_Toc9838)

[3.1.3. Description of Use Case Model 14](#_Toc20318)

[3.1.4. Activity Diagram 23](#_Toc5100)

[3.1.5. Data Dictionary 27](#_Toc25272)

[3.1.6. Class Model 28](#_Toc8857)

[3.1.7. Dynamic Modeling 28](#_Toc27543)

[3.1.8. User Interface 32](#_Toc3688)

[CHAPTER FOUR 33](#_Toc9609)

[4. SYSTEM DESIGN 33](#_Toc13979)

[4.1. Introduction 33](#_Toc16500)

[4.2. Current software architecture 33](#_Toc27241)

[4.3. Proposed software architecture 33](#_Toc23033)

[4.3.1. System decomposition 33](#_Toc14123)

[4.3.2. Hardware/ Software mapping 33](#_Toc32607)

[4.3.3. Persistent data modeling 34](#_Toc7448)

[4.3.4. Access control and security 35](#_Toc24546)

[4.3.5. Detailed class diagram 37](#_Toc1743)

[4.3.6. Package diagram 38](#_Toc15002)

[4.3.7. Deployment 39](#_Toc570)

[CHAPTER FIVE 41](#_Toc11331)

[5. IMPLEMENTATION 41](#_Toc28057)

[5.1. Mapping Models to Code 41](#_Toc25784)

[5.2. Screen Images 55](#_Toc8616)

[5.3. Testing and validation 57](#_Toc18658)

[5.3.1. Unit testing 57](#_Toc10819)

[5.3.2. White box testing 57](#_Toc27999)

[5.3.3. Black box testing 57](#_Toc26879)

[CHAPTER SIX 58](#_Toc30518)

[6. CONCLUSION AND RECOMMENDATION 58](#_Toc6527)

[6.1. Conclusion 58](#_Toc6989)

[6.2. Recommendation 58](#_Toc11672)

[References 59](#_Toc31363)

[Appendix 60](#_Toc26504)

# List of figure

[Figure 1 : Gantt chart for Project schedule 5](#_Toc11271)

[Figure 2 : Use case diagram for WBSCMS 13](#_Toc26239)

[Figure 3 : Activity diagram for generate report 24](#_Toc13803)

[Figure 5 : activity diagram for give feedback 25](#_Toc4585)

[Figure 6 : Activity diagram for write lab result 26](#_Toc25356)

[Figure 7 : class model 28](#_Toc19109)

[Figure 8 : Sequence diagram for register patient 29](#_Toc10705)

[Figure 9 : Sequence diagram update patient info 29](#_Toc27120)

[Figure 10 : Sequence diagram give feedback 30](#_Toc23036)

[Figure 10 : Sequence diagram generate lab 31](#_Toc20302)

[Figure 11 : Sequence diagram view patient info 32](#_Toc28116)

[Figure 12 : Hardware/software mapping 34](#_Toc23246)

[Figure 13 : persistent data diagram 35](#_Toc30339)

[Figure 14 : WBSCMS detailed class diagram 38](#_Toc23695)

[Figure 15 : Package diagram 39](#_Toc10173)

[Figure 16 : Deployment diagram 40](#_Toc7555)

[Figure 20 : Register drug to store 55](#_Toc18082)

[Figure 21 : View store drug information 56](#_Toc25971)

[Figure 22 : Register employee 56](#_Toc32742)

# List of tables

[Table 1 : Cost of project 6](#_Toc32074)

[Table 2 : use case documentation for login 15](#_Toc12426)

[Table 3 : use case documentation for register patient 16](#_Toc5734)

[Table 4 : use case documentation for search patient information 17](#_Toc24327)

[Table 5 : use case documentation for view patient info 17](#_Toc19059)

[Table 6 : use case documentation for update patient information 18](#_Toc15219)

[Table 7 : use case documentation for make appointment 18](#_Toc746)

[Table 8 : use case documentation for give feedback 19](#_Toc11954)

[Table 9 : use case documentation for generate report 19](#_Toc16491)

[Table 10 : use case documentation for register employee 20](#_Toc5109)

[Table 11 : use case documentation for create account 20](#_Toc14942)

[Table 12 : use case documentation for view report 21](#_Toc30084)

[Table 13 : use case documentation for update account 21](#_Toc8662)

[Table 14 : use case documentation for view appointment 22](#_Toc1724)

[Table 15 : use case documentation for post news 22](#_Toc3002)

[Table 16 : data dictionary for register patient 27](#_Toc32595)

[Table 17 : data dictionary for register employee 27](#_Toc17958)

[Table 18 : Data dictionary for create account 27](#_Toc15332)

[Table 19 : access control and privacy 37](#_Toc7439)

# Acronyms and Abbreviation

1. BR Business rule
2. BU Bonga University
3. E.C Ethiopian Calendar
4. HTML Hyper Text Markup Language
5. JS Java Script
6. OOSD Object Oriented Software Development
7. PHP Hypertext Preprocessor
8. SQL Structured Query Language
9. UML Unified Modeling Language
10. WBSCMS Web based Student Clinic Management

# Abstract

Currently Bonga university student clinic uses fully manual system. The proposed web based student clinic management system is to solve manual system problems like reduce searching time exist patient information, patient information redundancy and drug related issues. The proposed system store information on database permanently, remove the redundancy of patient information, reduce circulation between clerk, doctor, lab technician and pharmacist and search existing patient record easily. Generally, it provides fast, efficient, flexible, reliable and secure services for better satisfaction of patient and improvement of motivation of the employee.

We used the object-oriented software development methodology to conduct this project, the language is PHP, JS, CSS and HTML and the data repository is MySQL. During the analysis and design process of project our team used iterative software model. In order to collect information about the clinic and it decide what the system should solve we use observation and interview. We hope the proposed system will bring a great change on the way currently Bonga University Student Clinic Management System is working.

# 

# CHAPTER ONE

# INTRODUCTION

## Background

Bonga University is new and rapidly developing universities in Ethiopia, which is established in September 2010 E.C. BU is giving different services to the students and other staff members. Among those services, student clinic is the one that provide health care service for student. This student clinic was established in September 2010 E.C together when the university established.

Now a day clinic has different medical departments such as card department, laboratory department, consultation department, OPD (Out Patient Department), Emergency and pharmacy department and there are 25 employees in the clinic; 7 in card department, 3 in laboratory department, 3 in OPD department, 3 in pharmacy, 1counseling, 4 Emergency, 1 Manager and 3 additional.

The clinic is giving medical services for 170 - 250 students per week. The number of students is increasing from year to year therefore the number of students getting the clinic service will increase. The clinic is using manual system for managing the overall activities of the work in the clinic. The admission or registration of patients is being done using paper and pen with manpower. The circulation of information from one department to another is performed by people with paper and card. The development of different computer-based system has provided tremendous advantages for many organizations and institutions. Many developed countries are using computer technology to accomplish their everyday activities effectively and efficiently.

Now a day many companies are using automated system to accomplish their task but some are still using manual system. Bonga university student clinic is one of such institute that uses paper-based system to perform all their tasks. The current system is inefficient for better performance of all tasks, for managing all medical related works to register and store patient’s record. So, in the clinic the patient data, writing report and anything done in clinic should be computerized to decrease the load of clerk and other clinic employee.

## Statements of the Problem

Bonga university student clinic is facing many problems with its manual (paper based) system. The number of patients is increasing from time to time. So that managing those patients by using existing system is very tedious that requires a greater number of human power, time and material. Some of the main problems of the existing system are listed as follows: -

* Difficult to manage patient information.
* Large storage medium (space) is required to store medical documents, cards, reports, and patient information.
* Searching for even single data is time consuming.
* Loss of patient record
* Redundancy (multiple record of the same data)
* Difficult in preparing an organized report
* Sudden damage on storage medium resulting in zero data
* Circulation of patient information such as (lab result generated by lab technician, doctor prescription to pharmacy) is late.
* Patient is not getting service according to their arrival and appointment

## Objectives

### **General Objectives**

The main objective of our project is to develop a web-based clinic management system for Bonga University student clinic.

### **Specific Objectives**

To achieve the general objective, we use the following specific objectives.

* To studying the background of the organization
* To identify the existing system problem
* To analyzing the existing system
* To defining system requirements
* To designing the new system
* To implementing or coding the designed system
* To testing the system
* To deploying the system
* To storing information in database

## Scope of the project

Basically, clinic management system comprises many tasks associated with health care in the clinic. But our proposed system mainly focused on clinic management in case of Bonga university student clinic. It supports only patient related activities such as: -

* Registering new patient
* Healthcare clinic registered private firm providing medical services to the Patients.
* Updating and accessing patient information and generating reports.
* Searching for the existing patient
* Drug related activity
* Store data permanently
* Generate laboratory report

## System Development Methodology

### **System development process model**

The development method we are using to develop the proposed system is the iterative model. Software Development life cycle (SDLC) is a spiritual model used in project management that defines the stages include in an information system development project, from an initial feasibility study to the maintenance of the completed application. There are different software development life cycles models specify and design, which are followed during the software development phase. Each process model follows a series of phase unique to its type to ensure success in the step of software development [2].

Iterative is one of the models that used in software development. It is a particular implementation of a software development life cycle that focuses on an initial, simplified implementation, which then progressively gains more complexity and a broader feature set until the final system is complete. In short, iterative development is a way of breaking down the software development of a large application into smaller pieces. Therefore, we use Iterative model to our project.

### System development approach

Object Oriented Development (OOD) has been touted as the next great advance in software engineering. It promises to reduce development time and resources required to maintain existing applications, increase code reuse, and provide a competitive advantage to organizations that use it.

The main aim of Object-Oriented Design (OOD) is to improve the quality and productivity of system analysis and design by making it more usable. To make our project reusable and modular as described above we OOSD to our project.

## Significant of the project

Some significant of the projects are: -

* To minimize the time in case of circulation of patient information and searching time
* To reduce cost in case of paper and reduce number of employees
* To protect loss of data by using back up method
* Store data for long period of time
* Write report easily from different health care departments
* Manager easily to manage the clinic, controls the expired date drug, more require drug and also number patient per day and employees to do work properly etc.

Generally, to make tasks simple and efficient in every aspect.

## Beneficiaries of the project

When one project is developed there is a benefit for customer, developer and also for users. Its benefits for: -

* **Student**: to get expeditiousness service
* **Clinic**: to decrease circulation time, number of employee and store patient information permanently
* **Employee**: to reduce labor and time during circulation and searching for existing patient information.
* **Developer**: to attain experience and payment

## Feasibility study (Analysis)

Feasibility study is used to investigate the proposed system in multiple dimensions. On the basis of the feasibility study decision is taken on whether to proceed or to cancel the project. The proposed system can be seen according to the following literals.

### Economic feasibility

Economic feasibility determines whether there are sufficient benefits in creating to make the cost acceptable, or is the cost of the system too high. This signifies cost-benefit analysis and saving.  On the behalf of the cost-benefit analysis, the proposed system is feasible and is economical regarding its pre-assumed cost for making a system. If the system applies on the clinic, it decreases many costs like labor, paper, etc. therefore our system is economic feasible.

### **Technical Feasibility**

Technical feasibility is the measure of practicality of the specific technical solution and the availability of technical resources and expertise, so the project team members have learned programming languages that required for the successful completion of the project such as java script, CSS, HTML, PHP, MySQL and the system was developed by familiar programming language (environment). So our system is technically feasible.

### Operational feasibility

No need of more special human power to run operation because of Bonga University has its own server and ICT center, it is possible to handle or run operation under the existing operation. Since the database to be built is going flexible and user friendly, workers won’t face quite a challenge in trying to use of the new system.

### **Legal feasibility**

The project team members built the system without violating rules and regulations of the governments as well as the organization. The system being built is for the sake of productivity of the organization, so that the project is legally feasible.

### Time feasibility

A project will fail if it takes too long to be completed before it is useful. Time feasibility is a measure of how reasonable the project is completed within the given time. By estimating the given time to each of the activities we will try to complete the project on time. Therefore, our project is timely feasible.

## Project schedule

Once the schedule activities are defined, they are sequenced in the order in which they must be performed. The resource requirements and the activity durations are then estimated for these activities. Finally, the project schedule is created which shows when each activity is scheduled to begin and end. The project schedule also shows the planned start date and planned finish date for the overall project.



Figure 1: Gantt chart for Project schedule

## Project budget

The following table lists budget required for the successful development of the proposed system.

Table 1 : Cost of project

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Types of costs | Tool name | Quantity | Unit price (in Birr) | Total price (in Birr) |
| Hardware costs | PC | 1 | 20000 | 20000 |
| Desktop | 1 | 10000 | 10000 |
| Flash (32 GB) | 2 | 350 | 700 |
| Pen | 3 | 15 | 15 |
| Paper | 100 | 1 | 100 |
| Note book | 1 | 50 | 50 |
| Software cost | Microsoft office 2016 | 1 | Free | Free |
| Visual studio | 1 | Free | Free |
| Wonder share e-draw max | 1 | Free | Free |
| Window’s 10 | 1 | free | Free |
| Total cost | - | - | - | 30865 |

# CHAPTER TWO

# 2. REQUIREMENT ANALYSIS

## Current system description

The current system of Bonga university students’ clinic system is performing its activity through manual way. All registering new patient, writing report, registering drug, lab result from lab technician and prescription writing is performed by manual system.

### Major function of current system

The current system of Bonga university student clinic uses manual based system. In manual system anything is done on paper. Some of current system functions are: -

* Register a new patient
* Search existing patient information from shelf and sent to doctor or nurse
* Doctor or Nurse writes medical perception and report
* Pharmacy or lab technicians write report
* No need of internet connection

### Problem of existing system

There are many problems on current system (using manual) rather than time, cost and human labor. Here are some problems of existing system: -

* Difficult to manage patient information.
* Large storage medium (space) is required to store medical documents, cards, and patient information.
* Searching for even single data is time consuming.
* Loss of patient record
* Redundancy (multiple record of the same data)
* Difficult in preparing an organized report
* Sudden damage on storage medium resulting may zero data
* Circulation of patient information such as (lab report generated by lab technician, Doctor Prescription to pharmacy) is late.

## Requirement Gathering

### Data gathering methods

We used the following methods to collect relevant data required to our project.

* **Interview**: - we gathered necessary information about the background of the clinic, their works activities and the function of their existing system using some structured (when did the clinic was established, how does the existing system function, how many patients get services per week, how many employers are there etc.) and unstructured interview questions.
* **Observation**: We also arrived to the clinic and observed how workers carrying out their work activities in a natural setting. Observation allows us to collect data in real time where activities are being performed.

### Business Rule

A business rule is effectively an operating principle or polices that must be fulfilled and obligated in order the system will function properly and effectively. In the clinic every employee does their activity through their authority.

**BR1:** only doctor and clinic manager has view the patient medical history

**BR2:** only pharmacist gives drug and tell drug taken instruction to the patient ordered by Doctor or Nurse.

**BR3**: test and report, lab result to doctor is the duties of lab technician.

**BR4**: Pharmacist, lab technician and doctor has duties to report to manager.

## Proposed system description

### Overview

The clinic system software that is to be developed is to minimize the problem of current system as it described in the problems of existing system. The systems are effective at the time of registration, update, search, store, and generate report. In the requirement analysis phase, the document we stated describes the functionalities of the system in terms of use case from the users’ point of view. But in the design phase those functionalities of the system will be decomposed into smaller sub system to easily handle by developer. Patient registration system provides a way for the employee and the manager to keep the patient information.

### Functional requirement

The functional requirements for a system describe what the system should do. These requirements depend on the type of software being developed, the expected users of the Software and the general approach taken by the organization when writing requirements. When expressed as user requirements, functional requirements should be written in natural language so that system users and managers can understand them [2].

The new proposed system is aimed to solve the problems in current system by creating web-based system that helps the clinic to manage the patient’s journey throughout the clinic. The system able to: -

* **Register new patient**: Our proposed system can register each patient and records their basic information on database. This helps the clinic to manage and manipulates patients’ information easily and to store their data and documents permanently.
* **Search for existing patient from database**: This enables the users of the system to find patient information from database which is previously registered. When already registered patient come for further treatment, the responsible person should search for that specific patient information. This is done easily by writing patient’s name or medical identification number. It takes only a few seconds to do this task.
* **Update and modify patient information**: The authorized users of the system can add, modify or change patient’s information whenever possible.
* **Show patient information (viewing and accessing**): Our system can display or show the stored basic information of the patients, other clinical documents, reports and records which stored in the clinic database. An authorized individuals can view and access some or all of the patients’ medical treatment information more easily. Patients also can view their own information.
* **Make patient appointment**: The doctor or nurse of the clinic using the system can assign appointment for a particular patient.
* **View appointment:** the patient can see their appointment and the doctor or nurse checks their appointment.
* **Create user account:** Admin creates account for the system user.
* **Delete user account:** Admin delete user account from the system if not more necessary.
* **Update user account:** Admin update user account, if necessary to add or remove something from or to databases of the user’s information.
* **Generate medical reports**: Various medical documentation and reports can be produced based on doctor’s prescription and patient’s sickness condition. These can be:

***Card room report****: -* the system enables manager to produce card department report for number of patient who has new comer and existing.

***Laboratory report****: -* the system enables manager to produce lab reports for individual number of patient result tested on laboratory.

***Order****:* Doctors or nurses can produce consultation and order for each patient who visited them.

* **View reports:** Clinic manager view number of patient per day, week, month and year and number lab test.
* **Feedback:** Our proposed system allow to the patient gives feedback to the manager and the manager to see the feedback.
* **Register employee:** Our proposed system allow to the manager to register new employee to the system.

### Nonfunctional requirement

Besides the functional requirements, our system possesses other requirements that are non-functional requirements, as the name suggests that, the requirements that are not directly concerned with the specific services delivered by the system to its users. These non-functional requirements usually specify or constrain characteristics of the system as a whole [2].

#### Performance: The response time of our system uses to process, quire and retrieve patient data and information from database is very short. That means it takes short response time for a given piece of work. The designed system will use low utilization of system resource in terms of space and time. Many tasks can be performed on the same time that in turns provide time and cost-effective services.

* + - 1. Scalability**:** The ability to add capacity (and users) to a deployed system over a time. Scalability typically involves adding resources to the system but should not require changes to the deployment architecture.
      2. Availability: The system able to give fulltime services to its users without any crash. It is available for 24 hours per day and 7 days per week, unless connection interruption. Is provides real information at right time.
      3. Reliability: Only an authorized users of the system (administrator, manager or other) can able to update, modify, delete or access patient data. Access is denied for unauthorized and unauthenticated users of the system.
      4. Maintainability: Additional functionalities and new features can be easily done because the whole system development tasks divided into several smaller work parts called *modules*. This process is known as *modularization*. Since the system was modularized not the whole system should be maintained rather only the specific modules that need modification will be modified and maintained.
      5. Security: Users of the system must be identified, authenticated and authorized before having access to the system’s services. Users will have their own password and username through which they could gain access to. It introduces a password size limit facility. The password with which user are provide will protect them from potential threats, who could jeopardize their responsibility. Our system protects any confidential information of the clinic as well as the patients from unauthorized users by verifying their user names and passwords.
      6. Usability: it is a term used to denote the ease with which people can employ a particular tool or other human-made object in order to achieve a particular goal. In this case our system possesses the following regarding to usability: -
* Easier to learn: operation can be learned by observing the object
* More satisfying to use.
  + - 1. Interoperability: the software system can interact with other software or hardware systems, or external factors such as safety regulations or privacy legislation.

# CHAPTER THREE

# 3. SYSTEM MODEL

As we mentioned in the above section, in this project, the team members used an object-oriented system development methodology which incorporates two principal phases. In this chapter, what the team will do is the object-oriented analysis (OOA). During Object Oriented Analysis the major activity is modeling the Functions of the system (Use Case Modeling).

The main activities that are performed in this part will be:

* Identifying actors and use cases,
* Constructing a use case model, sequence diagram, and activity diagram.

## Scenario

It identifies detail description of use cases. It describes in which task actors solve the given problems. It can be understood in a variety of ways, several of them useful in systems engineering. A scenario can be a sequence of activities, or a more or less richly branched structure of such sequence. Branches can represent alternatives or parallels, or various intermediate options.

### Use case model

A use case model is a model of how different types of user interact with the system to solve a problem. It describes the goals of the users, the interaction between the users and the system, and the required behavior of the system in satisfying these goals.

1. Actors of the system

**Actor**: is a person, or external system that plays a role in one or more interaction with the system. The following are the identified actors (users) that will be participating in the system.

* Admin
* Manager
* Doctor/Nurse
* Clerk
* Lab technician
* Pharmacist
* Patient

1. **Use Case Identification**

**Use case**: - is a description of set of interaction sequence the system performs to provide a result observable or measurable value to one or more actors. Each Use Case describes the functionality to be built in the proposed system, which can include another Use Case's functionality or extend another Use Case with its own behavior. The most important and basic use cases of this system are the following:-

* Login
* Register new patient
* Register employee
* Search exist patient information
* View patient information
* Update patient information
* Give patient appointment
* Write lab result
* Manage account
* View appointment
* View report
* Give feedback
* View feedback
* Post news
* View news
* Order to pharmacist and lab technician
* Generate report
* View lab result
* Accept order
* Register drug
* Provide drug
* Accept drug
* Logout

### Use Case Diagram

UML offers the use case diagram to enable you to define the requirements that a system must fulfill. This diagram describes which users use which functionalities of the system, But does not address specific details of implementation. Use Case represents interaction between a user (human or machine) and the system [1].

**Use Case Components:**

* **Actor:** is a person, or external system that plays a role in one or more interaction with the system. And represented with:



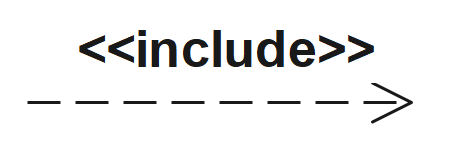
* **Use Case**: describes a sequence of actions that provides something of measurable value to an actor and is drawn as a horizontal ellipse.



* **System Boundary**: indicates the scope of the system project. Anything within the box represent functionalities in side in scope.



* **Connector**: describes the connection between use case and actor.
* **Include**: a way of simplifying a use-case by separating it’s functionality in another use-case. It defines the relationship of the base use case with another use case, the functional behavior of which is always used by the base case of use. The arrow go from base use case to included use case.



* **Extend:** used to extend the behavior of base use case.

A kind of relationship of dependence between the basic use case and its extended use case or special case. It indicates the possibility of special use of the base use case (the arrows go from extended use case to base use cases.

****

****

Figure 2 : Use case diagram for WBSCMS

**1) Admin**: is the one who manage user account

* Activate account
* Dec activate user account
* View news

**2) Manager**: is the one who manages the system.

Main activities of manager include:

* Post news
* Register employee
* View feedback
* Generate report
* View patient information
* Approve drug request

**3) Doctor/Nurse:** A person skilled on medical diagnosis.

Main activities include:

* View patient information
* Update patient information
* Order to pharmacist and lab technician
* View news
* Accept lab result
* Make appointment

**4) Clerk:** An employee worker in card class.

Main activities:

* Register patient
* Search existing patient
* View news

**5) Lab Technician:** skilled person on laboratory.

Main activities:

* Send lab result to doctor
* view news
* accept order from doctor

**6) Pharmacist:** skilled person on pharmacy.

Main activities:

* accept order
* registered drug
* view news
* Accept drug
* View drug information

**7) Patient:** The patient can be defined as the actor that will receive services from the clinic.

The main activities of patient:

* View his/her information
* View his/her appointment
* Give feedback
* View news

### Description of Use Case Model

Use cases are defining what the actor will be able to do with the system. Since all needs of a system typically cannot be covered in one use case, it is usual to have a collection of use cases. The table below shows the descriptions of use case.

Table 2 : use case documentation for login

|  |  |
| --- | --- |
| use case name | Login |
| Identifier | Uc01 |
| Actor | User(employee and patients) |
| Description | This use case allows all users to login to the system by his/her username and password. |
| pre-condition | Users should be registered on the system |
| basic course of action | 1. opens home page and click on login button 2. system show form 3. enter username and password 4. system validate it and display the users home page 5. Use case ends |
| alternative course of actions | If the user enters incorrect information the system display error message. |
| post condition | Users login to the system |
| Goal | Login to the system |

Table 3 : use case documentation for register patient

|  |  |
| --- | --- |
| use case name | Register Patient |
| Identifier | Uc02 |
| Actor | Clerk |
| Description | This use case allow the clerk to register patient information that given from patient. |
| pre-condition | Clerk should be login to the system and the patient should be the regular university student. |
| basic course of action | 1. Login to the system 2. click register button 3. system shows the registration form 4. system validate it and display patient registered successfully message 5. Use case ends |
| alternative course of actions | If the clerk enters incorrect information the system display error message. |
| post condition | Information registered |
| Goal | Clerk to register new patient |

Table 4: use case documentation for search patient information

|  |  |
| --- | --- |
| use case name | Search patient information |
| Identifier | Uc03 |
| Actor | Clerk |
| Description | Clerk has to view basic patient information |
| pre-condition | The clerk logged into the system. |
| basic course of action | 1. clerk click on search text field 2. Search by student ID or first name 3. Clerk checks the patient registered before 4. Click on patient name and send to doctor 5. Use case ends |
| alternative course of actions | If the user enters incorrect information system display error message. |
| post condition | The requested patient information is exist or not |
| Goal | To check the patient info exist before |

Table 5: use case documentation for view patient info

|  |  |
| --- | --- |
| use case name | View patient info |
| Identifier | Uc05 |
| Actor | User (Doctor/Nurse and Patient) |
| Description | Users has to view patient information |
| pre-condition | The users logged into the system. |
| basic course of action | 1. Click view patient information button 2. System show form 3. Users fill patient ID or registration no 4. View the information 5. Use case ends |
| alternative course of actions | If the user enters incorrect information system display error message. |
| post condition | The requested information is viewed |
| Goal | To view patient information |

Table 6: use case documentation for update patient information

|  |  |
| --- | --- |
| use case name | Update patient information |
| Identifier | Uc06 |
| Actor | User (Doctor/Nurse, Manager) |
| Description | Users has to update patient information |
| pre-condition | The users logged into the system. |
| basic course of action | 1. view patient information 2. Add or remove some patient information 3. Click update button 4. Use case ends |
| alternative course of actions | If the user enters incorrect information system display error message. |
| post condition | The required patient information is updated |
| Goal | To update patient information |

Table 7: use case documentation for make appointment

|  |  |
| --- | --- |
| use case name | Make appointment |
| Identifier | Uc07 |
| Actor | Doctor/Nurse |
| Description | Doctor/Nurse has to make appointment for patient |
| pre-condition | The Doctor/Nurse logged into the system. |
| basic course of action | 1. Doctor/Nurse click make appointment button 2. Fill appropriate information 3. Click ok 4. Use case ends |
| alternative course of actions | If the user enters incorrect information system display error message. |
| post condition | The required appointment is made |
| Goal | To make appointment |

Table 8: use case documentation for give feedback

|  |  |
| --- | --- |
| use case name | Give feedback |
| Identifier | Uc09 |
| Actor | Patient |
| Description | Users has to give feedback about the clinic |
| pre-condition | The users logged into the system. |
| basic course of action | 1. Patient selects the feedback button 2. System show form 3. patient fill appropriate information 4. Submit feedback 5. Use case ends |
| alternative course of actions | If the user enters incorrect information system display error message. |
| post condition | The required feedback is submitted |
| Goal | To give feedback |

Table 9: use case documentation for generate report

|  |  |
| --- | --- |
| use case name | Generate Report |
| Identifier | Uc09 |
| Actor | manager |
| Description | manager has to generate report |
| pre-condition | The users logged into the system. |
| basic course of action | 1. manager selects the report button 2. System show form 3. manager fill appropriate information 4. Generate report 5. Use case ends |
| alternative course of actions | If the user enters incorrect information system display error message. |
| post condition | The required report is generated |
| Goal | To generate report |

Table 10: use case documentation for register employee

|  |  |
| --- | --- |
| use case name | Register employee |
| Identifier | Uc10 |
| Actor | manager |
| Description | Manager has to register employee |
| pre-condition | The users logged into the system. |
| basic course of action | 1. Manager selects the register button 2. System show form 3. Users fill appropriate information 4. Register employee 5. Use case ends |
| alternative course of actions | If the user enters incorrect information system display error message. |
| post condition | The required registration is done |
| Goal | To register employee |

Table 11: use case documentation for create account

|  |  |
| --- | --- |
| use case name | Activate account |
| Identifier | Uc11 |
| Actor | Admin |
| Description | Admin can create account |
| pre-condition | The users logged into the system. |
| basic course of action | 1. Admin selects add account 2. System show form 3. Admin fill the appropriate information 4. Then Admin click create button 5. Use case ends |
| alternative course of actions | If the user enters incorrect information system display error message. |
| post condition | The required account is created |
| Goal | To activate account for user |

Table 12: use case documentation for view report

|  |  |
| --- | --- |
| use case name | View report |
| Identifier | Uc13 |
| Actor | Manager |
| Description | Manager can see report sent from different departments on clinic |
| pre-condition | The users logged into the system. |
| basic course of action | 1. Click view report button 2. Enter start and end date 3. See the report 4. Use case end |
| alternative course of actions | If the user enters incorrect information system display error message. |
| post condition | The required report is viewed |
| Goal | To view report |

Table 13: use case documentation for update account

|  |  |
| --- | --- |
| use case name | Update account |
| Identifier | Uc13 |
| Actor | Admin |
| Description | Admin can update account |
| pre-condition | The users logged into the system. |
| basic course of action | 1. Admin selects update account 2. Then Admin click update button 3. Use case ends |
| alternative course of actions | If the user enters incorrect information system display error message. |
| post condition | The required account is updated |
| Goal | To update user account |

Table 14: use case documentation for view appointment

|  |  |
| --- | --- |
| use case name | View appointment |
| Identifier | Uc14 |
| Actor | Patient |
| Description | Patient can view his/her appointment |
| pre-condition | The users logged into the system. |
| basic course of action | 1. Users selects view appointment 2. System show appropriate information 3. Use case ends |
| alternative course of actions | If the user enters incorrect information system display error message. |
| post condition | The appointment is viewed |
| Goal | To view patient appointment |

Table 15: use case documentation for post news

|  |  |
| --- | --- |
| use case name | Post news |
| Identifier | Uc15 |
| Actor | Manager |
| Description | Manager can post news |
| pre-condition | The users logged into the system. |
| basic course of action | 1. Manager open form to write news 2. Write news in the appropriate space 3. Manager click post news button 4. Use case ends |
| alternative course of actions | If the user enters incorrect information system display error message. |
| post condition | The news is posted |
| Goal | To post news |

### Activity Diagram

It is important diagram in UML to describe dynamic aspects of the system. Activity diagram is basically a flow chart to represent the flow form one activity to another activity. The activity can be described as an operation of the system. So the control flow is drawn from one operation to another. This flow can be sequential, branched or concurrent [1].

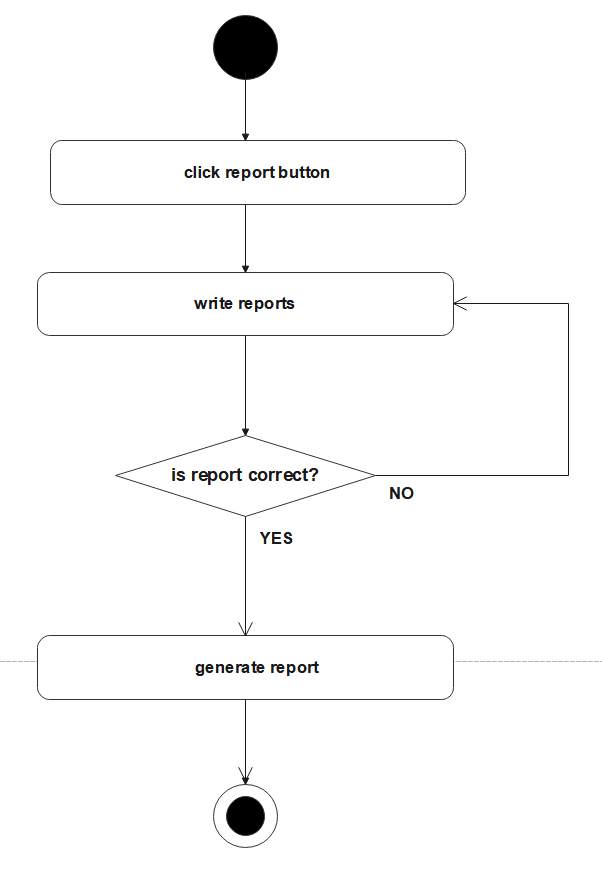


Figure 3: Activity diagram for generate report

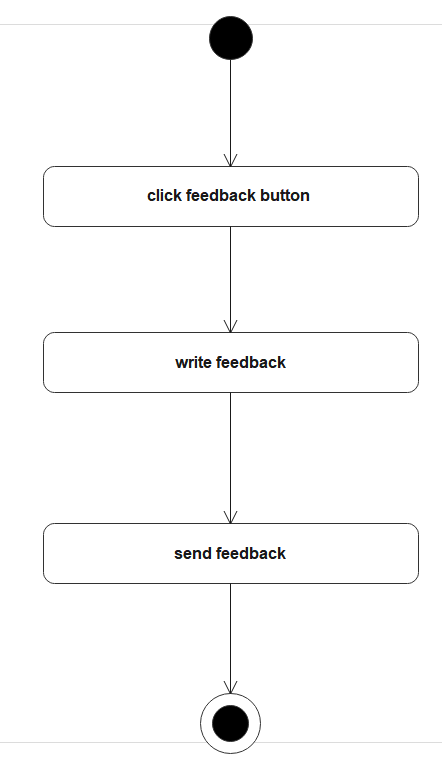


Figure 5: activity diagram for give feedback

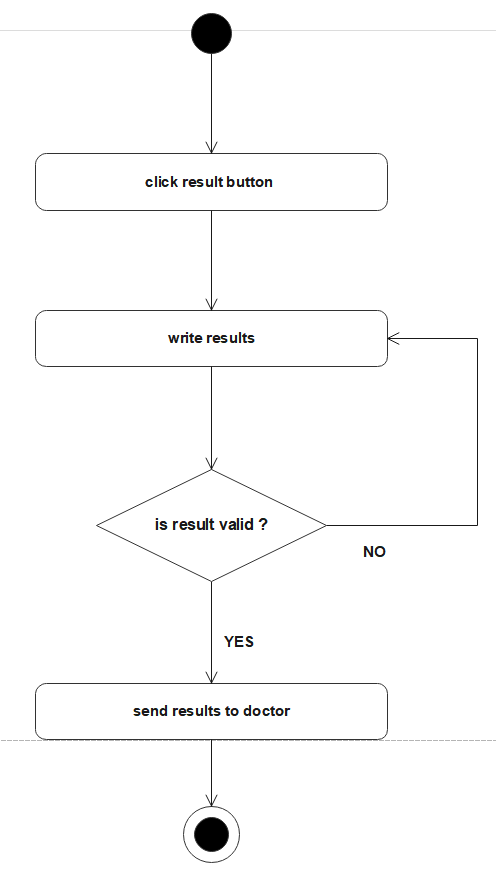


Figure 6: Activity diagram for write lab result

### Data Dictionary

A data dictionary is a file or a set of files that includes a database's metadata. The data dictionary hold records about other objects in the database, such as data ownership, data relationships to other objects, and other data. In our system the following table describe data dictionary. The table below illustrate data dictionary of our system.

Table 16: data dictionary for register patient

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Attribute | Data type | Data size | Constraint | Key Constraint |
| id | int | 10 | NOT NULL | PRIMARY KEY |
| firstName | Varchar(string) | 20 | NOT NULL |  |
| lastName | Varchar(string) | 20 | NOT NULL |  |
| dept | Varchar(string) | 15 | NOT NULL |  |
| Year | int | 5 | NOT NULL |  |
| region | Varchar(string) | 20 | NOT NULL |  |
| zone | Varchar(string) | 20 | NOT NULL |  |
| password | Varchar(string) | 45 | NOT NULL |  |
|  |  |  |  |  |

Table 17: data dictionary for register employee

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Attribute | Data type | Data size | Constraint | Key Constraint |
| firstName | Varchar(string) | 20 | NOT NULL |  |
| lastName | Varchar(string) | 20 | NOT NULL |  |
| postion | Varchar(string) | 15 | NOT NULL |  |
| phoneNo | Varchar(string) | 20 | NOT NULL |  |
| address | Varchar(string) | 20 | NOT NULL |  |
| position | Varchar(string) | 20 | NOT NULL |  |

Table 18: Data dictionary for create account

### Class Model

The class model shows the attributes and the behavior associated with the objects. The class diagram is used to show the class model. The class diagram shows the class name followed by the attributes followed by the functions or the methods that are associated with the object of the class.

****

Figure 7: class model

### Dynamic Modeling

The dynamic model is used to express and model the behavior of the system over time. It includes support for activity diagrams, sequence diagrams, and etc.

**Sequence diagram**

The sequence diagram describes the interactions between objects to fulfill a specific task.



Figure 8: Sequence diagram for register patient

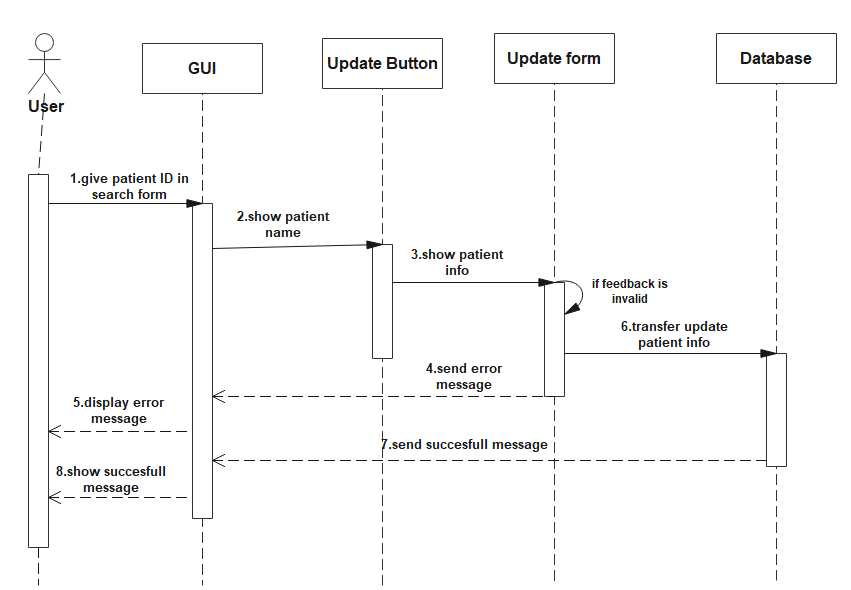


Figure 9: Sequence diagram update patient info

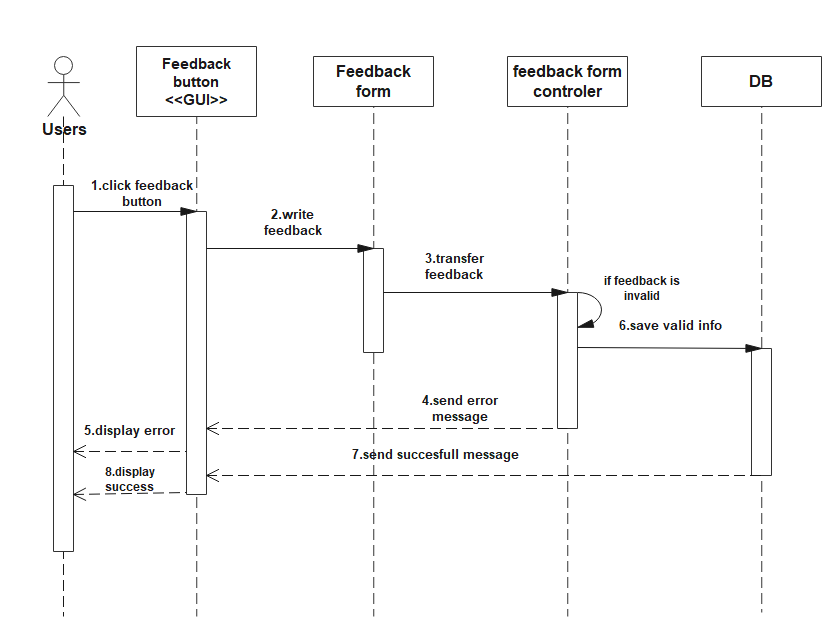


Figure 10: Sequence diagram give feedback

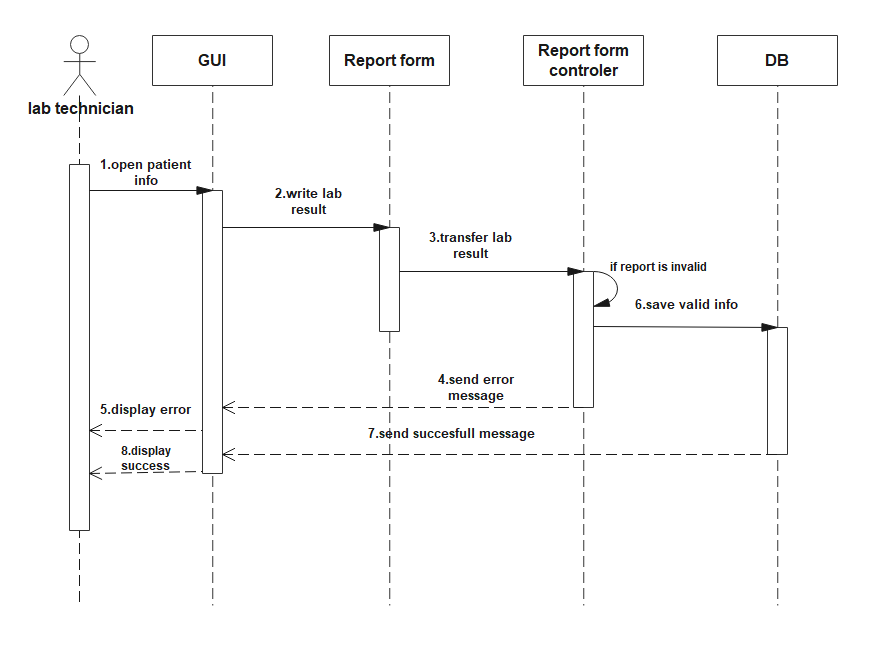


Figure 10: Sequence diagram generate lab



Figure 11: Sequence diagram view patient info

### User Interface

User interface is the external part of the system which is used to access and interact with the system easily. User interface design focuses on the user's experience and interaction. Our goal in user interface design is to make the user's interaction as simple and efficient as possible, in terms of accomplishing user goals. The user interface design should be also flexible. Graphical user interface design is utilized to support its usability by allowing users to interact with the new system through graphical icons and visual indicators.

# CHAPTER FOUR

# 4. SYSTEM DESIGN

## Introduction

The purpose of system design is to transform the analysis model to design model. Project design includes an array of activity form generating ideas to planning how these ideas could be become a realizable project. System design can reduce gap between problem and an exist in machine, decomposes the overall system into manageable parts and uses the principles of cohesion and coherence. The purpose of this design document is to present the system design at level that can be directly trace to specific system objective along with providing more detail data, functional and behavioral requirement. This design document will verify that the current design meets all of the explicit requirements contained in the system model as well as the implicit requirements desired by the user.

## Current software architecture

Currently Bonga university student clinic uses fully manual based system.

## Proposed software architecture

The communication between the client and server is through Http protocol. The second component of the system is a web server on which an application runs and communicates with database to provide responses for the user. There are three tier architectures for the system

1. **The presentation tier**: Clients directly interacted to provide GUI and allow the client gaining access of the system.
2. **Logical tier (middle tier):** acts as bridge between client and server.
3. **Data access tier**: supports data persistence mechanism and storage to the data.

### System decomposition

The whole patient management system decomposed into smaller subsystems called modules. These subsystems are can include two big branches: conceptual modeling which includes system design, and database schema. On the other hand, there is implementation branch which include coding of different subsystems such as: GUI (home pages and forms using HTML and CSS coding), storage subsystem (creation of DB and tables using my SQL), account management subsystem, validation subsystem. This enables us to maintain and modify the system without affecting the whole system. The reason for implementing subsystems in this project is to organize related software classes into groups. This organization is useful to provide a software good overall architecture of the system and helps in the reduction of complexity.

### Hardware/ Software mapping

Hardware software mapping diagrams are used for describing the hardware components where software components are deployed.



Figure 12: Hardware/software mapping

### Persistent data modeling

Persistent data management is basically used to represent the design of the database, usually a Relational Database. Database design is the process of producing a detailed data model of a database. This logical data model contains all the needed logical and physical design choices and physical storage parameters needed to generate a design in a Data Definition Language, which can then be used to create a database. Persistent data management deals with how the persistent data that is the files with database are stored and managed. The overall information of the clinic system for BU student is the persistent data. So it must be stored on a secure database system. The persistence classes are used to store most important and permanent information of the system. The new proposed systems have a number of systems and the persistent data of these systems will be stored in SQL database.

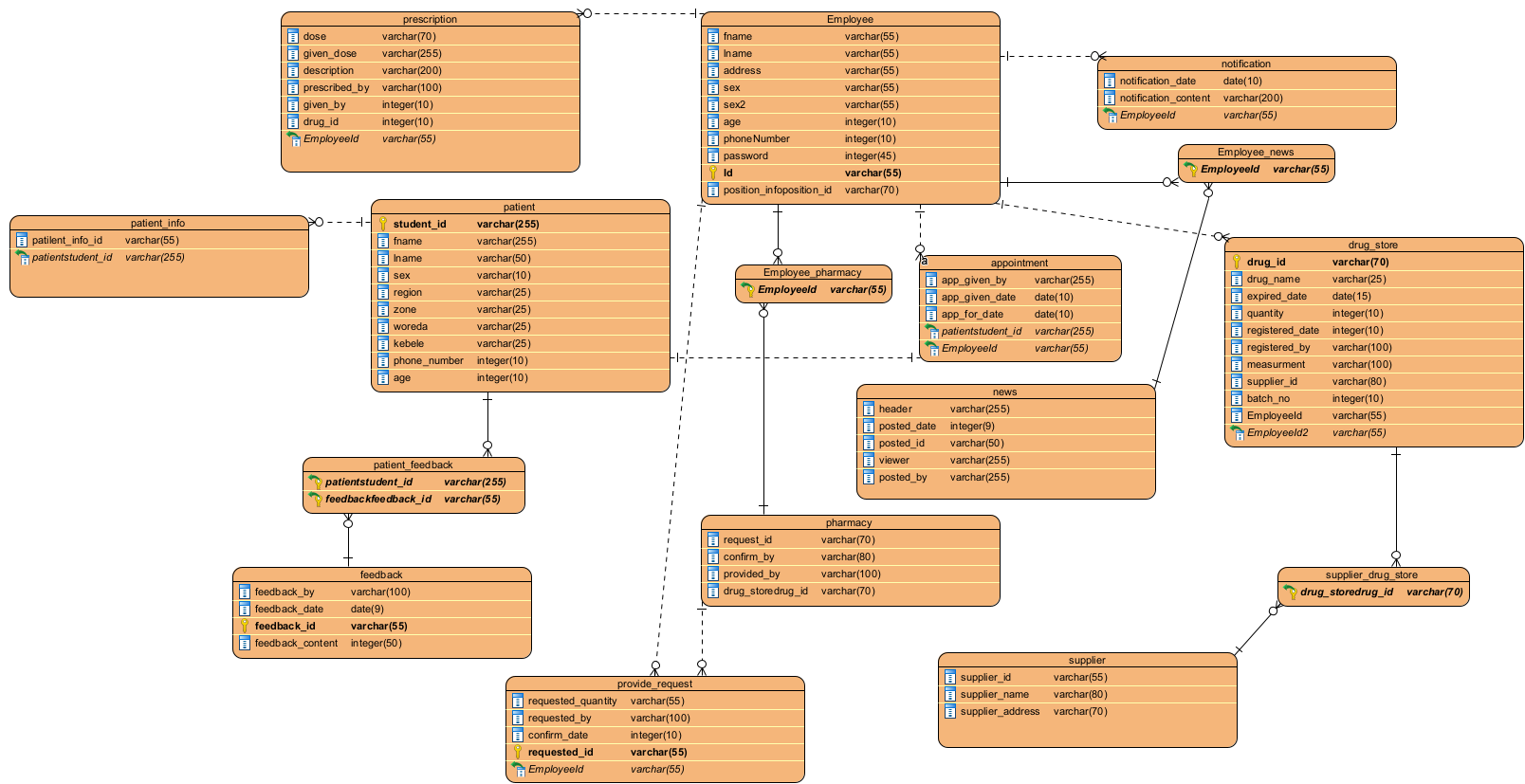


Figure 13: persistent data diagram

### Access control and security

One of the processes is access control and security. The Clinic Management System must establish mutual trust and secure access between the authenticating users, authorizing access, and enforcing security features. Thus, processes establish that a customer and officer site are who user names and passwords, or digital certificates and signatures. The Clinic management system site must then authorize access to only those parts of the site that an individual user needs to accomplish his or her particular communications. Thus, individual usually will be given access to all resources of Clinic management system.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Actors  Use case | Patient | Manager | Clerk | Doctor/nurse | Lab technician | Pharmacist | Admin |
| View Patient info |  |  |  |  |  |  |  |
| Register new patient |  |  |  |  |  |  |  |
| Search exist patient |  |  |  |  |  |  |  |
| Update patient info |  |  |  |  |  |  |  |
| Give appointment |  |  |  |  |  |  |  |
| Write lab result |  |  |  |  |  |  |  |
| Manage account |  |  |  |  |  |  |  |
| View appointment |  |  |  |  |  |  |  |
| Give feedback |  |  |  |  |  |  |  |
| View feedback |  |  |  |  |  |  |  |
| Post news |  |  |  |  |  |  |  |
| View news |  |  |  |  |  |  |  |
| Order |  |  |  |  |  |  |  |
| View lab result |  |  |  |  |  |  |  |
|  | Patient | Manager | Clerk | Doctor/nurse | Lab technician | Pharmacist | Admin |
| Accept order |  |  |  |  |  |  |  |
| Register drug |  |  |  |  |  |  |  |
| Provide drug to pharmacy |  |  |  |  |  |  |  |
| Generate report |  |  |  |  |  |  |  |
| Accept drug |  |  |  |  |  |  |  |
| Register employee |  |  |  |  |  |  |  |

Table 19: access control and privacy

### Detailed class diagram

We use the class diagram to model the static structure of a system, thus Class diagram describing the elements of the system and the relationships between them. These elements and the relationships between them do not change over time [1].



Figure 14: WBSCMS detailed class diagram

### Package diagram

Package is an organized and functionality-based set of related interfaces and classes. Packages organize classes that belong to the same category or provide similar functionality.

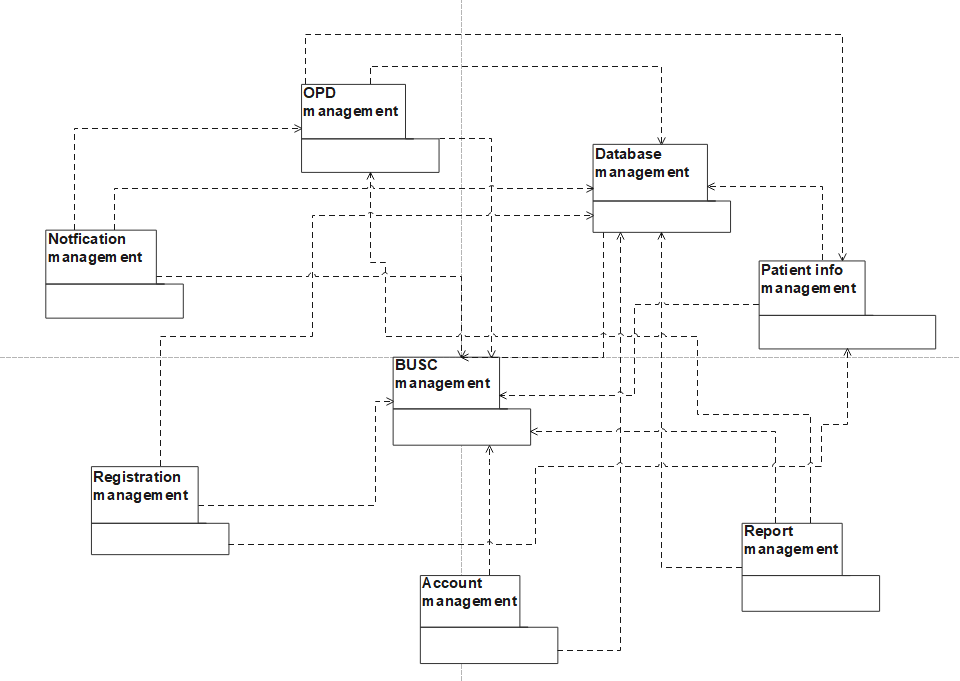


Figure 15: Package diagram

### Deployment

The system is built as a client-server application and communicates any other computers or mobile phone. The system is dependent component that runs on any computer node. UML deployment diagram illustrates the hardware /software mapping of the system. Deployment diagram is implementation type diagram that describes the physical architecture of the hardware and software of the system. It is concerned with the physical deployment of the application. This includes the network layout and location of the components on the network. A deployment modeling depicts a static view of the run time configuration of process nodes and the components that run on these nodes. It shows:

* The hardware for the system
* The software that installed on the hardware
* Middle ware used to connect different machines to one another

For our system we have chosen to adopt a three tires client-server approach in the deployment diagram. The tier is the user interface tier, an application server tier and database server tier.



Figure 16: Deployment diagram

# CHAPTER FIVE

# IMPLEMENTATION

We used different programming languages, markup languages, stylesheet and different JavaScript library such as jQuery, Ajax and structural query language in our project.

**Java script**

A scripting language is a lightweight programming language. The purpose of the statements is to tell the browser what to do. JavaScript to specify the behavior of web pages.

**Php**

* PHP can generate dynamic page content
* PHP can create, open, read, write, delete, and close files on the server
* PHP can collect form data
* PHP can send and receive cookies
* PHP can add, delete, modify data in your database
* PHP can restrict users to access some pages on your website
* PHP can encrypt data

## Mapping Models to Code

**Register\_employee**

**<!DOCTYPE html>**

**<html>**

**<head>**

**<title>Manager | Add employee</title>**

**<link rel="stylesheet" type="text/css" href="include/style.css">**

**<link rel="stylesheet" type="text/css" href="../include/bootstrap/css/bootstrap.css">**

**<script src="../include/bootstrap/js/bootstrap.js"></script>**

**<script src="../include/jquery/jquery-3.6.2.js"></script>**

**<link rel="stylesheet" type="text/css" href="../include/fontawesome/css/all.css">**

**<script src="js/register\_employee.js"></script>**

**</head>**

**<body>**

**<?php include "include/dbc.inc.php"; ?>**

**<?php include 'include/side\_bar.php'; ?>**

**<?php include 'include/navbar.php'; ?>**

**<div class="home">**

**<div class="text">**

**<span class="left\_second\_header">Manager</span>**

**<span class="right\_second\_header"><i class="fas fa-"></i> Employee/ Add</span>**

**</div>**

**<hr class="text-primary">**

**<div class="row content">**

**<!-- -->**

**<div class="col col-12 col-md-12">**

**<div class="inside\_text">**

**Id: <input type="text" name="id" id="emp\_id" onkeyup="validateId(this.value,'id\_error');" onblur="validateId(this.value,'id\_error');"> <br><br>**

**<code id="id\_error" style="display: none;"></code>**

**First Name: <input type="text" name="fname" id="fname" onkeyup="validateName(this.value,'fname\_error');" onblur="validateName(this.value,'fname\_error');"> <br><br>**

**<code id="fname\_error" style="display: none;"></code>**

**Last Name: <input type="text" name="lname" id="lname" onkeyup="validateName(this.value,'lname\_error');" onblur="validateName(this.value,'lname\_error');"> <br><br>**

**<code id="lname\_error" style="display: none;"></code>**

**Position: <select id="position" onchange="validatePosition(this.value,'position\_error');" onblur="validatePosition(this.value,'position\_error');">**

**<option value="">Select Position</option>**

**<option value="Doctor">Doctor</option>**

**<option value="Pharmacist">Pharmacist</option>**

**<option value="Lab technician">Lab Technician</option>**

**<option value="Clerk">Clerk</option>**

**</select> <br><br>**

**<code id="position\_error" style="display: none;"></code>**

**Phone Number: <input type="text" name="phone\_no" id="phone\_no" onblur="validatePhonNo(this.value, 'phone\_no\_error');"> <br><br>**

**<code id="phone\_no\_error" style="display: none;"></code>**

**Address: <textarea id="address"></textarea><br><br>**

**<code id="address\_error" style="display: none;"></code>**

**Salary: <input type="number" name="salary" id="salary" min="1" onkeyup="validateSalary(this.value, 'salary\_error');" onblur="validateSalary(this.value, 'salary\_error');"> <br><br>**

**<code id="salary\_error" style="display: none;"></code>**

**<button class="btn btn-success" type="button" name="submit" id="register">Register</button><br>**

**<hr class="text-success">**

**<div id="display\_reg\_success">**

**</div>**

**</div>**

**</div>**

**</div>**

**<?php //include 'include/footer.html'; ?>**

**</div>**

**<script src="js/script.js"></script>**

**<script type="text/javascript">**

**var subMenu = document.getElementById('subMenu');**

**function toggleMenu(){**

**subMenu.classList.toggle('open-menu');**

**}**

**</script>**

**</body>**

**</html>**

**Register\_drug.html**

**<div class="container">**

**<div class="mb-2">**

**<label>Drug Name: </label>**

**<input type="text" name="drug\_name" id="drug\_name" onkeyup="validateName(this.value, 'drug\_name\_error');">**

**<code id="drug\_name\_error" style="display: none;"></code>**

**</div>**

**<div class="mb-2">**

**<label>Qty. : </label>**

**<input type="number" id="drug\_qty" onkeyup="validateQty(this.value, 'drug\_qty\_error');">**

**<code id="drug\_qty\_error" style="display: none;"></code>**

**</div>**

**<div class="mb-2">**

**<label>Measure: </label>**

**<select id="drug\_measure" onchange="validateMeasure(this.value, 'drug\_measure\_error');" onblur="validateMeasure(this.value, 'drug\_measure\_error');">**

**<option value="default">----select measure-----</option>**

**<option value='box'>Box</option>**

**<option value="pk">PK</option>**

**<option value='doozen'>Doozen</option>**

**<option value="tube">Tub</option>**

**<option value='roll'>Roll</option>**

**<option value="littre">Littre</option>**

**</select>**

**<code id="drug\_measure\_error" style="display: none;"></code>**

**</div class="mb-2">**

**<div class="mb-2">**

**<label>Batch No. : </label>**

**<input type="number" id="drug\_batch\_no" onkeyup="validateBatchNo(this.value,'drug\_batch\_no\_error');">**

**<code id="drug\_batch\_no\_error" style="display: none;"></code>**

**</div>**

**<div class="mb-2">**

**<label>Exp. Date: </label>**

**<input type="date" id="drug\_expire\_date" onblur="validateDate(this.value, 'drug\_expire\_date\_error');">**

**<code id="drug\_expire\_date\_error" style="display: none;"></code>**

**</div>**

**<div class=" mb-2">**

**<br><button class="btn btn-success" id="add\_drug">Add drug</button>**

**</div>**

**</div>**

**<div>**

**<table class="table" id="tbl">**

**<thead>**

**<tr>**

**<th>#</th>**

**<th>Drug Name</th>**

**<th>Quantity</th>**

**<th>Measure</th>**

**<th>Batch No</th>**

**<th>Expire Date</th>**

**<th>Action</th>**

**</tr>**

**</thead>**

**<tbody>**

**</tbody>**

**</table>**

**</div>**

**<div class="mt-5 text-center ">**

**<button class="btn btn-success" id="register">Register </button>**

**</div>**

**Register\_drug.php**

**<!DOCTYPE html>**

**<html>**

**<head>**

**<title>Pharmacist | Register Drug</title>**

**<script src="js/validate.js"></script>**

**</head>**

**<body>**

**<?php include\_once "include/header.php"; ?>**

**<div class="column" style="border: 2px solid blue;">**

**<?php require 'section/register\_drug.html'; ?>**

**</div>**

**</div>**

**</div>**

**<script src="js/register\_drug.js"></script>**

**</body>**

**</html>**

**Register\_drug.js**

**var edit\_row = null;**

**$(document).ready(function(){**

**var count = 0;**

**var a = 0;**

**$("#add\_drug").click(function(){**

**var drug\_name = document.getElementById('drug\_name');**

**var drug\_qty = document.getElementById('drug\_qty');**

**var drug\_measure = document.getElementById('drug\_measure');**

**var drug\_batch\_no = document.getElementById('drug\_batch\_no');**

**var drug\_expire\_date = document.getElementById('drug\_expire\_date');**

**if (!validateName(drug\_name.value, 'drug\_name\_error'))**

**drug\_name.focus();**

**else if(!validateQty(drug\_qty.value,'drug\_qty\_error'))**

**drug\_qty.focus();**

**else if(!validateMeasure(drug\_measure.value,'drug\_measure\_error'))**

**drug\_measure.focus();**

**else if(!validateBatchNo(drug\_batch\_no.value, 'drug\_batch\_no\_error'))**

**drug\_batch\_no.focus();**

**else if(!validateDate(drug\_expire\_date.value, 'drug\_expire\_date\_error'))**

**drug\_expire\_date.focus();**

**else{**

**if(edit\_row == null){**

**count += 1;**

**var drug\_name = document.getElementById('drug\_name').value;**

**var drug\_qty = document.getElementById('drug\_qty').value;**

**var drug\_measure = document.getElementById('drug\_measure').value;**

**var drug\_batch\_no = document.getElementById('drug\_batch\_no').value;**

**var drug\_expire\_date = document.getElementById('drug\_expire\_date').value;**

**var insert\_table = "<tr id='row"+count+"'>";**

**insert\_table += "<td>"+count+"</td>";**

**insert\_table += "<td class='drug\_name'>"+drug\_name+"</td>";**

**insert\_table += "<td class='drug\_qty'>"+drug\_qty+"</td>";**

**insert\_table += "<td class='drug\_measure'>"+drug\_measure+"</td>";**

**insert\_table += "<td class='drug\_batch\_no'>"+drug\_batch\_no+"</td>";**

**insert\_table += "<td class='drug\_expire\_date'>"+drug\_expire\_date+"</td>";**

**insert\_table += "<td> <button class='btn btn-primary edit' name='edit' data-row='row"+count+"'>Edit</button>"+**

**"<button class='btn btn-danger delete' name='delete' data-row='row"+count+"'>Delete</button> </td>";**

**insert\_table += "</tr>";**

**$("#tbl").append(insert\_table);**

**}else{**

**var drug\_name = document.getElementById('drug\_name').value;**

**var drug\_qty = document.getElementById('drug\_qty').value;**

**var drug\_measure = document.getElementById('drug\_measure').value;**

**var drug\_batch\_no = document.getElementById('drug\_batch\_no').value;**

**var drug\_expire\_date = document.getElementById('drug\_expire\_date').value;**

**document.getElementById(edit\_row).childNodes[1].innerHTML = drug\_name;**

**document.getElementById(edit\_row).childNodes[2].innerHTML = drug\_qty;**

**document.getElementById(edit\_row).childNodes[3].innerHTML = drug\_measure;**

**document.getElementById(edit\_row).childNodes[4].innerHTML = drug\_batch\_no;**

**document.getElementById(edit\_row).childNodes[5].innerHTML = drug\_expire\_date;**

**}**

**resetData();**

**}**

**});**

**$(document).on('click','.edit',function(){**

**edit\_row = $(this).data('row');**

**var drug\_name = document.getElementById(edit\_row).childNodes[1].innerHTML;**

**var drug\_qty = document.getElementById(edit\_row).childNodes[2].innerHTML;**

**var drug\_measure = document.getElementById(edit\_row).childNodes[3].innerHTML;**

**var drug\_batch\_no = document.getElementById(edit\_row).childNodes[4].innerHTML;**

**var drug\_expire\_date = document.getElementById(edit\_row).childNodes[5].innerHTML;**

**document.getElementById('drug\_name').value = drug\_name;**

**document.getElementById('drug\_qty').value = drug\_qty;**

**document.getElementById('drug\_measure').value = drug\_measure;**

**document.getElementById('drug\_batch\_no').value = drug\_batch\_no;**

**document.getElementById('drug\_expire\_date').value = drug\_expire\_date;**

**});**

**$(document).on('click','.delete',function(){**

**var delete\_row = $(this).data('row');**

**if(confirm("Are You Sure delete the row?"))**

**$("#"+delete\_row).remove();**

**});**

**$("#register").click(function(){**

**if(document.getElementById("tbl").getElementsByTagName('tbody')[0].childNodes[1] != undefined) {**

**if(confirm("Are you sure to Register the data listed below?")){**

**var drug\_name = [];**

**var drug\_qty = [];**

**var drug\_measure = [];**

**var drug\_batch\_no = [];**

**var drug\_expire\_date = [];**

**$(".drug\_name").each(function(){**

**drug\_name.push($(this).text());**

**});**

**$(".drug\_qty").each(function(){**

**drug\_qty.push($(this).text());**

**});**

**$(".drug\_measure").each(function(){**

**drug\_measure.push($(this).text());**

**});**

**$(".drug\_batch\_no").each(function(){**

**drug\_batch\_no.push($(this).text());**

**});**

**$(".drug\_expire\_date").each(function(){**

**drug\_expire\_date.push($(this).text());**

**});**

**$.ajax({**

**url: 'php/save\_drug\_db.php',**

**method: 'post',**

**data:{name:drug\_name,qty:drug\_qty,measure:drug\_measure,batch\_no:drug\_batch\_no,expire\_date:drug\_expire\_date},**

**success:function(response){**

**if(response == 1){**

**alert("Data Inserted Succesfully");**

**document.getElementById("tbl").getElementsByTagName('tbody')[0].innerHTML = "";**

**}else{**

**alert(response);**

**}**

**}**

**});**

**}**

**}else{**

**alert("Please fill data to Register");**

**}**

**});**

**});**

**function resetData(){**

**document.getElementById('drug\_name').value = "";**

**document.getElementById('drug\_qty').value = "";**

**document.getElementById('drug\_measure').value = "";**

**document.getElementById('drug\_batch\_no').value = "";**

**document.getElementById('drug\_expire\_date').value = "";**

**edit\_row = null;**

**}**

## Screen Images

Figure 20: Register drug to store

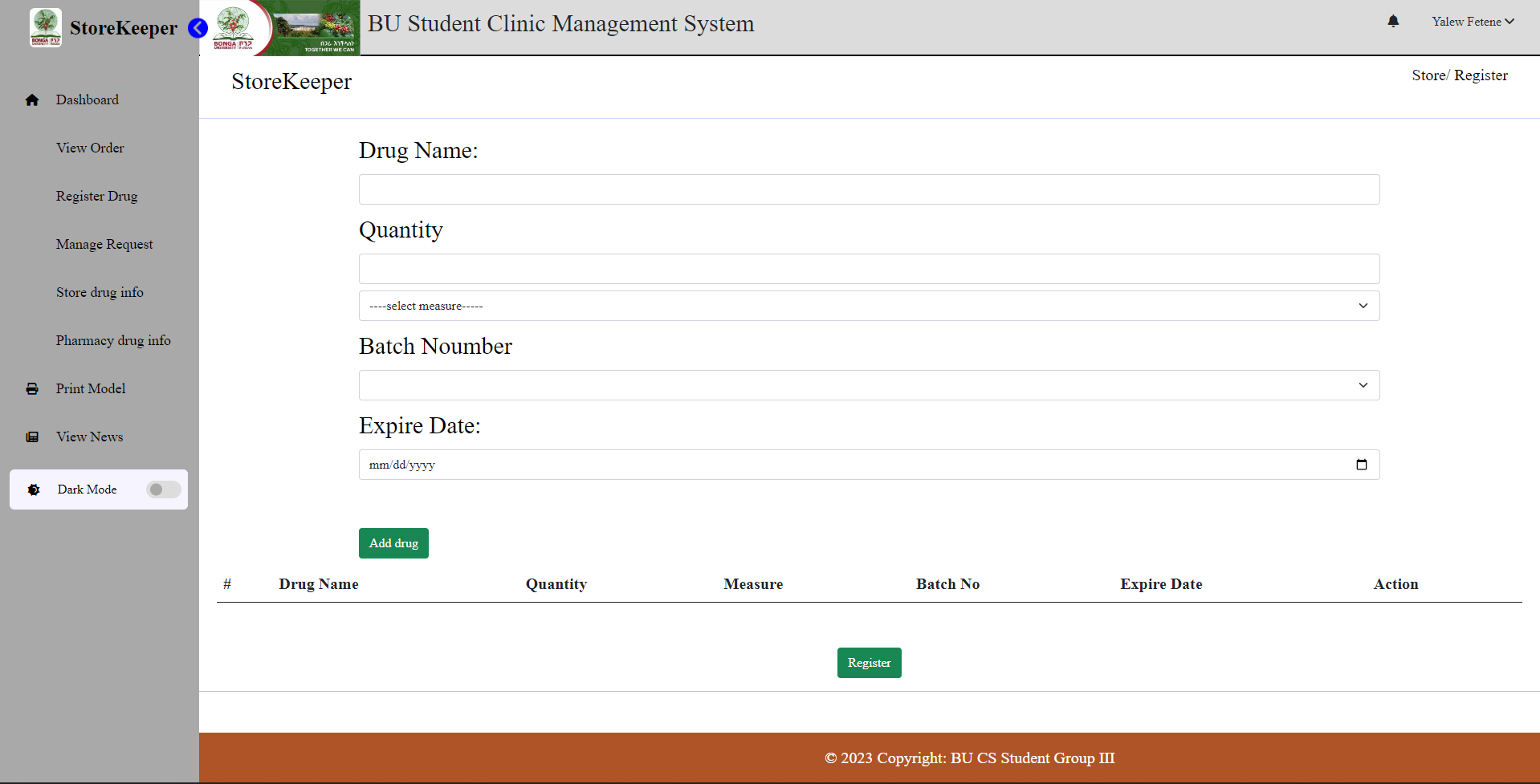


Figure 21: View store drug information

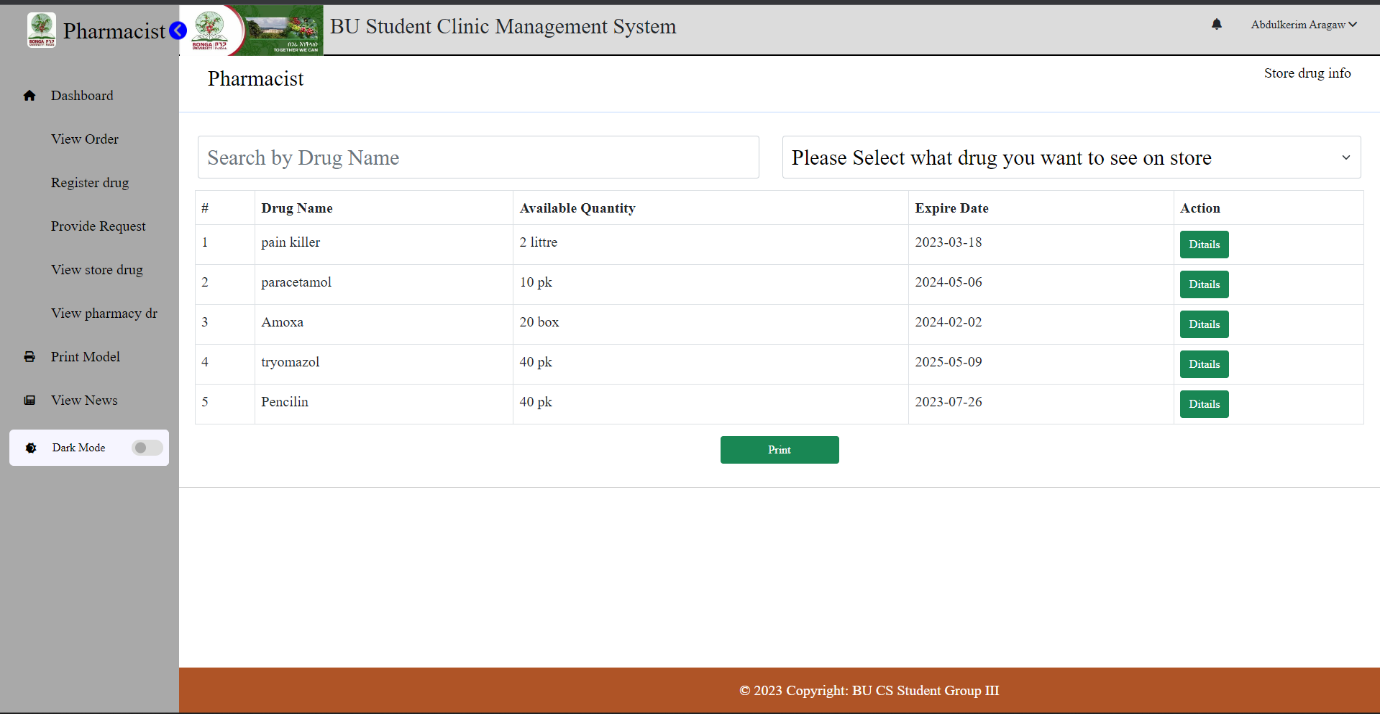
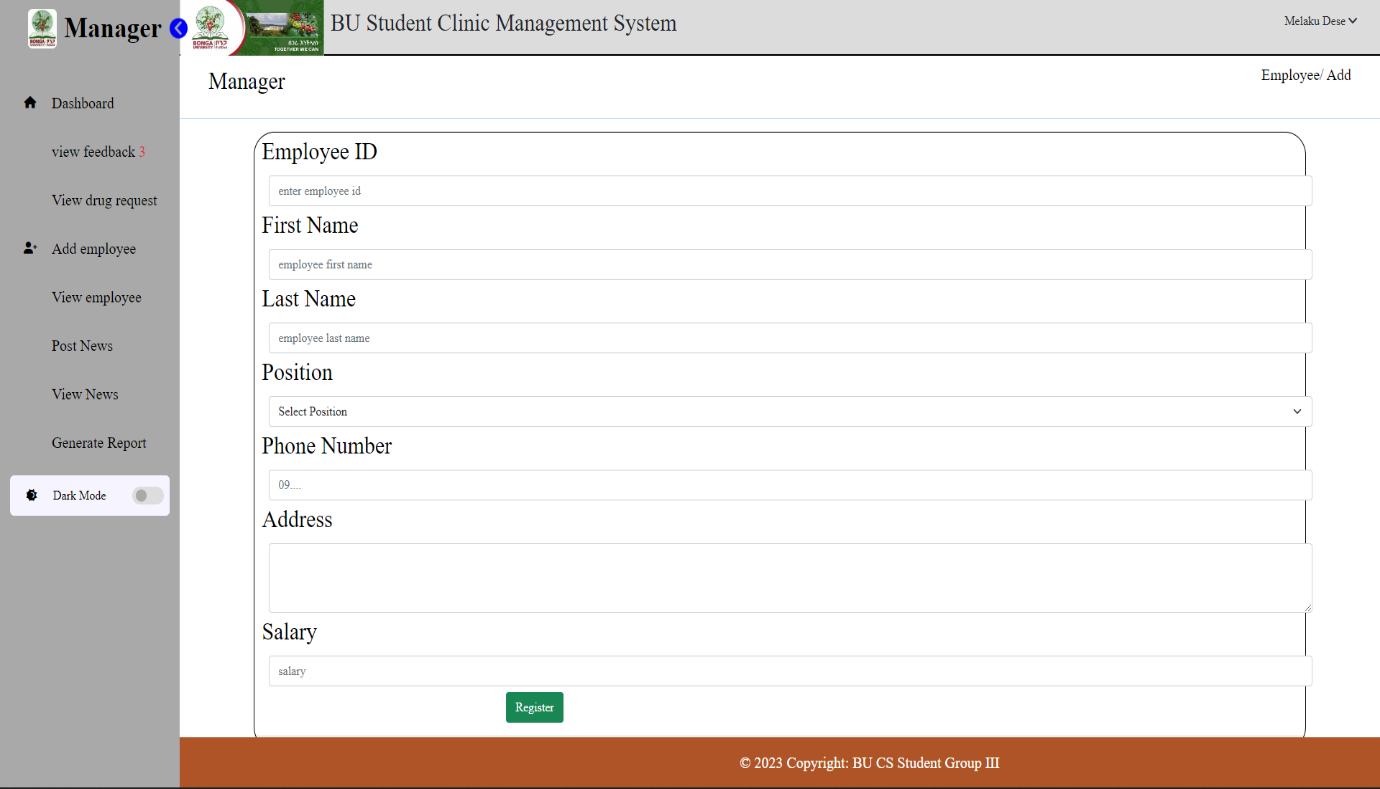


Figure 22: Register employee



## Testing and validation

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub-assemblies, assemblies and/or a finished products.

We test our project in various ways to ensure the system is function properly and we detect errors and then solve the problems that appear in our systems.

### Unit testing

It is the testing of individual software units of the application. We test the design of test cases that validate that the internal program logic is functioning properly, and that program accept inputs and produce valid outputs. All decision branches and internal code flow should be validated.

### White box testing

In white box testing the software tester has the knowledge of its internal code structure and the inner working language of software. We test internal structure, coding and design to verify input output flow and improve security of the system. We test the areas that cannot reached to black box level.

### Black box testing

In black box testing the software tester has the knowledge of the functionalities of the system only buy he has the knowledge of internal coding and inner working language of software.

We test in our project in case of black box testing the functionalities of the systems like register patient, search existing patient info, about drug related activities and the circulation among doctors, lab technician and pharmacists.

# CHAPTER SIX

# CONCLUSION AND RECOMMENDATION

## Conclusion

Due to the abundance of patients and the growing number of operations at the BU student clinic, the handling of paperwork had begun to bottleneck and the updating of records had been difficult. The purpose of this project was to enable the clinic to carefully document all necessary patient and related information and to manage and manipulate patients’ information effectively. The system also helps them for the proper functioning of their operation in the considerable time and accuracy. With all of that in mind, the newly proposed system will allow doctors, manager, system administrators, clerks, lab technicians and pharmacists to maintain and manage patient’ information and transactions more efficiently and securely. The proposed partial system will make it possible to maintain accurate up-to-date records, access various types of users at any moment, and allow certain users the capacity to print or display various reports related to patients and/or clinic.

Through several iterations of analysis and design, the newly designed system believes that the proposed system will be much more efficient and will allow the daily operations of the clinic to run more smoothly.

## Recommendation

According to scope of our project the team develops the web based application for Bonga University student clinic, in the future the team member recommended the following things to be added:

1. We develop this web site only for Bonga University student clinic, if there exist a case that needs to be referred for Gebretsadik Shawo general hospital the system of hospitals and clinic must be connected with each other and to connect them first there have to be a developed system for Gebretsadik Shawo general hospital.

2. This system uses user name, password and roles as main security mechanism we recommended for the next to develop identification of digital signature and finger print of all authorized users.

3. This system is not helpful for visually impaired peoples, so for the future we recommend system like this to be helpful for all peoples.

4. This web site is using English language only we recommend it translated to different national and regional languages.

5. The system does not schedule the working day of employee, we recommend the system automatically schedule to employee the working day and show on their pages.

## References

[1]. UML @ Classroom Martina Seidl · Marion Scholz Christian Huemer · Gerti Kappel an Introduction to Object-Oriented Modeling

[2]. Software Engineering Tenth Edition Ian Sommerville

[3]. Foundation of software engineering Ashfaque Ahmed and Bhanu Prased

## Appendix

Interview Questions

1. How many departments are there in the clinic?
2. What are the departments?
3. What is the responsibility of the departments?
4. The flow of actions or the communications of the department
5. What are the problems in each department?
6. What is the exist system?
7. The problem in using exist system?
8. What are the business rules?

Interviewee name: Mr. Kifile Gichilo (clinic manager)