

UNIVERSITY OF BUEA

Faculty of Science

Department of Computer Science

BACHELOR OF SCIENCE IN COMPUTER SCIENCE

Project Report
STORE MANAGEMENT APPLICATION

DONE BY
CYPRIAN TINGIWEH SIMTENYUI
(SC17A317)

SUPERVISOR
DENIS L. NKWETHEYIM, Ph.D.

JULY 2020

DECLARATION

I CYPRIAN TINGIWEH S. hereby declares that the project Stores Management Application System (developed in Express Node JS and Angular) and its accompanying report have been developed and written by me and have not received any previous academic credit at this or any other institution.

CYPRIAN TINGIWEH SIMTENYUI
(SC17A317)
Department of Computer Science
Faculty of Science

CERTIFICATION

This is to certify that this report entitled “ STORE MANAGEMENT APPLICATION SYSTEM ” is the original work of CYPRIAN TINGIWEH S. with Registration Number SC17A317, student of the Department of Computer Science at the University of Buea. All borrowed ideas and materials have been duly acknowledged by means of references and citations. The report was supervised in accordance with the procedures laid down by the University of Buea. It has been read and approved by:

DENIS L. NKWETEYIM, Ph.D.
University of Buea

Date

DENIS L. NKWETEYIM, Ph.D.
Head of Department of Computer Science

Date

ACKNOWLEDGMENT

This end of year project has been realized with the motivation and encouragement of so many persons that, I find it impossible adequately, to express my gratitude for all the help received.

I am particularly grateful to the following persons:

- Dr. Denis L. Nkweteyim my supervisor for displaying extraordinary patience in going through this work and making very salient and brilliant suggestions
- My parent Mr Tingiweh George S. and Mrs Tingiweh Philomen Ndeh for their financial and emotional support to realized not only this project but all my achievements.
- My benefactors and relatives, especially Mr Tumba Afred and Mrs Muma Immaculate for their financial supports to see that I realized this project.
- My classmates and friends, especially Mark Ngoran, Chiatia Carson, Lumba Loenard and Mformi Blandine for making my time at the University a wonderful one.
- The last but not the least, the Almighty GOD. For YOU, I adore for all my being.

ABSTRACT

The project titled Store Management Application System is developed with the following technologies: Express Node JS framework, MySQL database server and Angular framework. The System developed is Desktop based and designed to run on all platforms with a Java Runtime Environment. It is equipped with a MySQL Database that holds the data processed in the system. The project is developed based on findings, data and transaction records obtained from REPACOL Mutengene Management. The system will help reduce drastically the amount of manual data processing involve in the management operations of REPACOL store and others who will procure the software. It will keep of track supply goods and services, and their distribution to their respective end users and others.

TABLE OF CONTENT

DECLARATION	2
CERTIFICATION	3
ACKNOWLEDGMENT	4
ABSTRACT	5
TABLE OF CONTENT	6
CHAPTER 1: INTRODUCTION	5
1. Brief History of Computer	8
2. Computerized systems ideology with benefits today	8
3. Existing System Description	8
4. Objective of the Project	9
1. General Objective of the projects	9
2. Specific Objective	9
5. Scope	9
1. Scope In	9
2. Scope Out	9
6. Report Structure	9
1. Chapter One Outline	10
2. Chapter Two Outline	10
3. Chapter Three Outline	10
4. Chapter Four Outline	10
5. References	10
Chapter 2: Analysis And Design	11
1. Introduction	11
2. Material	11
1. Hardware Requirements	11
2. Software Requirements	11
3. Methodology	11
1. Problem Definition	11
2. Feasibility Analysis	12
1. Economic Feasibility	12
2. Technical Feasibility	12
3. Time Feasibility	12
3. Analysis	13
1. Non-functional Requirements And Constraints	13
2. Functional Requirements	13
3. Product Requirements Specification	13
4. Process Requirements Specification	13
5. Logical Model	14
1. Data Flow Diagrams	14
2. E-R Diagrams	15
4. Design	15
1. Purpose And Goal Design	17
2. High Level Design	18
1. UML Diagrams	18

2. System Flowchart-----	18
3. User Interface Design-----	18
5. Testing-----	21
1. Unit Testing-----	21
2. Integration Testing-----	21
3. System Testing-----	21
6. Implementation-----	22
7. Deployment Diagrams-----	27
Chapter 3: Result And Discussion-----	27
1. Introduction -----	27
2. Results-----	27
3. Discussion-----	27
1. Realized Specification-----	27
2. Unrealized Specification-----	27
CHAPTER 4: Conclusion-----	29
References-----	30

CHAPTER ONE INTRODUCTION

1.1. Brief History of Computer

Early computers were only conceived as calculating devices. Since ancient times, simple manual devices like the abacus aided people in doing calculations. Early in the Industrial Revolution, some mechanical devices were built to automate long tedious tasks, such as guiding patterns for looms. More sophisticated electrical machines did specialized analog calculations in the early 20th century. The first digital electronic calculating machines were developed during World War II. The first semiconductor transistors in the late 1940s were followed by the silicon-based MOSFET (MOS transistor) and monolithic integrated circuit (IC) chip technologies in the late 1950s, leading to the microprocessor and the microcomputer revolution in the 1970s. The speed, power and versatility of computers have been increasing dramatically ever since then, with MOS transistor count increasing at a rapid pace (as predicted by Moore's law), leading to the Digital Revolution during the late 20th to early 21st centuries.[1]

1.2. Computerized systems ideology with benefits today

Computerized systems now, that were tools meant only for scientists, mathematicians, and engineers are widely used by most organizations. Today, many businesses and organizations use computerized systems to manage different areas of their business. Computerized systems are changing our language, our habits, our environment and, in general, our lives. In Cameroon today, people use computerized systems on a daily basis. For example:

- Civil servants using Automated Teller Machines to receive their salaries.
 - Paying of electricity, school fees online.
 - Sending and receiving money electronically through agencies such as Expression Union, Express Exchange etc.
 - Bio-metric System that manages elections in Cameroon.
- Etc.

The advantages of using a computerized system are enormous and cannot be overemphasized. Most businesses in Cameroon are getting themselves computerized in order to render faster and more efficient services to their customers. Despite this awareness and the realization of the power of computerized systems by many business men and women in different sectors in Cameroon, store management in most organizations is still manually; therefore, lagging behind the technologies of today's world.

Store management system is an automated goods procurement process, receipt of goods, stock book entry, bill passing, issue of material for departments, stock book maintenance, and write-offs for the organization. It generates various store reports.[2] Stores offer quality storage of products for every office or room in an organization, homes, business centers etc.

The concern on the application is for an organization to track goods, services and equipment belonging to the organization. Example of categories of goods: Equipment (Computers,

Air conditions, printers, etc.); furniture (Tables, Chairs, etc.); stationary (paper, toner, etc.).

1.3. EXISTING SYSTEM DESCRIPTION

In most establishments there is still manual Store keeping system to manage the organization's resources such as stationary, furniture's Equipment, materials cleaning tools, clothing etc. The stores in most organizations are ruled or governed by one manager and they have their own rule to govern this system.

1. For example: Any material received may be registered in a book.
2. First if any office wants any material from them, the office must call to the manager of stores to check whether the material they want is available or not, if available the office worker prepare three papers for request; one paper kept in the office, one for store manager and the other one for store keeper.
3. After the store manager accept the request he/she sign on the papers then the store keepers check whether the paper is signed by the store manager or not, after check the store keeper register the materials and then give it out.

1.4. Objective of the Project

1.4.1. General Objective

The general objective of this project is to develop automated Web Application system to track supply of goods and services, and their distribution to their respective end users in order to eradicate the manual Store management system of many organizations.

1.4.2. Specific objective:

The specific objectives of the project are:-

- I. **Register resources**:-The proposed system registers all the resources that entered the organization.
- II. **Distribution**:-The proposed system distribute organizations registered resources into stores.
- III **Send request**:-The request sent from offices(offices) to get any service from the store.
- IV. **Send resources**:-The proposed system sends goods from store to offices.
- V. **Register Salvage resource**: To register resources such as computers, furniture's and others. It helps to know the number of resources that are used and returned to the store.
- VI. **Update Resources Data**: It updates the resource information when needed.
- VII. **Search resource Data**: To search resource data within short period of time.
- VIII. **Generate report**: To generate monthly as well as yearly report

1.5. Scope

1.5.1. Scope in

The scopes of my project are:-

1. Increase to the work efficiency of the office.
2. Saving time that takes for sending request.

3. Reducing unwisely used materials.
4. Checking expired date of materials. This proposed system shows warning when the materials expired.
5. Generating report monthly and yearly.
6. Registering the outgoing materials and receiving the incoming materials.

1.6.2. Scope out

The project has its own scope out. The scopes out of this project is:-
The system cannot provide online service for the organization

1.6 Report structured

1.6.1. Chapter One outline - This chapter gives a general introduction to the project with a brief background on the evolution of computers and its use of computing technology in the world and particularly Cameroon, and the relevance of this project.

1.6.2. Chapter Two outline - The chapter is titled Analysis and Design. It explains how the system was conceived, designed and implemented. It also gives a brief narrative of the methodology used in developing the system.

1.6.3. Chapter Three outline- This is the core of this project and report. Here, an evaluation of the project in terms of implementation is made. The chapter highlights what was achieved and equally what was not. It explains how quality was achieved.

1.6.4. Chapter Four outline- This chapter outlines some suggestions and recommendations for anyone who might want to carry out a similar project.

1.6.5. References - This section of the report lists all the sources from where ideas and material were gotten to enable the project succeed.

CHAPTER TWO ANALYSIS AND DESIGN

2.1 - Introduction

The world is changing at the speed of light and characteristics of impatient seems to have shadow everyone and time being the resource at hand. People always want to get information and perform the task they desire within a short period of time and with a high degree of efficiency and accuracy. Thus developers of computerized systems turns to focus on performance, efficiency and accuracy as top most priorities. In order to realize these objectives, the correct material and methodology must be used.

2.2 - Material

2.2.1 - Hardware Requirements

The following hardware are needed to develop the system.

- Laptop computer,
- WiFi Modem,
- Flash disk
- Standard Color Monitor

2.2.2 - Software Requirements

The following software are needed in the development.

- MySQL database workbench and server.
- Visual Studio Code(VS code)
- Microsoft Office Word,
- Advance Installer
- Linus OS 18.2

2.3 - Methodology

In developing the system, I have chosen to use the **Waterfall methodology** because the requirements specifications were well defined in the project document issued by the department. This methodology was devised in the early days of software development. The methodology divides the system development process into seven distinct phases as illustrated below: Problem Definition, Feasibility study, Analysis, Design, Implementation, Testing, and Maintenance.

Waterfall methodology-It says all the phases of Software Development Life Cycle will function one after another in linear manner.[3]

2.3.1 - Problem Definition

You should not begin to solve a problem until you have defined the problem you are going to solve. This may seem a rather obvious statement, but a surprising number of unsuccessful projects have begun with the developer not having a clear idea of what it is he/she is tackling. With the case in question, I defined the problem this project is set out to solve as follows.

Experience have found the following problems in existing manual store system of most Items and materials record files:- when items and materials are enter to the store, the storekeeper records the items and materials on paper. Due

to this problem records of items and materials are lost and expose the store for data loss. This problem not only affects the store but also affect the working system of the organization.

Lack of communication: - there is communication gap between the storekeeper and Property manager. Due to this problem the Administrators loss their time as well as they doesn't get the service in a good way.

Store keepers may not know expired date of some equipment. Store keepers didn't manage status of equipment to know whether it is broken or not.

The time it takes for head offices such as small offices to order equipment manually is very boring and it also makes overload of work on administrators and head offices.

2.3.2. Feasibility Analysis

2.3.2.1. Economic Feasibility

The organization will lose money and time since the system is manual. So if this proposed system is implemented, unwise use of resource as well as time is saved. In manual System store management, if the materials that are expired in a year from the whole stores are taken out, most establishment lose around 1million francs CFA in a year. From this, 30% is lose because of unknown materials expire date, 50% lose by misuse of materials and 20% lose because of materials order in their inappropriate place. And also when I calculate the time it take for the manual record of received material stock book and give materials for requested on outgoing book it takes seven day for one person to complete but in the proposed system it will take around 1 hours and minimize around 5000frs CFA paid for one person in a month that are used for papers, pen and salary.

2.3.2.2 Technical Feasibility

Talking of expecting the system to be technically feasible means that the store keepers, store manager and offices worker uses the system simply because the language I choose is familiar with them and the requirements for the proposed system is available in the organization. These resources are Network cables with RJ-45, Computers with minimum of 2GB RAM, 200GB Hard disk and since it is developed by the Object Oriented System Development technique it is simple for any user of organization. Finally I have the ability to develop this system without any difficulty since I have studied the required methodologies and tools. So the system will be technically feasible.

2.3.2.3 Time Feasibility

To develop the whole proposed system I need around total of 4months duration.

- One months for Analysis and Design
- Three months for Implementation; the details of these two categories can include the following explanations.

How long will it take to get the technical skill?

I may have the technology, but that doesn't mean I have the skills required to properly apply that technology since I am taking the course of distributing system

in parallel with the design phase of this project to implement it in networking technology.

The real constraints on project deadlines

- Deliver a properly functioning information system within given time of period. Since delivering an error-prone, useless information system on time is useless.
- Missed schedules are bad, but inadequate systems are worse.

2.3.3 - Analysis

In this phase of the development, I was concerned with, “What exactly must the system do?”[4]

2.3.3.1 Non-functional Requirements and Constraints

Non-functional (supplementary) requirements pertain to other information needed to produce the correct system and are detailed separately.[5] Constraints on the services or functions offered by the system such as timing constraints, constraints on the development process, standards etc.

- **User interface**:- The system provides java user interfaces that are compatible with windows platform.
- **Hardware consideration**:-The organization should have computers having typical storage capacity and processing speed.
- **Error handling**: our system handles error by showing the message” invalid input” when the user enters invalid input.
- **Security**: the system should have a security privilege that secures the system. And also there must be a physical security that secures (especially) the server computer. That means the server computer is only allowed for the server admin.
- **Performance characteristics**: the end user computer should have medium processor and the server computer should have large processor. It’s measured by its speed of processor.
- **Physical environment**: The system needs good environment.
- **Back up**: The system should have back up using external hard disk. The backup is taken weekly.
- **Availability**-The system should be available all the time

2.3.3.2 Functional Requirements

Functional Requirements are those that refer to the functionality of the system [6], i.e. what services it will provide to the user. Statements of services the system should provide, how the system should react to particular inputs and how the system should behave in particular situations.

- The system would be able to register the received materials.
- The system would be able to register salvage resources.
- The system also send request from users i.e. from offices.
- The system would be able to control expire date of equipment, materials etc.
- The system would be able to search and update the material information when it is needed.
- The system would be able to notice for buying materials.
- Record items and materials when items and materials are received.
- Change Status of items and materials when items and materials are used.

- Receive request and reply for the request.
- The system should generate timely or year report about the allocation of resources.
- The system should store all the data related with all the tasks performed into a database.

2.3.3.3 - Product Requirements Specification

Product requirements here are described in terms of the main deliverable. That is, the physical check list of the system's components and their respective attributes. Product Requirements for this project is as follows.

- Graphical User Interface
- A database
- A printer

2.3.3.4 - Process Requirements Specification

Process requirements describe how users interact with the system and how the system interacts with other systems. Process requirements for the "Store management system"

- The system must be able to save, update and retrieve goods records from a database. The system must be able to display results and information to a user on a proper graphical user interface.
- The system must be able to generate and print monthly and yearly reports of goods received and given out.

2.3.3.5- Logical Model

The Product and Process Requirements specifications were used to create the logical model of the system. The logical model was created using Data Flow Diagrams, and Entity Relationship Diagram.

2.3.3.5.1 - Data Flow Diagram (DFD)

The DFD in figure 1, summaries the sources and destinations of data and the processing that the data passes through in the system. It shows how data is transformed into information as it moves through the system.

Figure 1 shows the Data flow Diagram for the Store management System.



Figure 1 – A Simplified Data Flow Diagram

2.3.3.5.2 - Entity Relationship Diagram (E-R Diagram)

An E-R diagram is a presentable way of identifying and summarizing the entities of a system to be developed. The entities are usually the things the project or system will store data about. The diagram helps to show graphically the relationships between such entities, which could be one-to-one, one-to-many or many-to-many. The Store Management System has the following entities: Category, Goods, Store, Office, Users, Supplier and roles. The E-R diagram in figure 2 shows the relationship between them.

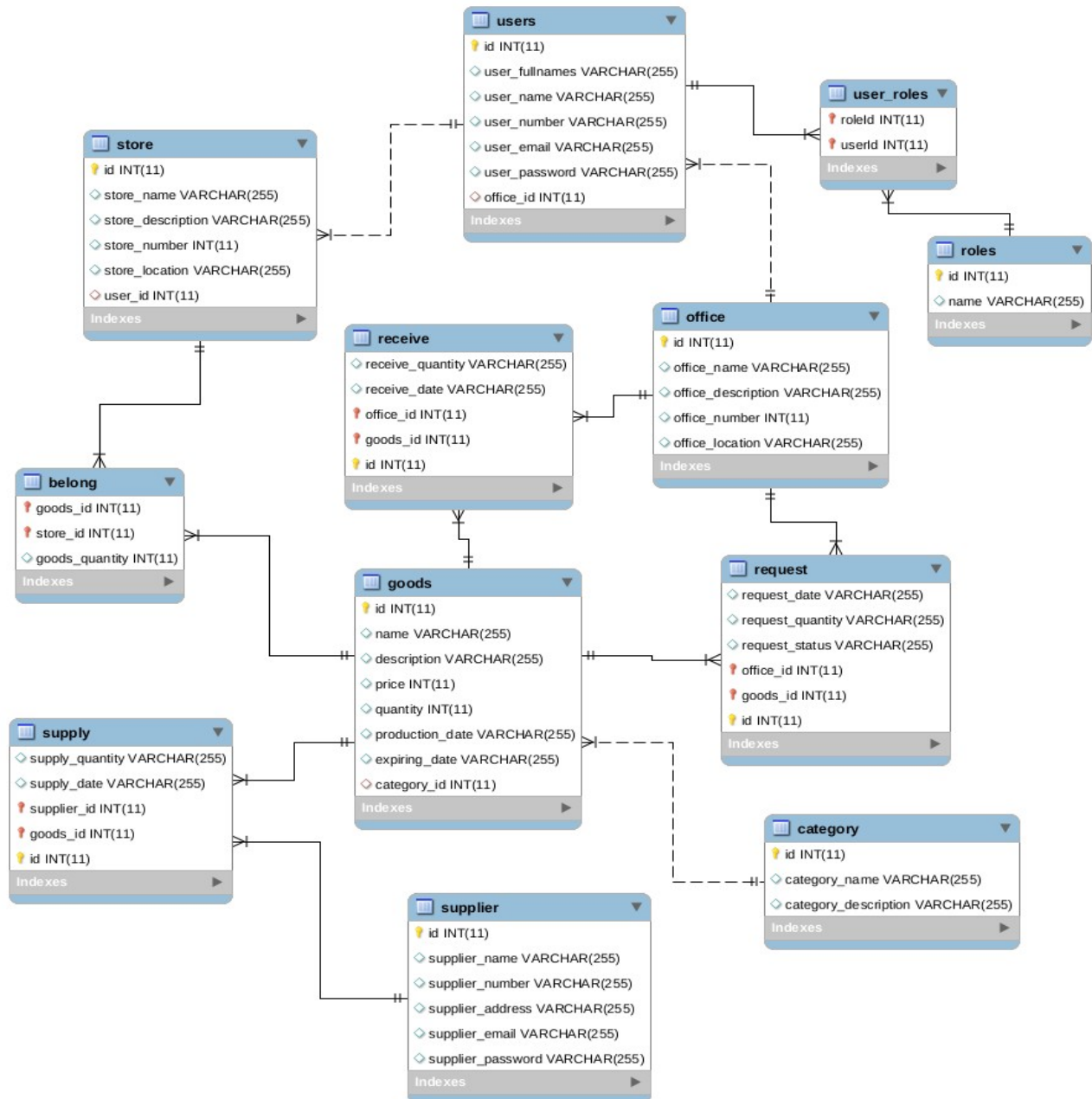


Figure 2 - An entity relationship diagram for a Store Management System

2.3.4 - Design

Design is process of describing, organizing, and structuring system components at architectural design level (High level design) and detailed design level (Low level design). Design converts functional models from analysis into models that represent the solution. Design may use structured or Object oriented approaches.

2.3.4.1 Purpose and Goals of Design

The design part is very important so as to make the implementation very easy. The different types of the system modeling techniques that are used for the implementation of the system such as deployment and component modeling are show in detail. Not only the system modeling techniques but also some system design techniques such as system decomposition design are cover in detail in this phase. The non-functional requirement is the description of the feature characters and attributes of the system. Some of the design goals are:-

Security- The system is secured that unauthorized user cannot access the data that does not concern with them as well as ability to withstand malicious attacks.

Reliability- The system has the ability to perform its required functions under stated conditions.

Fault Tolerance-The system should be able to give response (Error Message) when the user enter incorrect input. This recommends the user to enter correct input.

Throughput: It is able to perform many tasks in fixed period of time. Different service center do different tasks in their working time without worrying the other service center are using the same system.

Robustness: The system has the ability to survive wrong user's inputs it didn't accept it instead it show error message and continue to enter correct input.

Modifiable: If there is any change to the system or fault it can be modified easily.

Usability: Store Management System provide easy user friendly interface for users of the systems. It also provides help menu which gives brief description how to use the system so that user can be able to use it easily.

Memory: Store Management System requires the following space to run the system. Desktop or laptop computers and web server computers having more than 1GB of RAM and high storage capacity and processing speed.

This phase of the development was to translate the Logical model into a Physical model. This was done in two levels: High Level Design and Low Level Design.

3.3.4.2 - High Level Design

The key question I asked here was, how, in general, is the problem to be solved?

The result of this stage is the production of an outline solution consisting of components of the system, and user interfaces. The system was broken down into three basic stages: input, process and output. A UML diagram and a flowchart were used to conceive a High Level Design for the system.

2.3.4.2.1 UML Diagram

Use case diagram describes the behavior of the system as seen from an actor's point view. A use case describes a function provided by the system as a set of events that yield a visible result for the actors. In the analysis phase they represent

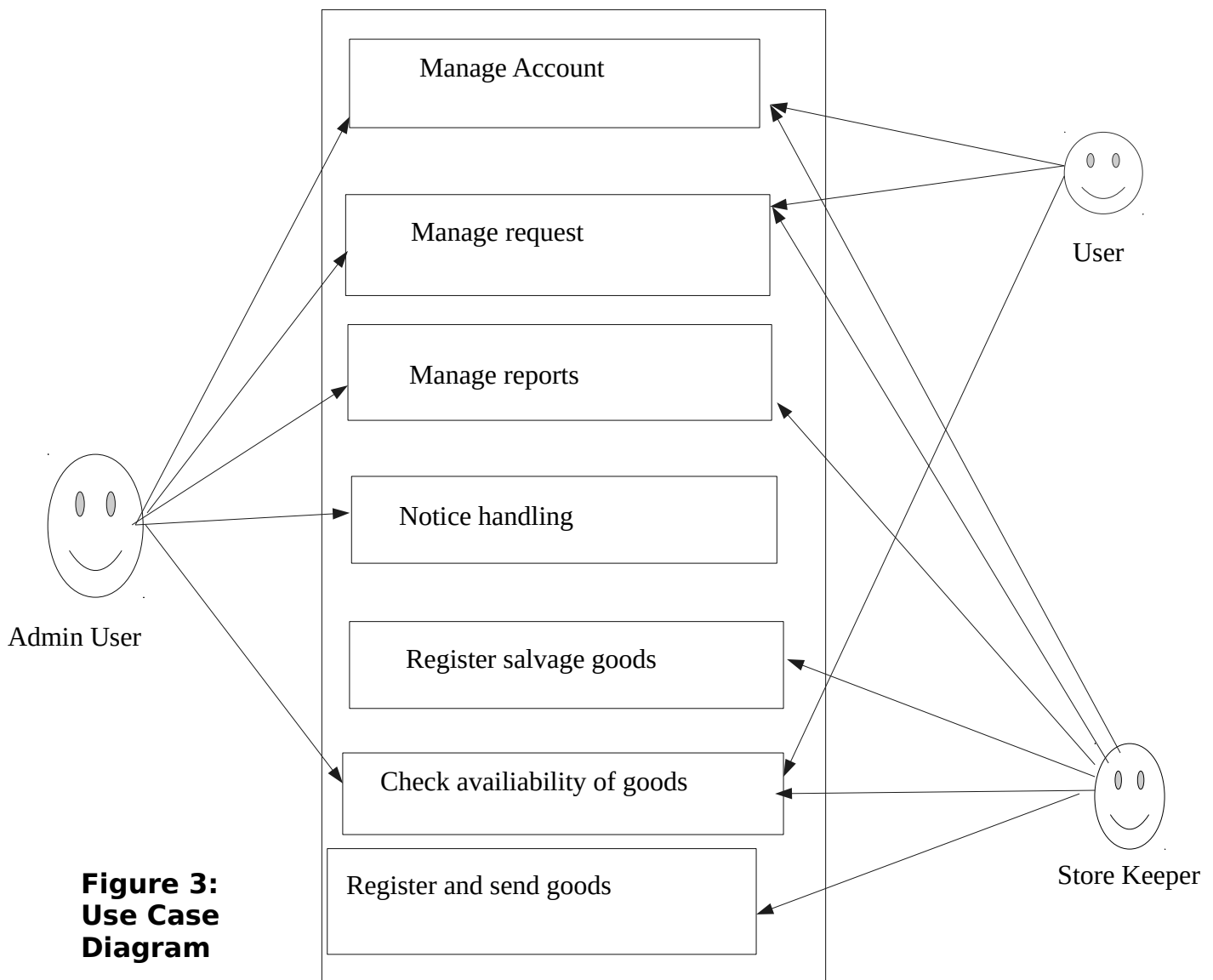
the functionality of the system. Admin is the store manager and users are office workers and store keeper.

2.3.4.1 - High Level Design

The key question I asked here was, how, in general, is the problem to be solved? The result of this stage is the production of an outline solution consisting of components of the system, and user interfaces. The system was broken down into three basic stages: input, process and output. A UML diagram and a flowchart were used to conceive a High Level Design for the system.

2.3.4.1.1 UML Diagram

Use case diagram describes the behavior of the system as seen from an actor's point view. A use case describes a function provided by the system as a set of events that yield a visible result for the actors. In the analysis phase they represent the functionality of the system. Admin is the store manager and users are office workers and store keeper.



**Figure 3:
Use Case
Diagram**

2.3.3.6.2 - System Flowchart

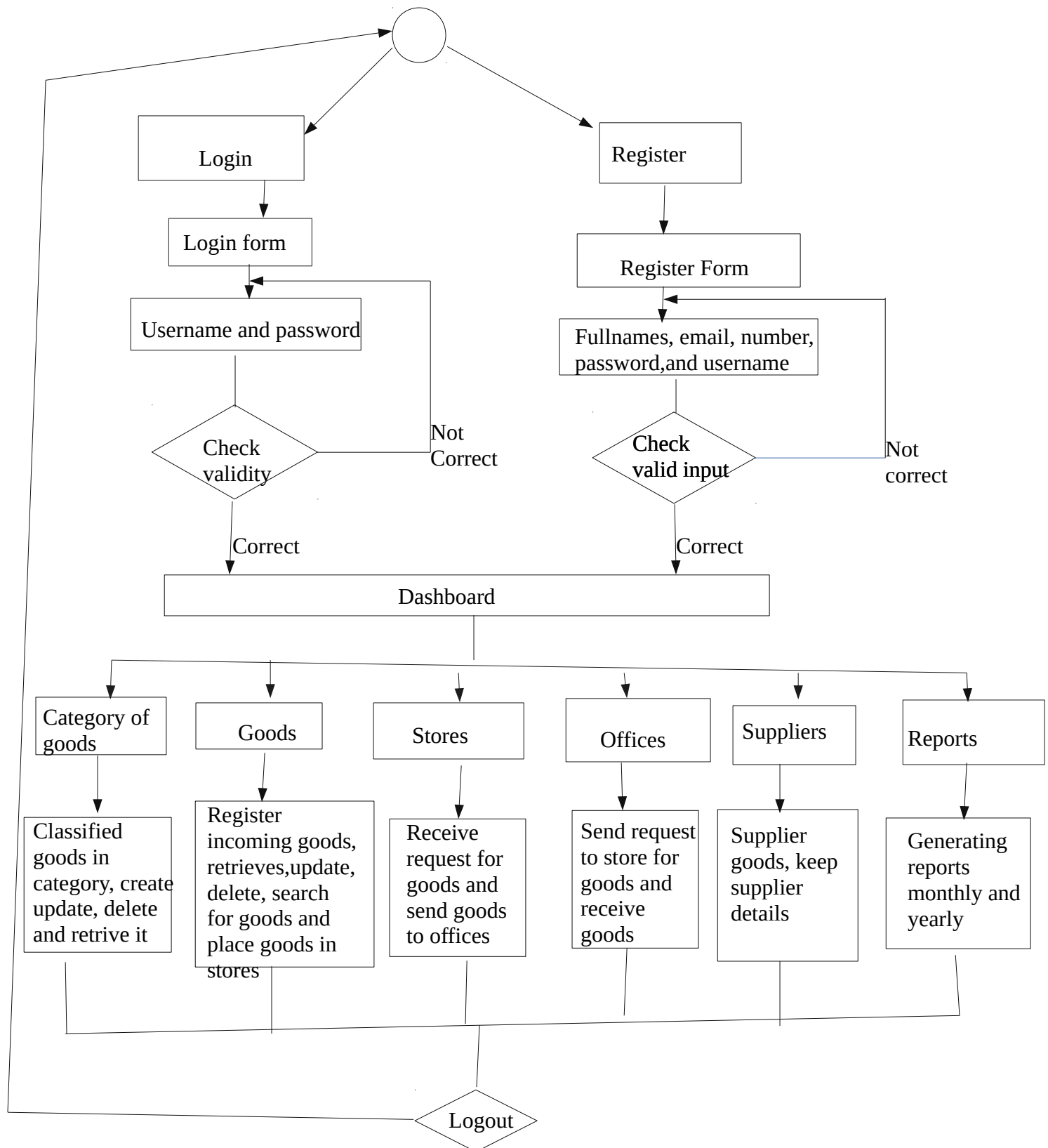


Figure 4: Flowchart Diagram

Generally, when the system is launched or opened, the user is presented with a Login screen. At this moment, if the user changes his/her mind, then he/she can simply close the form and exit the system. If on the other hand, the user chooses to proceed, he/she will have to supply a user name and a password for the authentication process. If the user passes this process, then the main menu interface opens and displays the various menu options available in the system. If the authentication process fails, then the user is struck with the login interface. From the main menu, the user can select options such as Goods, category, stores, supplier, Settings, etc. Whenever the user is done with a menu option, he/she can either exit the system or move to another option.

2.3.4.1.2 - User Interface Design

The system’s inputs and outputs will be through user interfaces. I decided to use a Graphic User Interface in order to make the system friendly. Below are figures of prototypes of the project. **Not All are listed here.**

Login

Username

Password

Login

Figure 5: Login Form

X

Logo

Login user name

Dashboard

All category

All goods

Stores

Offices

Footer

Dashboard

Register Form

Fullnames

Tel

Email

Username

Password

Sign-in

Figure 6: Register Form

Figure 7:Menu dashboard

2.3.5 - Testing

The coding involves translating the Low Level Design into program codes. This was done using Angular Framework and Express Node Js.

2.3.5.1 - Testing

2.3.5.1.1 - Unit Testing

One is always tempted to think that the purpose of testing is to show that the program works correctly. However, my main aim of testing here is to discover any errors that might be present in program. Testing does not prove the correctness of the program but it can reveal the existence of errors []. According to Edsg Dijkstra, "Program testing can be used to show the presence of bugs[], but never to show their absence." It is impossible to test any application exhaustively, but you must set about it systematically so that as many potential problems as possible can be located and put right. When all the parts of the system have been tested, the system as a whole also needs to be tested, to ensure the integration of the parts has been successful. In order to carryout unit testing, I decided to use two approaches. Dry-run testing, and Top-down testing.

I) Dry-run testing

Dry running is sometimes called walk-through or desk checking. During this testing phase, I read the codes thoroughly and checked them by hand. I also used a trace table to check the values of variables at every stage of execution. This desk checking helped me show so many hidden flaws in the Low Level design of the system.

II) Top-down testing

In this testing strategy, I started off by testing the user interface and the various components on it. From there, I walked my way down to the code operating behind the scene. This method of testing helped me to identified flaws in the navigation between the various user interfaces.

3.3.5.2.2 - Integration Testing

Integration testing takes place when the various modules of a system are put together. This test was performed to verify if modules that were separately developed would work together as planned, without error. I did integration testing incrementally. At first two modules (Main Menu and Goods functionality) were combined and tested together. This test showed no errors and actually performed as planned. The other modules were gradually added and tested in like manner until all the fifteen modules were integrated with each other functioning as single composite entity.

2.3.5.2.3 - System Testing

After performing integration testing, one needed to test the entire system to check it is working correctly. This was done with reference to the original specifications as outlined in the Low Level and High Level Designs. For this to be done one had to develop a test plan as shown below.

Test Series	Purpose of test series
1	To determine if the Menu Options do lead to correct form, or report

	when the user clicks on them?
2	To determine if validation of input data is performed correctly.
3	Iterations and decisions performed correctly. Calculations, searches and sorting performed correctly.
4	To determine if data is saved into the correct records in the correct tables.
5	To determine if the system produces the required results as per specification.

Table 1:Entire System Test Plan

2.3.6 - Implementation

This means installing the software on user machines. At times, software needs post-installation configurations at user end. Software is tested for portability and adaptability and integration related issues are solved during implementation. In this project the application is deployed in the implementation phase.

2.3.7 Deployment Diagram

It shows how the system is deployed on computers. In other words, it shows which component of the software is installed on which machine and how they communicate with each other if they are on different machines.

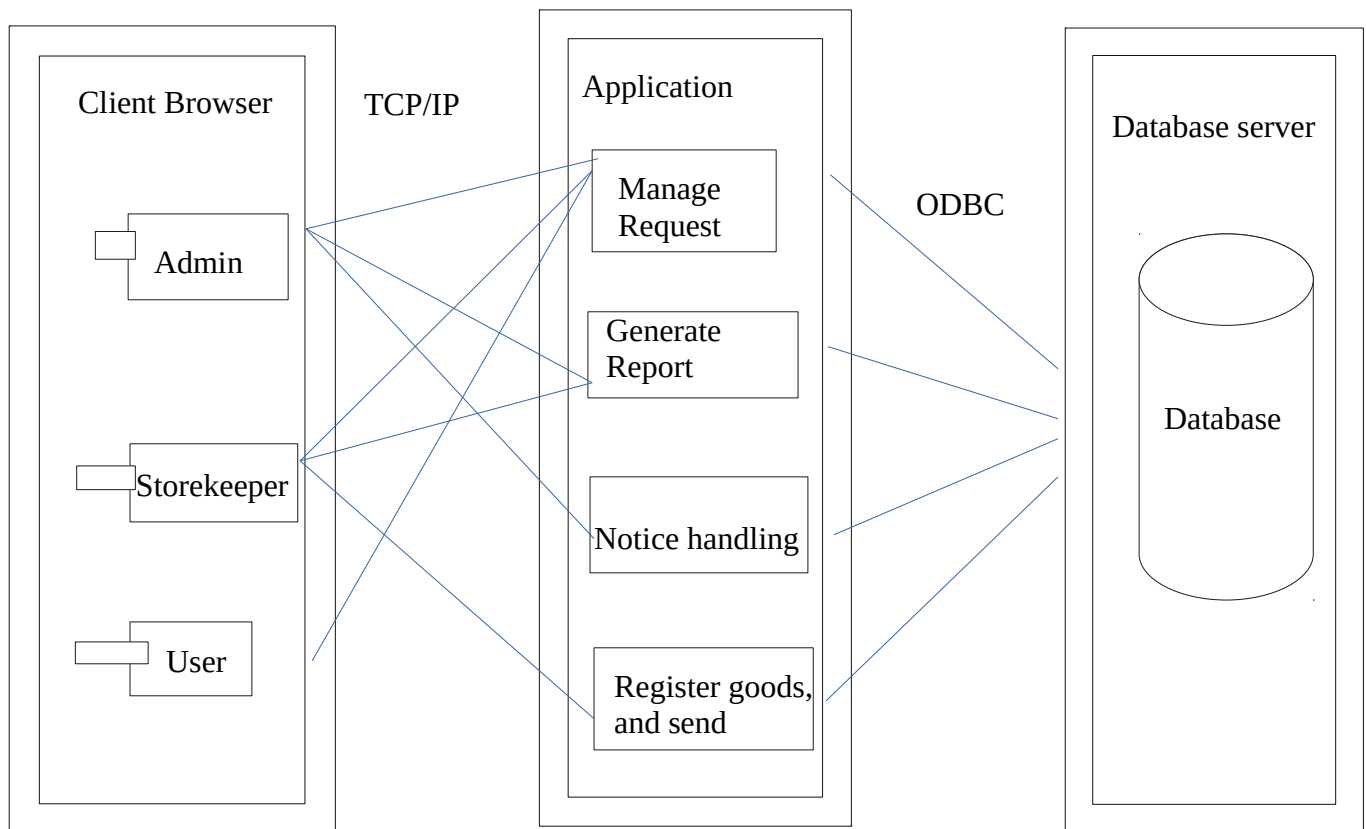


Figure 8: Deployment Diagram

CHAPTER THREE RESULTS AND DISCUSSION

3.1 Introduction

The core of these project is result and discussion. It is important because it an evaluation of the project in terms of implementation. The focus here is what was achieved and equally what was not. It examines how quality everything was achieved. The database code and interfaces screenshot of implemented modules are presented as well under here. The source code is too long that cannot be included here. The link to the source code will be given in the reference section of this report.

3.2 Results

Below are figures of interfaces of the project screenshot.

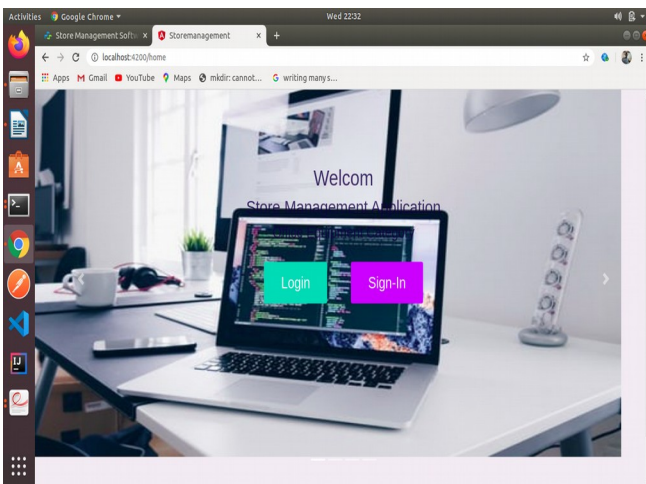


Figure 9: Opening page of the application

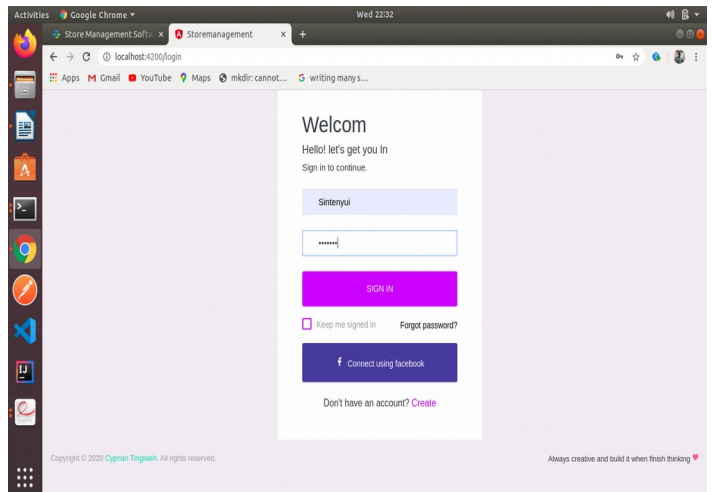


Figure 10: Login Form interfaces

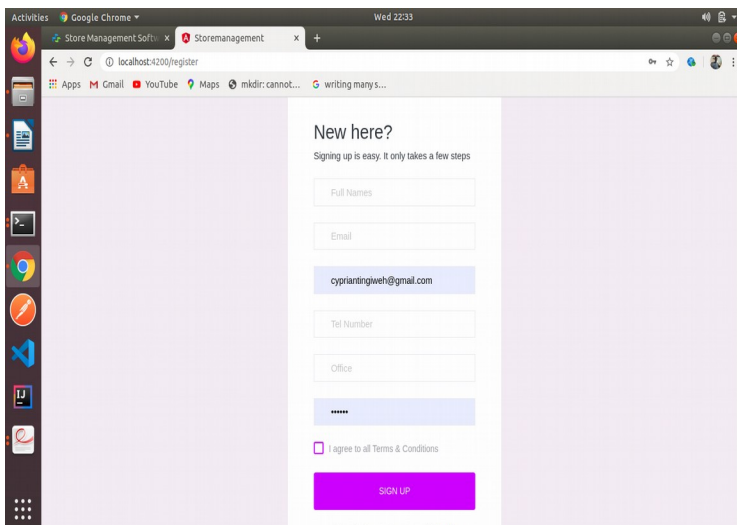


Figure 11: Sign-in form interface

Figure 12:Retrieve category of goods

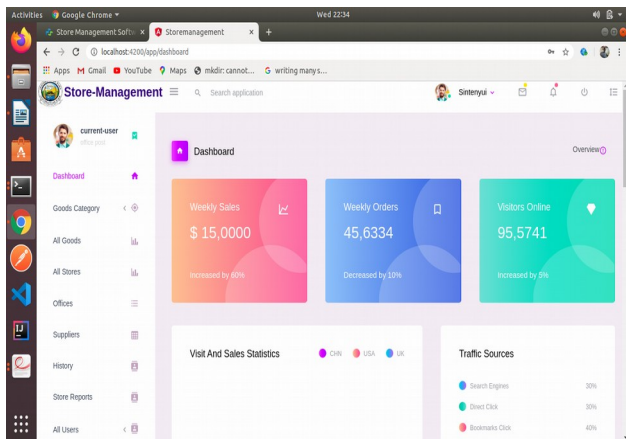


Figure 13: Dashboard intereface

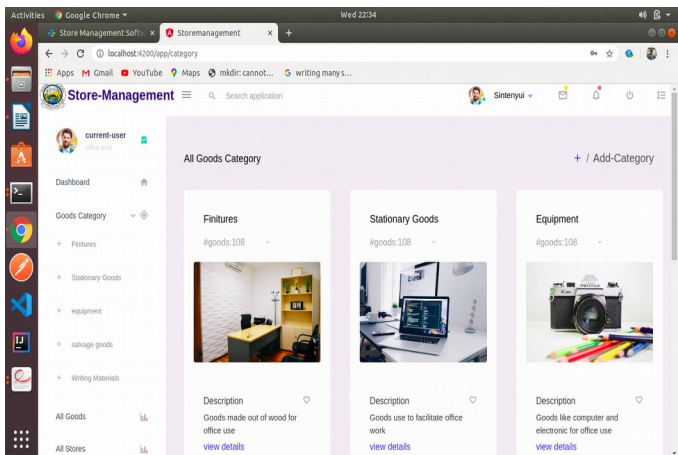


Figure 14:Retrieve category of goods

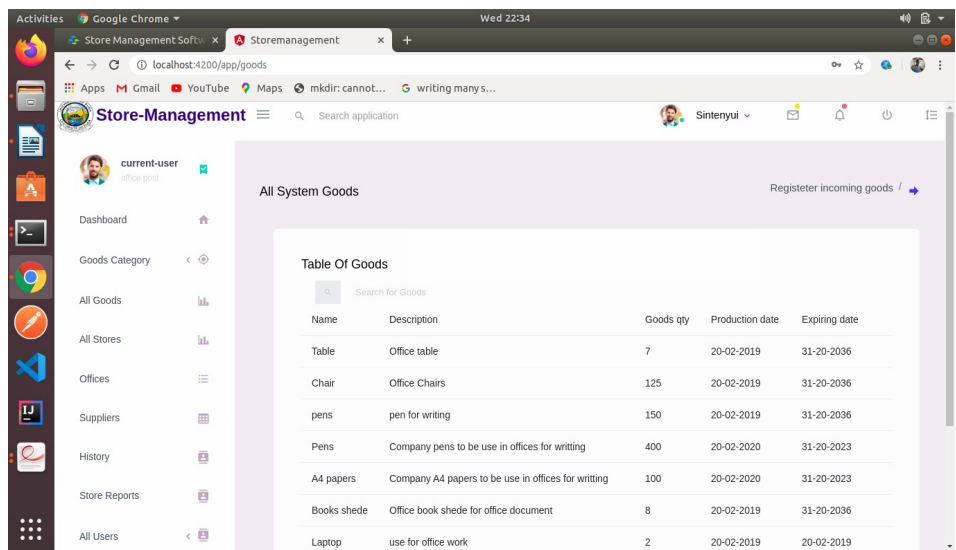


Figure 15: All goods table in the system

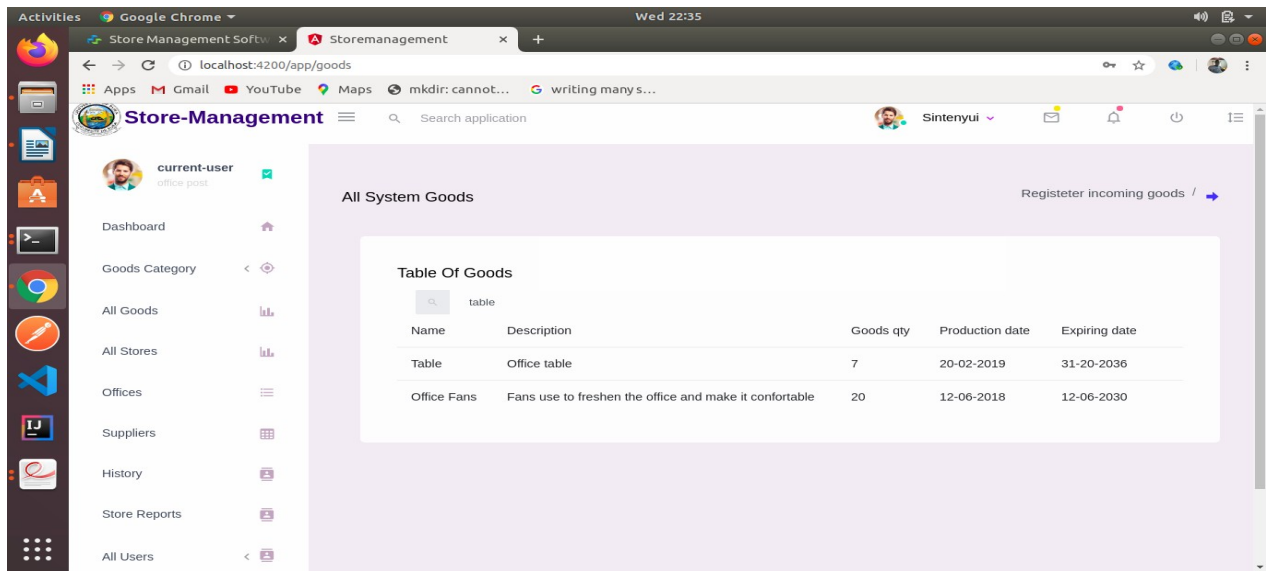


Figure 16:Active search of goods in the system

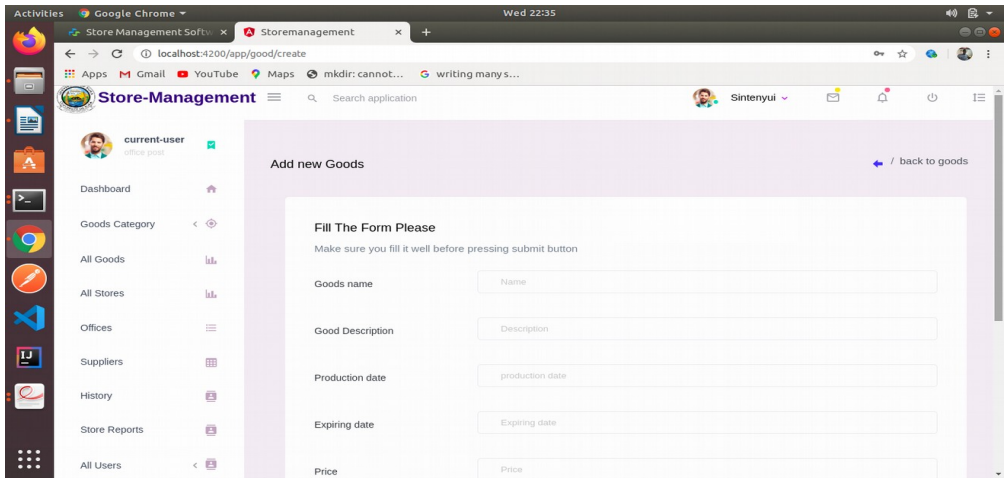


Figure 17: register incoming goods to the organisation

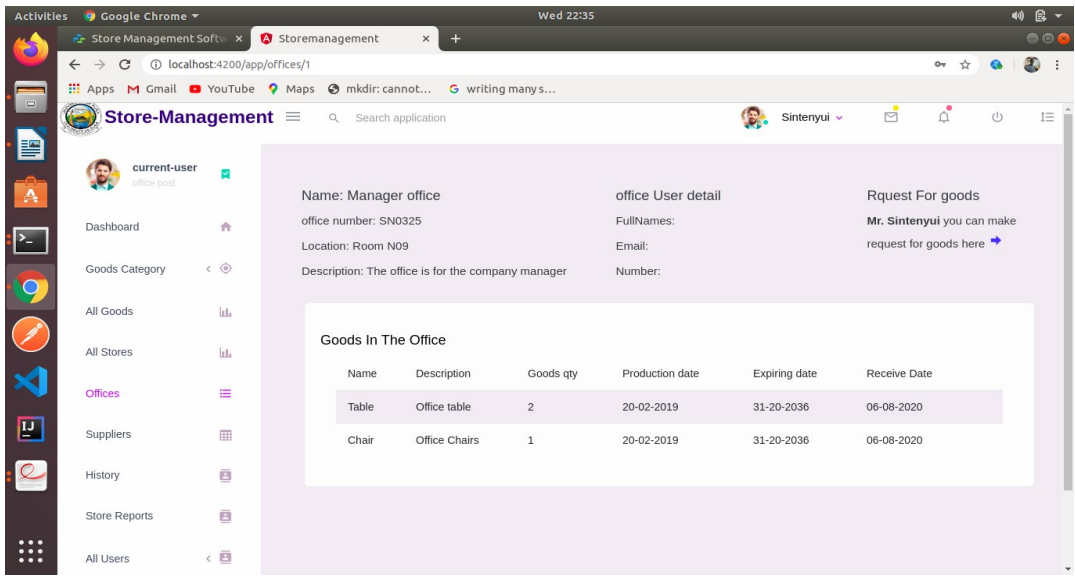


Figure 18: Example of Office content with goods and office user with request button to request for goods from store

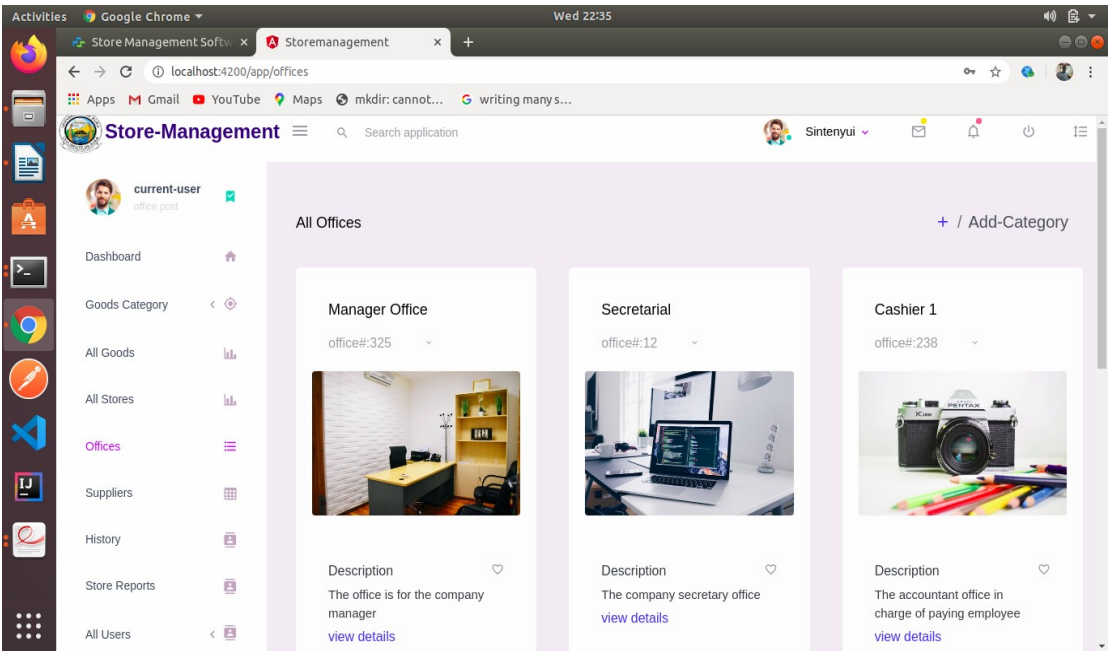


Figure 17: All offices in the organization.

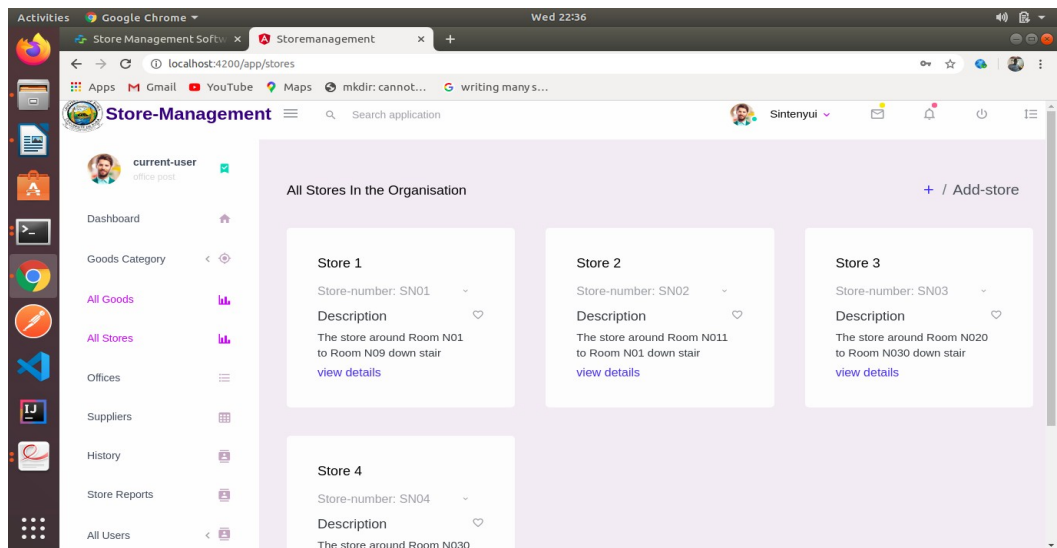


Figure 18: All stores in the system

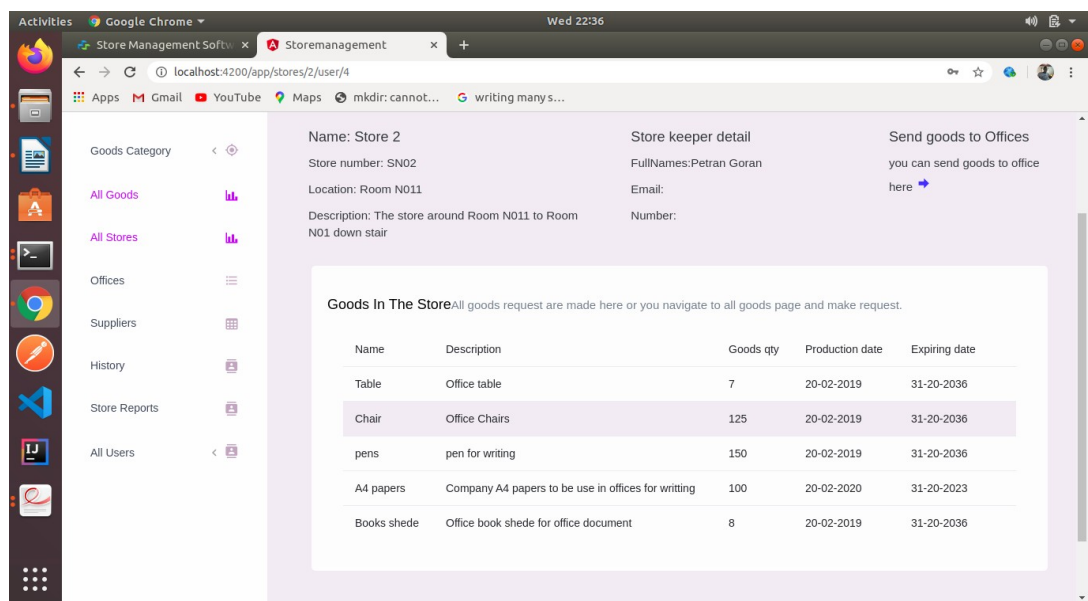


Figure 19: A store and all goods in a store with a send button

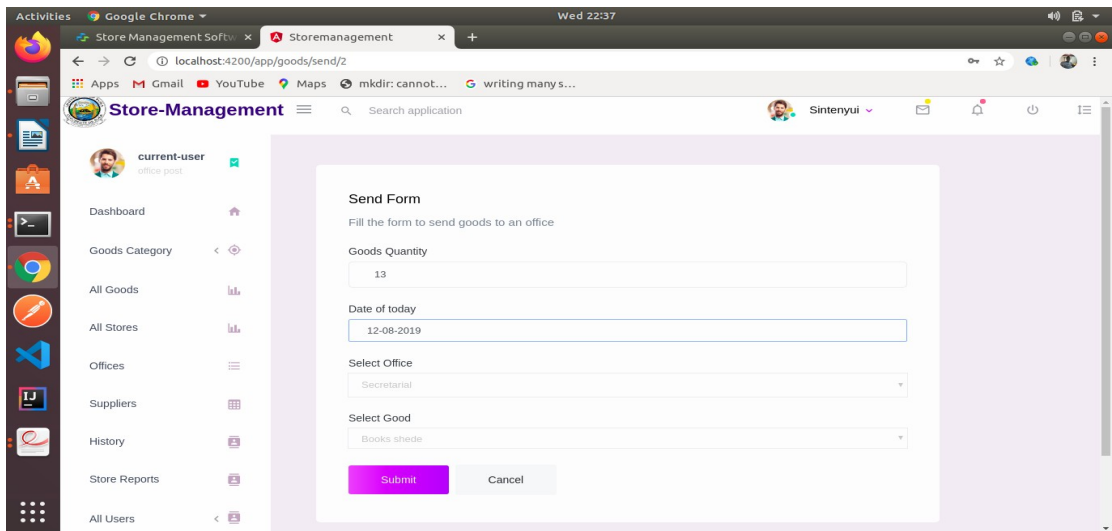


Figure 20: Send goods form to use and send goods to offices

3.3 -Discussion

Software development involves, intellectual, time, and material investments. When a project is said to be realized it simply means, the project's specifications have been met. In this project, not all the specification requirements have been reached.

3.3.1. Realized specification

1. Registration of incoming goods to the system.
2. Searching of goods to verified their availability in the store.
3. Sending of goods from store to offices and updating the system balance goods.
4. Sending of requests from offices to store for goods.
5. Approving of requested goods so they can be send to offices.
6. Security and user access control met.
7. Creating, retrieving, editing and deleting of system data in database is completed.

3.3.2. Unrealized specification

1. generation of monthly and yearly store report left out.
2. Notice handling and alert of goods expiring date/status in the store is left out.

In order to ensure that the system is accepted by a client in need of a Store Management System, I have ensure that the system possess high degree of effectiveness, learnability, usability, readability, navigability and maintainability.- --

Effectiveness

Though the project's specifications have not all been met, one still has to know how well the system solves the problem it was designed to solve. Here, I consider the speed it takes the system to load, register goods, send request etc. The testing shows that the system takes about three to five seconds to load fully on a single core machine with Clock Speed 1.3 GHz.

- Learnability

Ideally, an application should not have a learning curve, unfortunately all applications do. If the learning curve is very stiff, then the system likely to be rejected by the client. In this project, the learning curve is smooth and gentle. A user will need less than 20minutes to master the system. Just one encounter with the system is enough.

- Usability

Usability is a measure of how easy is it to use a system. The system developed in this project has high degree of usability. One does not need any special training to be able to use it. Just a few minutes with the system is enough to get the required experience necessary to use it.

- Readability

A lot of metaphors has been in the design of the user interface in order to render it readable. Metaphors used include icons and some basic animations. I make good use of this in the project.

- Navigability

This system has a high degree of navigability. Users can easily find their way as they navigate through the various interfaces. The user interface has been designed

in such a way that users do not get lost or stuck on any interface. The hints and feedback generated when a user performs an illegal are crafted such that they enable the user to know what is expected. Example is visited link appear purple.

- Maintainability.

In this project, the developed system can be reversed engineered. Thus, even after compilation, system can still be adjusted to remover errors. This system can be adjusted any time. And thus has a high level of maintainability.

CHAPTER FOUR

CONCLUSION

It has been a great pleasure for me to work on this exciting and challenging project. This project proved good for me as it provided practical knowledge on programming in Type script and JavaScript in Angular and express node js applications and MySQL Server. It also enabled me to gain more insight in developing Web application. My future plan is to make more features enriched software that can fulfill the users demand.

Recommendations

This project like any other, cannot be claimed to be 100% perfect or free from any bugs on my opinion I gauge it 80% success. I strongly propose the following recommendations to anyone who is interested in improving this project in future.

- Generate monthly and yearly report of stores.
- Send notification to offices and users and give alert of goods expiring date/status
- Convert the application to Desktop application.
- Develop Android version of the system to permit it run on mobile devices.
- Make the application multilingual so that non English users can easily use it.

REFERENCES

1. <https://en.wikipedia.org/wiki/Computer>
2. <https://www.iitms.co.in/higher-education-erp/purchase-and-stores/#:~:text=Store%20management%20system%20automates%20goods,It%20generates%20various%20store%20reports.>
3. Lectures notes on software engineering by Dr Joan Wacka.(page 10. link = <https://classroom.google.com/u/0/w/NzcwMTc2NDU0OTJa/t/all>)
4. Lectures notes on software engineering by Dr Joan Wacka.(page 8. link= <https://classroom.google.com/u/0/w/NzcwMTc2NDU0OTJa/t/all>).
5. Lectures notes on software engineering by Dr Joan Wacka.(page 22. link= <https://classroom.google.com/u/0/w/NzcwMTc2NDU0OTJa/t/all>)
6. Lectures notes on software engineering by Dr Joan Wacka.(page 23. link= <https://classroom.google.com/u/0/w/NzcwMTc2NDU0OTJa/t/all>).
7. Scott W.Ambler, The object primer Agile Model-Driven Development with UML, Cambridge University, ronin international, inc, 3 rd edition, 2004
8. http://www.dba-oracle.com/t_object_database_model.htm (Oracle Tips by BurlesonConsulting) Assessed on 12/11/2017 G.c
9. <http://www.au.edu.et>. Assessed on 20/11/2017 G.c
10. <http://www.agilemodeling.com/artifacts/deploymentDiagram.htm> Assessed on 20/11/2017 G.c
11. <http://www.agilemodeling.com/artifacts/classDiagram.htm> Assessed on 18/12/2017 G.c