SALES AND INVENTORY MANAGEMENT SYSTEM

K. A. D. K. S. PERERA

June 2023

SALES AND INVENTORY MANAGEMENT SYSTEM FOR CODE TECHNOLOGIES

K. A. D. K. S. PERERA

R191028

1910282

Name of the Supervisor:

MRS. W. I. P. SOYSA

June 2023

This dissertation is submitted in partial fulfillment of the requirement of the Degree of Bachelor of Information Technology of the University of Colombo School of Computing.

Declaration

"I certify that this dissertation does not incorporate, without acknowledgement, any material previously submitted for a degree or diploma in any university and to the best of my knowledge and belief, it does not contain any material previously published or written by another person or myself except where due reference is made in the text. I also hereby give consent for my dissertation, if accepted, to be made available for photocopying and for interlibrary loans, and for the title and abstract to be made available to outside organizations.

Signature of Candidate: ..... Date 6/2/2023

Name of Candidate: K. A. D. K. S. Perera

Countersigned by:

Signature of Supervisor(s)/Advisor(s): ..... Date 6/2/2023

Name(s) of Supervisor(s)/Advisor(s): Mrs. W. I. P. Soysa

Abstract

“CODE Technologies” is a reputed computer shop in Kegalle district, where serves a great service to all over the country. There goods are in superb condition. They sell brand-new items and used computer accessories. So, most schools, institutes and people tend to buy their computer items in trust. In the last 8 years, they sell goods and after that they serve a good service to their customers. As others, also I bought some computer accessories from that shop. However, they have been having trouble for a long time to adequately carry out their day-to-day activities due to the manual system they are now using. These manual activities that might have been automated ensured that staff spent a significant amount of time doing them rather than concentrating on other vital duties. They fail to make crucial decisions impacting their firm because they lack proper knowledge regarding their organization's performance. These were the most significant annoyances for all stakeholders involved in the business.

The necessity for a sales and inventory management system with automated operations was obvious for CODE Technologies, thus I proposed a solution for them. Prior to the construction of the program, the scope of what should be focused on was established. The above-mentioned scope included managing item categories, item management, suppliers management, purchases management, report production, managing customers, sales management, and invoice controlling. Afterwards, the vital attributes of the system were pinpointed. These encompassed inventory management, the generation of productivity and staff reports, order handling, stock returns, invoicing, and account administration.

The system was built with using languages and techniques such as HTML, CSS, Bootstrap, jQuery, MySQL Workbench, AJAX, WAMPP Server, Figma tool and Photoshop. The project is already finished and has been well received by both the supervisor and the client.

The web-based sales and inventory management system had been tested by several users, including the client, at the end of the development phase. The designed solution was successfully delivered to the owner with the goal of streamlining their day-to-day business routines.

Acknowledgment

First and foremost, I want to express my deepest thanks to the BIT Coordinator as well as the Project Evaluation Examination Board of the University of Colombo School of Computing for providing me with this invaluable opportunity to pursue a globally renowned degree program.

I extend my gratitude to Mrs. W.I.P. Soysa, my supervisor and the Teacher in Charge at Pindeniya Central College. Her exceptional expertise, deep understanding, generous guidance, and unwavering support were instrumental in enabling me to engage in a project that greatly captivated my interest. Collaborating with her was truly a delightful experience.

Finally, I'd want to convey my heartfelt gratitude to Mr. Sameera Madusanka, the owner of Code Technologies, and his team for their unwavering support and dedication to assisting me in successfully completing my final year project.

Table of Contents

List of figures

List of tables

List of acronyms

AJAX	–Asynchronous JavaScript And XML
BIT	– Bachelor of Information Technology
CSS	–Cascading Style Sheet
ER	–Entity Relationship
GRN	–Good Receive Note
HTML	–Hyper Text Markup Language
IDE	–Integrated Development Environment
IT	– Information Technology
JS	–JavaScript
PDF	–Portable Document Format
PHP	–Hypertext Preprocessor
PO	–Purchase Order
RUP	–Rational Unified Process
SQL	–Structured Query Language
UCSC	– University of Colombo School of Computing
UI	–User Interfaces
UML	–Unified Modeling Language

Chapter 1 - Introduction

1.1 Motivation for the project

"Code Technologies" is a reputed computer shop, Kegalle district, where serves a great service to all over the country. There goods are in superb condition. They sell brand new items and used computer accessories. So, most schools, institutes and people tend to buy their computer items in trust. In the last 8 years, they sell goods and after that they serve a good service to their customers. As others, also I bought some computer accessories from that shop. Once I went, I heard bad news. That is a person asked for a used motherboard, but salesman said, they don't have, it mean out of stocks. But the customer before leaving his home, he has given a call to the shop and got a message that there are enough motherboards in their stocks. Employees details are not updated properly, and employee details are misplaced. Customers purchasing item bills are misplaced. For Warranty claim and for repairs inability to contact the customer for above conditions. Inability to update the supplier details and to check the supplier's condition; It means his blacklisted or trusted supplier. So, will have the warranty problems. Also, I faced the same problem. There is another problem is stock maintain and communication problems. Other thing is misplaced of supplier's Goods Receive Note copy details. When some schools, institutes are requesting the quotations, they don't know whether the stocks are sufficient or not. Some returned items have not marked as properly. Above all weak points are very trouble some questions for me.

So, I intended that automated sales and Inventory Management System is the best way instead of paper-based system. Lastly, I propose the Automated Sales and Inventory Management System, to grow up the business with efficient and accurately in the computer field.

1.2 Objective

The main objectives of this sales and inventory management system are given below.

- Maintain records of the employees' details.
- Maintain records of the supplier details.
- To minimize the effort and time of employees in the Computer Shop.
- To improve the security of private data and information (Privacy) about the operations of the Computer Shop.
- To keep all data and information in a centralized databased for ease of access for Sales and Inventory management.
- To improve the reporting facility by generating various kind of reports in the form of both textual and graphical, required by the management.
- To reduce and eliminate the drawbacks in the current paper based manual system.
- Maintain records of the customer details.
- To maintain records of the Quotation details.
- To maintain records of the Item details.
- Instead of Manual hand write Bills this system generates of automated receipts.
- Improve the accuracy inventory management process and stock management.
- Show the information and description of the Profiles Logins.
- All fields such as Profiles, Logins Fees are validated and does not take invalid value.
- To Maintain Branch detail in Kegalle Area
- Maintain daily records of Sales and Purchase details.
- To Maintain records of Goods Receive Note receipt.

1.3 Scope

The project's scope can be stated as follows:

- User Administration (Administrator, Cashier, Storekeeper, Technician)
    - Add new users.
    - Edit System user information.
    - Change user status
- Provide all the necessary analysis and reporting.
    - Generate new reports.
- Item Management
    - Add new Items.
    - Edit saved items.
    - Get full detailed list of items.
- Managing and controlling bills
    - Add new daily bills.
    - Get full detailed list of bills.
- Management of sales and purchase details of Computer shop.
- Provide an efficient way to system users' details.
- Monitoring and Management of activities of each module
- Provide an effortless way to manage inventory.
- Managing item categories and related prices
- Customer Detail Management
- Ability to manage all income and expenses details and make reports system generate.
- Supplier Detail Management
- Warranty Claim Detail Management
- Return Detail Management
- Item Purchase Detail management
- Sales Detail management
- Good Receive Note Detail Management
- Quotation Detail management
- Branch Details Management
- User Management

1.4 Structure of the dissertation

This dissertation included high-level descriptions of the proposed system. The dissertation structure is shown in the following chapters.

The project introduction shows an overall quick explanation of the planned computer sales and inventory management system employing an existing system, the scope of the project, and the objectives.

Chapter 02 - Analysis

The system deserves thorough testing before being introduced into the working environment. This chapter will address testing techniques, test cases and more. It will also demonstrate that suitable testing was carried out.

Chapter 03 - Design

The design should be completed when the requirements have been elicited. This chapter contains information on the design process, diagrams generated by us, such as use cases, classes, databases, and activities.

Chapter 04 - Implementation

This chapter will contain implementation specifics. It describes the major code and module structure, as well as the implementation environment, which comprises hardware and software.

Chapter 05 - Evaluation

Before deploying the system into the field, it should be thoroughly tested. This chapter will explain testing methodology, test cases, and other topics, as well as demonstrate that proper testing was performed.

Chapter 06 - Conclusion

With the evaluation, this chapter will bring the dissertation report to a close.

Chapter 2 – Analysis

The analysis step of the software development life cycle is crucial. In this phase, the data acquired in the previous phase is carefully evaluated to discover the client's business requirements. Once this is finished, the discovered requirements are classified as functional or non-functional.

2.1 Existing system

Code Technologies are now used in paper-based systems. All information concerning daily activities, sales, purchases, item management and employee management is done manually. for example, generate an invoice done by using a preprinted structured sheet.

2.2 Review of Similar Systems

To comprehend the business process, certain similar systems on the market were examined. Several of these comparable systems are given below.

2.2.1 Zoho Inventory Management System

figure 2.1 - depicted the screenshot of Zoho Inventory Management System. [1]

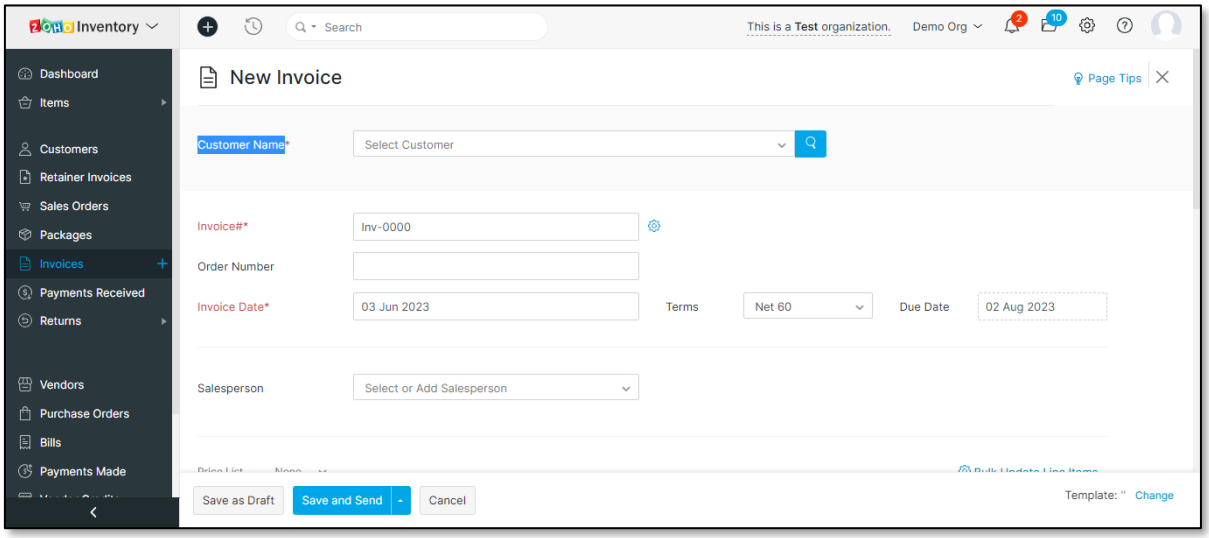


Figure 2.1: Screenshot of Zoho Inventory Management System

Zoho Inventory Management System features. [1]

Functional features

- POS System
- Centralized and Large Database
- Report Integration
- Return Management

2.2.2 Zap Inventory Management System

figure 2.2 - depicted the screenshot of Zap Inventory Management System. [2]

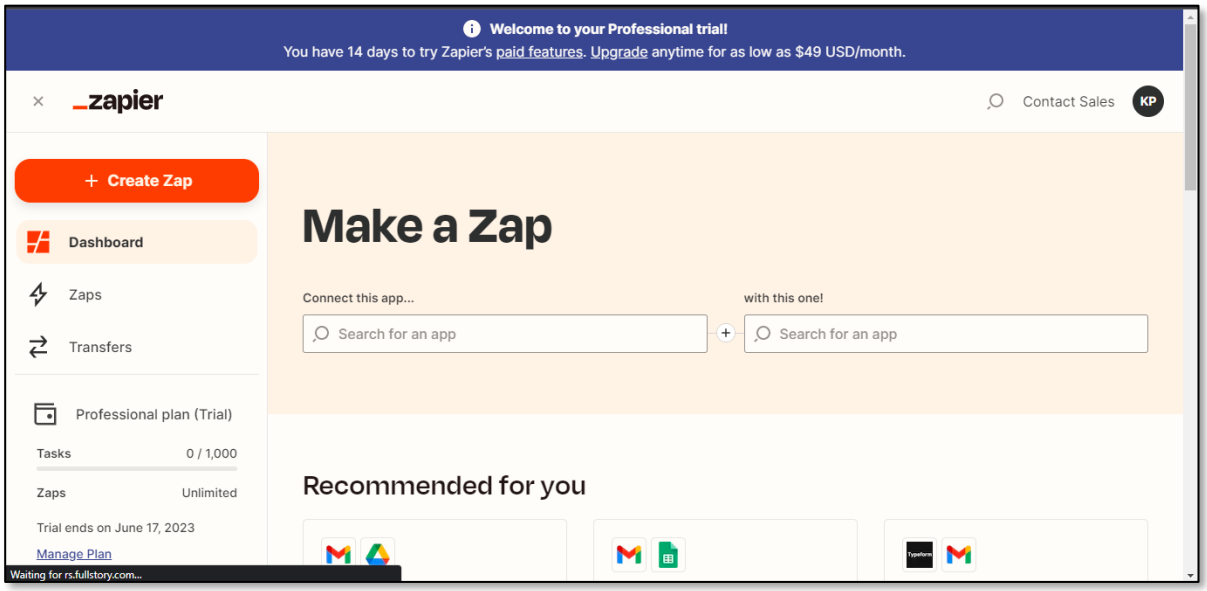


Figure 2.2: Screenshot of Zap Inventory Management System

Functional features of Zap Inventory Management given below [2]

- GRN Management
- Purchase Order Management
- Employee Management
- Client Management

2.2.3 Comparison of Systems

Table 2.1 shows a comparison between the studied existing similar systems and this web-based production and distribution management system.

Features	Zoho	Zap	Proposed System
Users Management	✓	✓	✓
GRN Management	✗	✓	✓
Order Management	✓	✓	✓
Employee Management	✓	✓	✓
Order Management	✗	✓	✓
POS System	✓	✗	✓
Return Management	✓	✗	✓

Table 2.1: Comparison of systems

The proposed web-based sales and inventory management system includes all of the features and capabilities needed for day-to-day operations.

Figure 2.3 shows a use case diagram of the existing manual system.

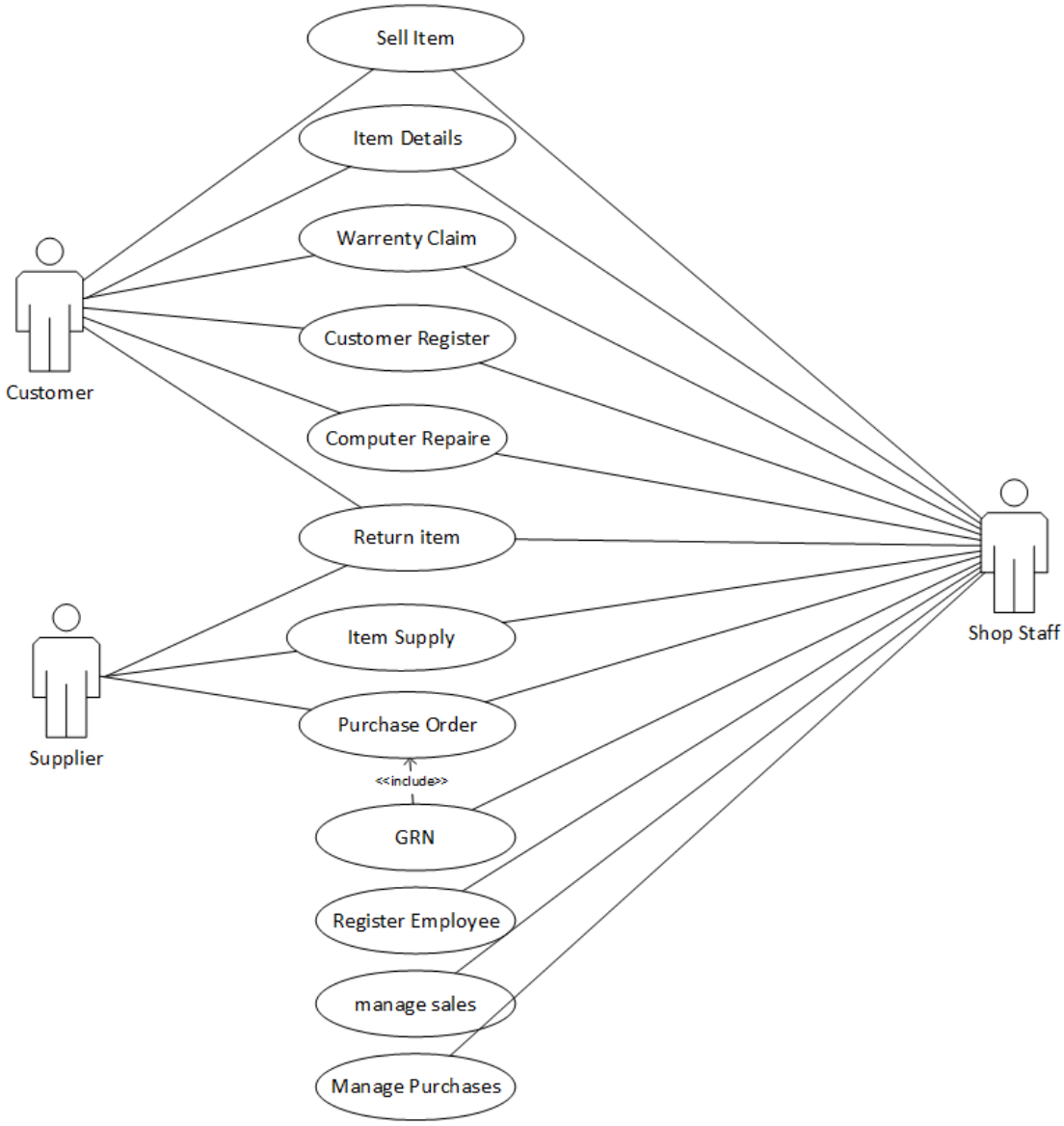


Figure 2.3: Existing System usecase diagram

2.3 Analysis of the requirements

The client's identified needs were divided into functional and non-functional requirements.

Customer Registration

In the current manual method, the customer's name, address, Contact, National Identification Card, and signature are added to the invoice. Customers are registered for both selling products and servicing computers. Customer information is not kept separately. In the billing book, carbon copies of invoices are stored. Invoices feature customer information.

Supplier Registration

Suppliers are registered by providing their name, address, NIC, phone number, and firm. Purchase orders are generated for approved vendors. Supplier information, like customer information, is not kept separately.

Employee Registration

Employees at this code technology firm are classified according to their job levels. Owner, Technician, Cashier, and Storekeeper are some examples. Employee information such as name, residence, recruit date, date of birth, NIC, designation, and gender are entered into the register.

Computer repair

This shop's mending area accepts computer repairs. Technicians examine the computer problem and then obtain the customer's phone number in order to refer that PC. After the computer has been repaired, an invoice with a repair description is issued to the client.

Populate Sales

Keep manual records of sales goods. Keep track of how many goods are issued and the amounts of that day's sales invoice to invoice under the date.

2.3.1 Functional requirements

Functional requirements are the characteristics and capabilities that the system must have in order to meet the client's expectations. The following needs have been determined as the Sales and Inventory Management System's functional requirements.

- Manage all stock categories and goods.
- Manage user administration and privileges.

- Keep track of all sales.
- Keep track of all purchases.
- Manage all the supplier and their information.
- Manage all the customers and their information.
- Manage all the product categories availability.
- Produce all required reports.
- Keep track of computer repair details.
- Keep track of return items details.
- Keep track of Order Management.

2.3.2 Non-functional requirements

Non-functional requirements encompass all the requisites beyond those classified as functional requirements. These non-functional requirements may not be essential for the system's basic functionality, but they serve as desirable improvements that elevate both user experience and the overall value of the system.

Enhance Security

Application security holds utmost significance and is approached from two distinct angles. Firstly, it involves safeguarding the application itself, while the second aspect pertains to securing the database. Upholding application-side security involves implementing user logins and data encryption methods. Notably, ensuring the security of payments and check management within the application holds paramount importance.

Particular significance are the payment handling and check management processes, which carry substantial weight in terms of data security. Emphasis is placed on fortifying the security of these processes. On the database server side, a robust security framework is maintained through the utilization of a database management system. This entails creating user logins, along with the assignment of roles and privileges to ensure heightened security.

Usability

The design of the system's user interface prioritizes simplicity to reduce complexity. This approach adheres to established user interface design principles. Additionally, intricate processes are broken down into more manageable components. Offering

diverse approaches to accomplish the same task enhances the system's user-friendliness. The incorporation of graphical icons, colors, messages, wizards, and menus holds significant importance as they expedite comprehension, ensure accuracy, and optimize the efficiency of the system. These methods collectively contribute to enhancing the usability and comprehensibility of the system.

Accuracy

Maintaining a high level of data accuracy is a critical consideration. To reduce the storage of inconsistent data within the database, rules and data validations are employed. To ensure accurate data entry, the system offers suggestions and assistance to users in various scenarios. Enhancing system accuracy involves utilizing appropriate data types and field sizes, while also minimizing the inclusion of null values and redundant data entries in the database.

2.4 Process model

A Rational Unified Process is a method for assigning tasks and duties inside a company that develops software. Its major goal is to enable the development of high-quality software that meets the needs of the end user within a reasonable price and timeline.

RUP is a method of allocating duties and duties within the development group that provides best practices and principles for efficient software development. As a result, it can provide high-quality software on schedule and under budget while meeting the needs of its clients. So, I decided to choose as a process model to develop my system.

Following figure 2. 4 visualizes the RUP model [3]

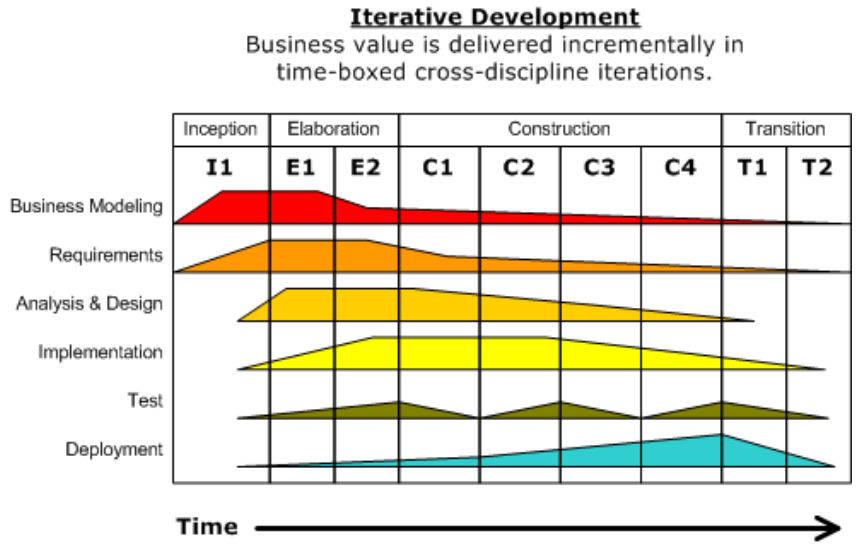


Figure 2.4: Rational Unified Process Model

For this computer sales and inventory management system, the Rational Unified Process Methodology was used. The Unified Modeling Language, also known as UML, was used to create the system and record the procedures. It is more suited for Object Oriented Methodology and reuse of components. RUP is compatible with a wide range of software development technologies. A development cycle is split into four stages. In the RUP process, each phase has a distinct goal and well-defined milestones.

- **Inception Phase** - Prepare a document during this phase. An early use-case model is being created. A preliminary risk assessment of the business concept is also completed. A project plan with phases and iterations is created.
- **Elaboration Phase** - At least 80% of a use-case model is completed in this phase. Additional needs were gathered. This phase includes the design and prototyping of software architecture. The entire project development plan has been completed.
- **Construction Phase** - The software package is installed on the appropriate platforms. In this RUP stage, supply user guides and descriptions of the current release.
- **Transition Phase** - Use beta testing to validate the new system. Parallel operations are carried out with the legacy system which it is substituting. This stage included the conversion of operational databases, as well as the training and maintenance of users.

Chapter 3 – Design

Software design is the process of creating software methods, functions, and objects, as well as the overall structure and interaction of the code so that the final functionality fulfills the demands of users.

3.1 Design strategies

The objective of this project is to create a web-based system that utilizes a MySQL database for data storage. The development process will adhere to MVC principles and primarily involve coding in HTML, JavaScript, AJX, JQUERY and PHP programming language. Additionally, the web application will be styled using the Bootstrap CSS framework to fulfill all the anticipated business requirements set by CODE Technologies.

3.1.1 System solutions alternatives

As potential alternatives for this proposed web-based application, we have identified the development of a native desktop application and the creation of Off-the-shelf System. The following are some alternatives to the web-based system.

3.1.2 Native Desktop application

Native apps are created for a certain software platform. These apps are built with specialized native frameworks supplied by those platforms. When compared to cross-platform software solutions, these applications perform better. These apps, however, cannot be run on platforms other than the one for which they were designed.

3.1.3 Off-the-shelf System

Off-the-shelf software is ready-made software that may be purchased in stores or downloaded from the internet. These systems are created with the mass market in mind, and they often have a large number of features that may be employed in a variety of use cases. As a result, several superfluous functions may be discovered in this software. Off the shelf software can be obtained at a lower cost than the development cost of bespoke software. In the long term, though, designing bespoke software is preferable.

3.1.4 Justification for Selected Solution

The primary arguments for selecting a web-based system over other options are stated below.

- A web-based system may be easily accessed through the internet by using a web browser. It is independent of the OS and may be used with any device such as a computer, tablet, or smartphone.
- There is no need to acquire any specialized hardware.
- If there's an internet connection, the system can be accessed from anywhere in the world.

3.2 Architectural design

A use case diagram is the most basic sort of software requirement for a freshly developed software program. Use cases define the intended system behavior but not the method to accomplish it.

3.2.1 Overall Use Case Diagram

Following figure 3. 1 visualizes the overall usecase diagram of the proposed system.

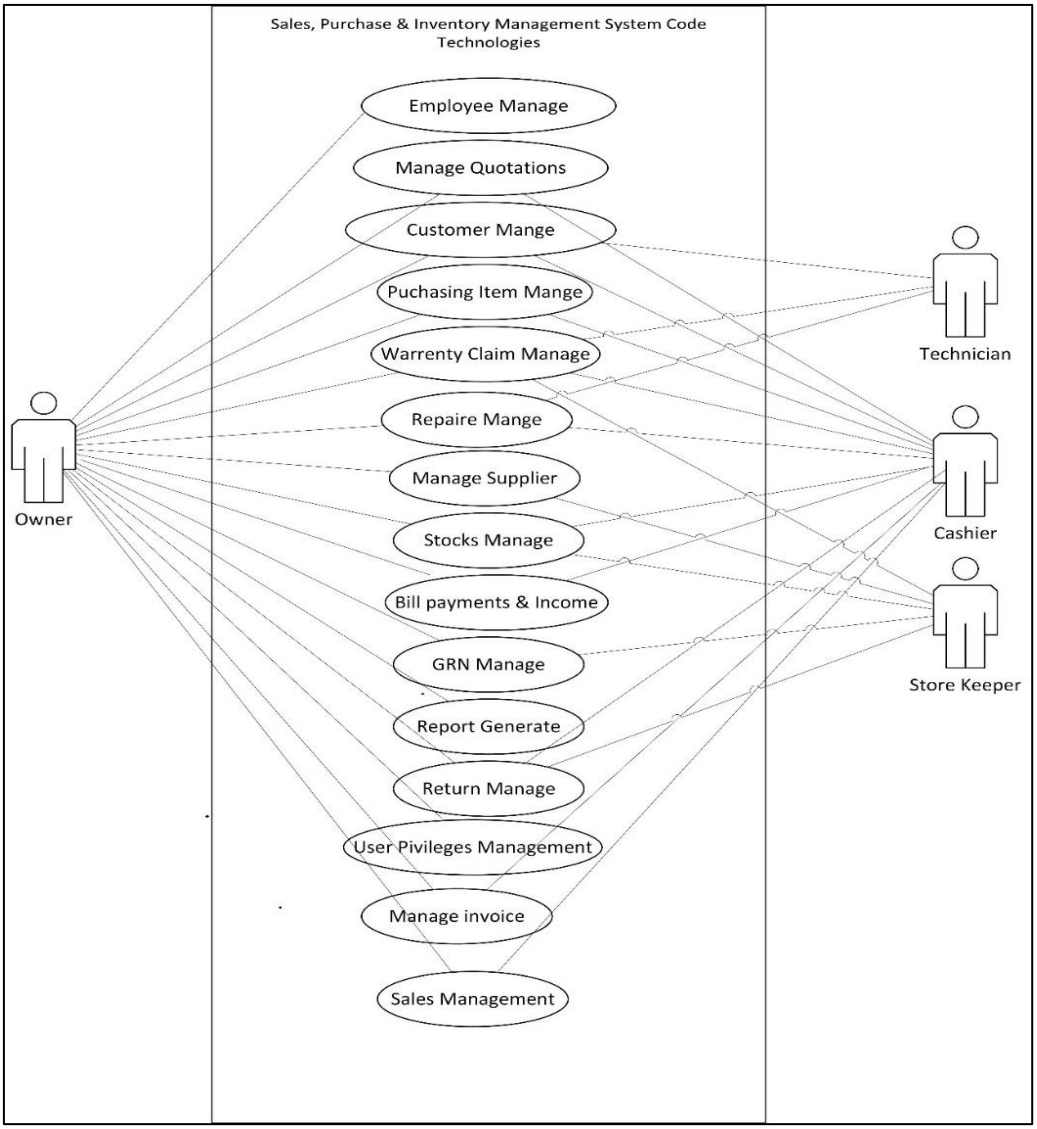


Figure 3.1 : Overall usecase diagram

3.2.2 Dependency Use Case Diagram

The figure 3. 2 below depicts the primary functions of the Sales and Inventory Management System.

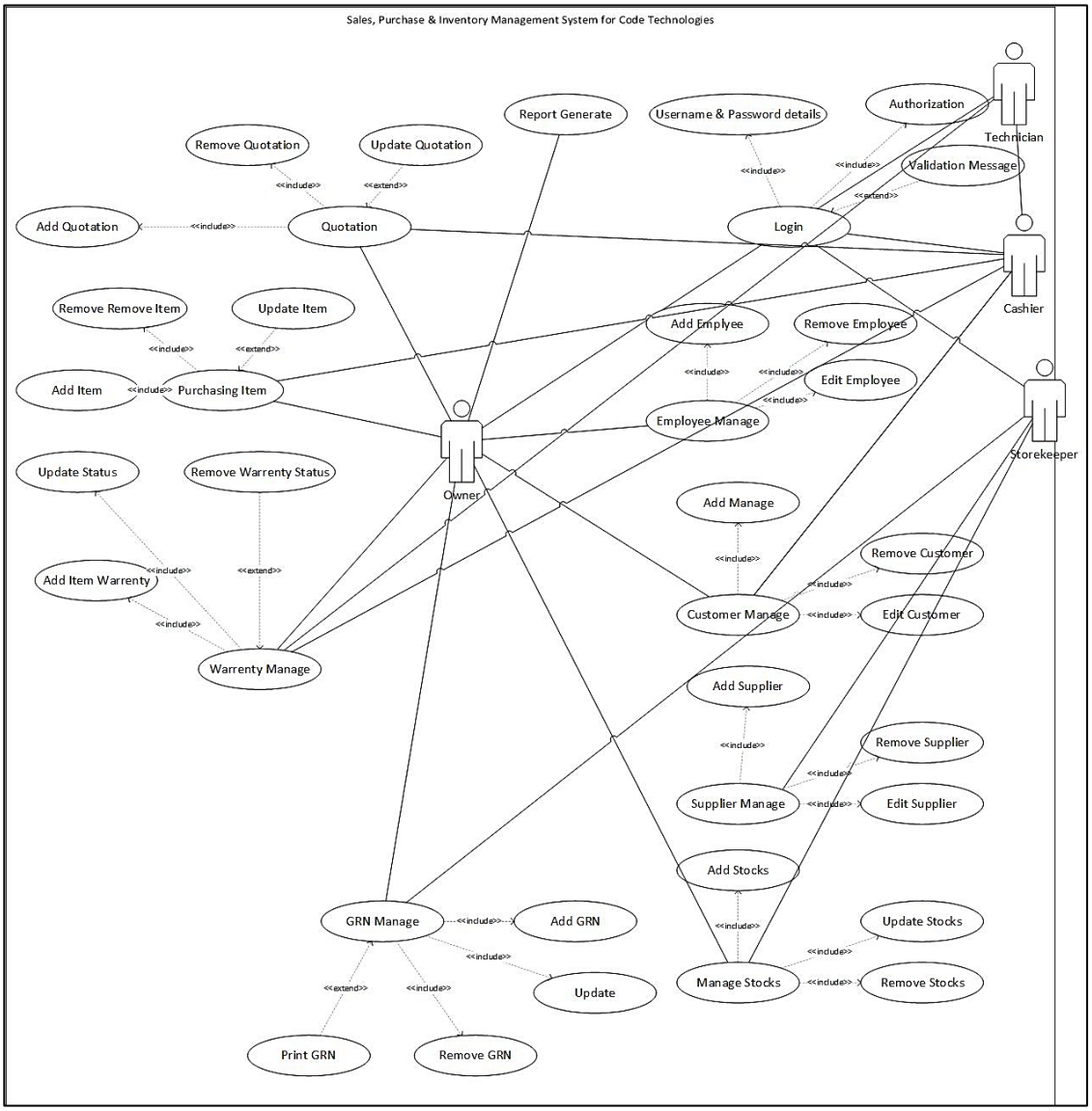


Figure 3.2 : Dependency Use Case Diagram

- **Customer Management**

Figure 3.3 depicts the customer details management use case diagram. To arrange a new sale, the customer must be registered in this system.

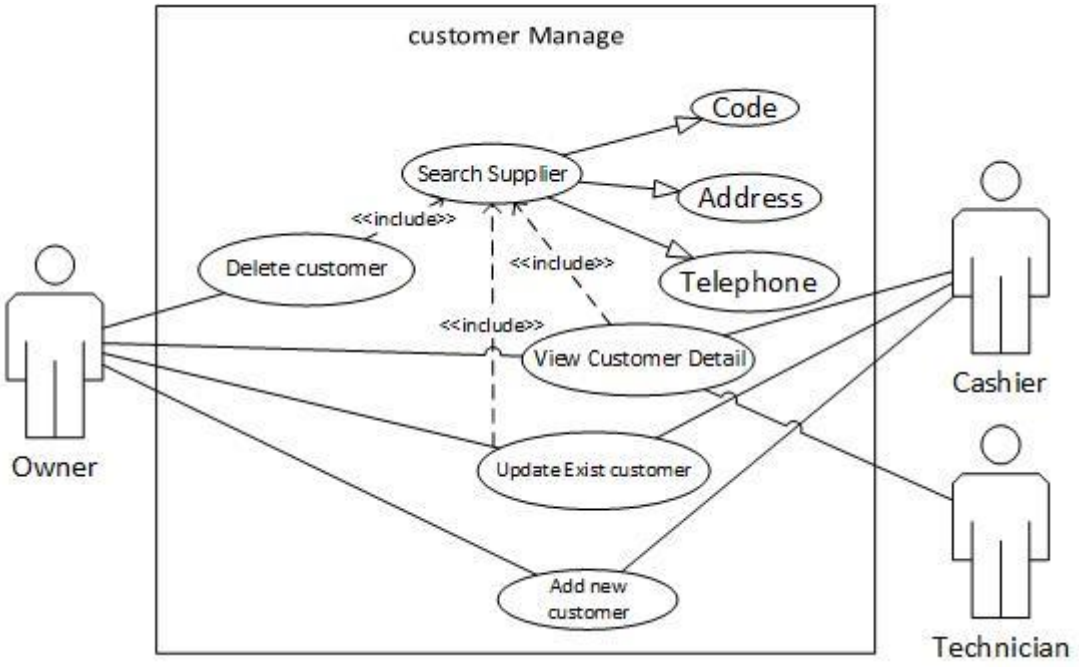


Figure 3.3 : Usecase diagram for manage Customer.

3.2.3 Activity Diagram

The activity diagram depicts a scenario of a certain system operation by displaying the flow of operations based on the decisions made by both the user and the system. In the Sales and Inventory Management System, the activity diagram below depicts the process of selling things to a consumer.

Figure 3.4 depicts the activity diagram to illustrate to when the login in to the system how it works.

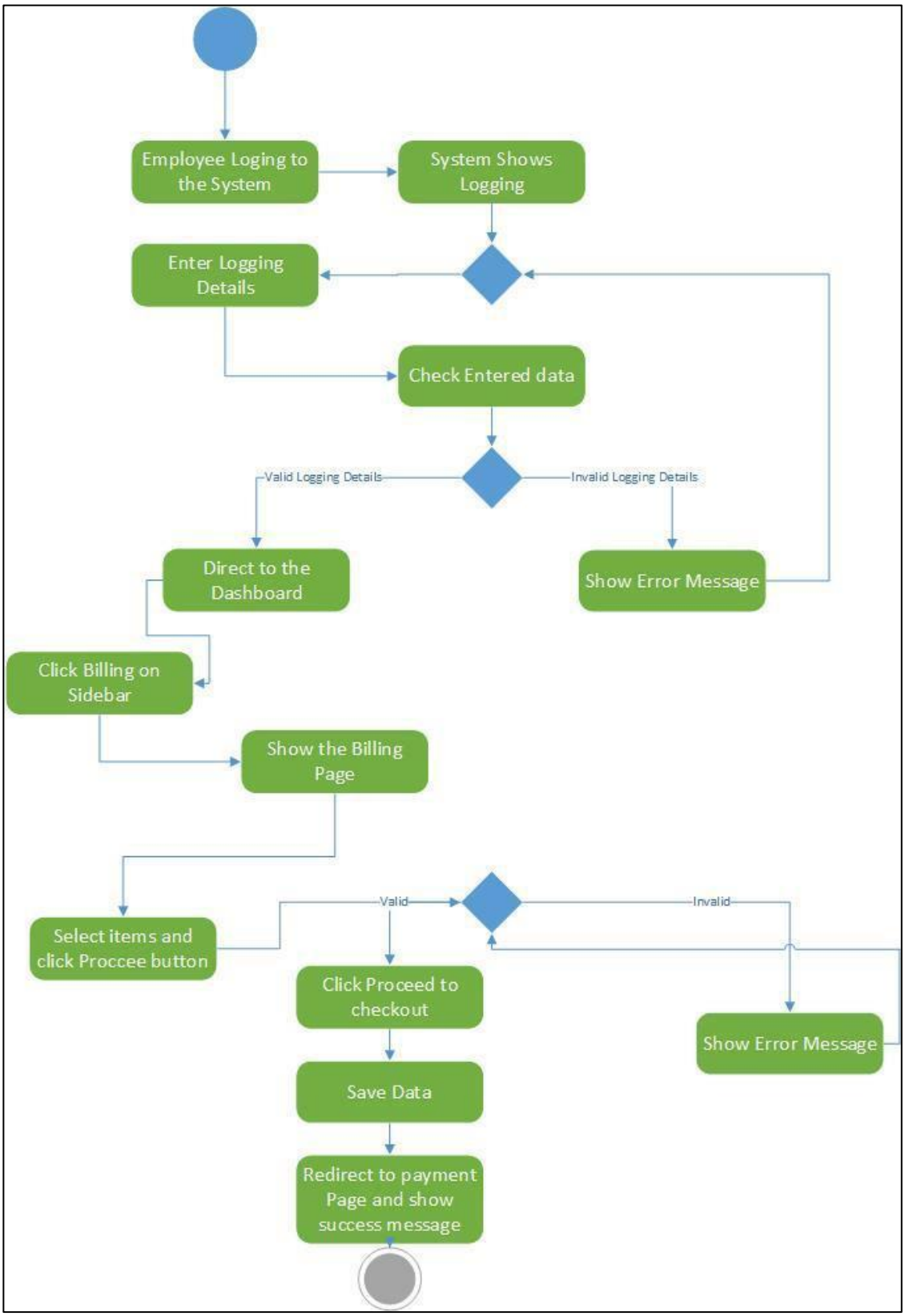


Figure 3.4 : Activity Diagram for user login

Figure 3.5 depicts the activity diagram to illustrate to when sell the item to the customer.

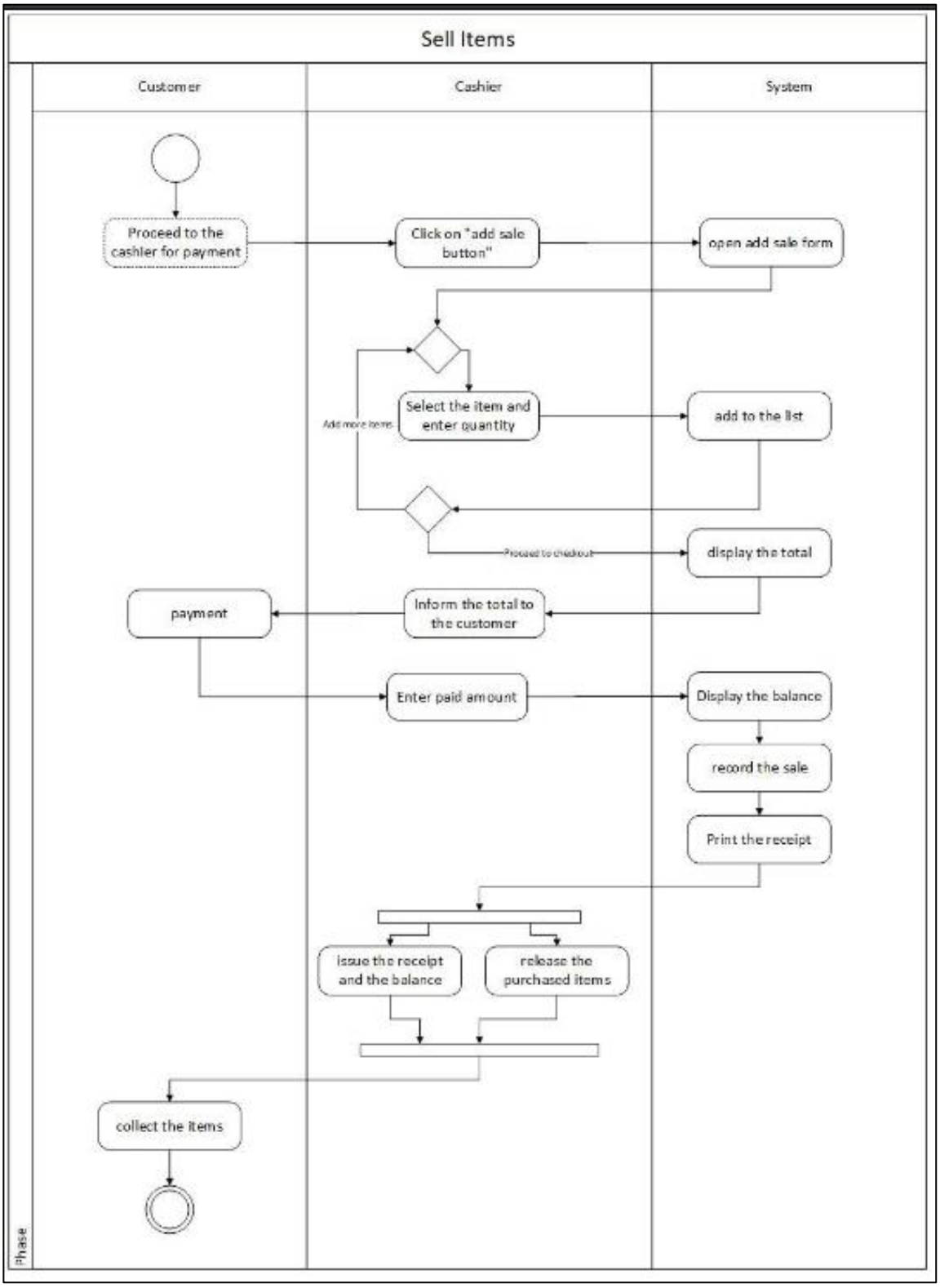


Figure 3.5 : Activity diagram for sale item

3.2.4 Class diagram

Class diagrams are employed in object modeling to illustrate the static structure of a system. On the system's degree of complexity, just a single class diagram can represent the entire system or many class diagrams can represent its components. System or component designs are represented using class diagrams. They can represent the system's objects by modeling them, displaying their relationships, and describing their operations and services.

Figure 3.6 shows the class diagram of the sales and inventory management system.

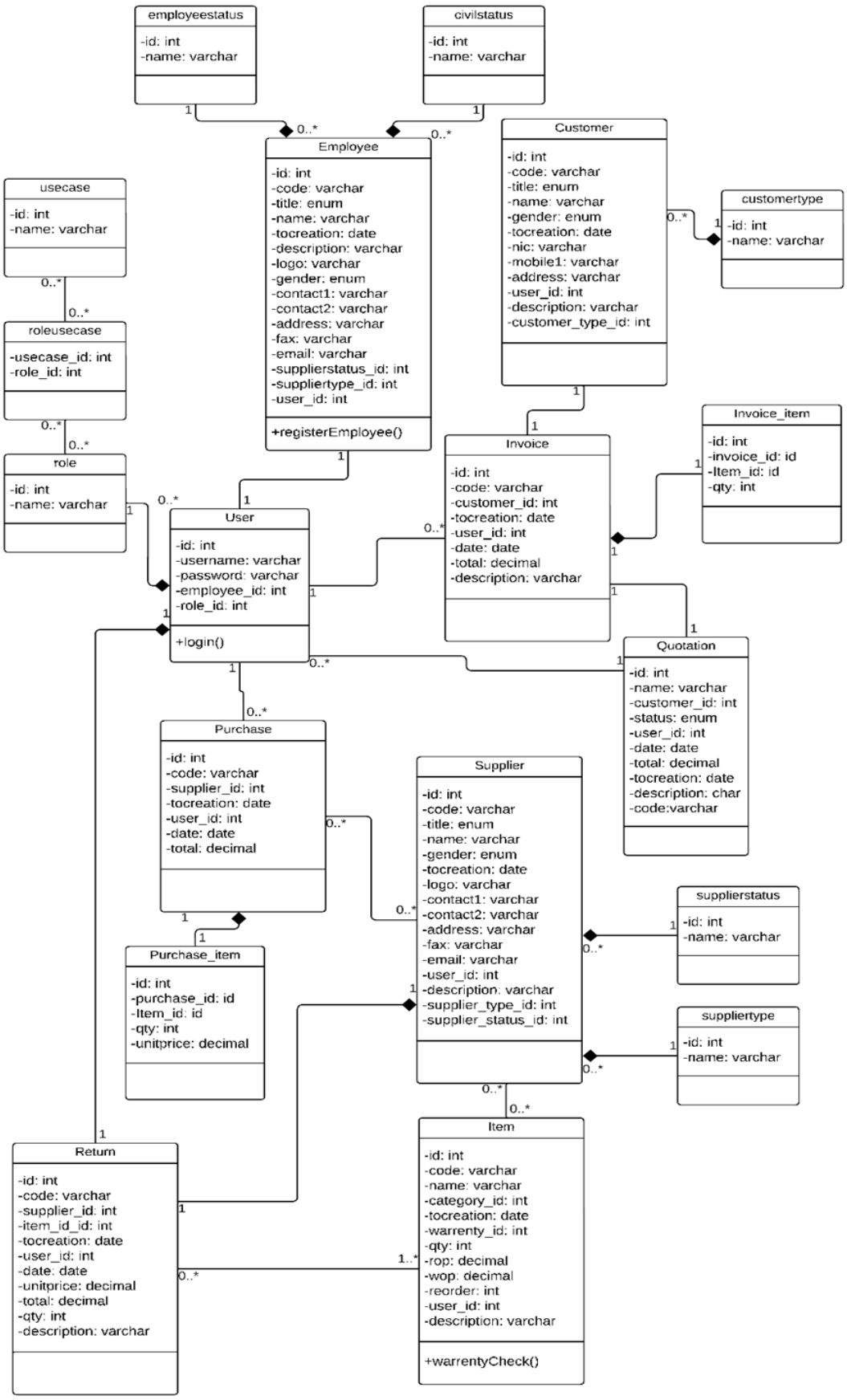


Figure 3.6 : Class diagram of the system

3.2.5 Sequence diagram

In the realm of software engineering, a sequence diagram, also known as a system sequence diagram, illustrates the chronological sequence of process interactions. It visually represents the processes, objects, and the sequential exchange of messages required to execute the intended functionality.

Following figure 3.7 shows sequence diagram for create new repair PC order accept.

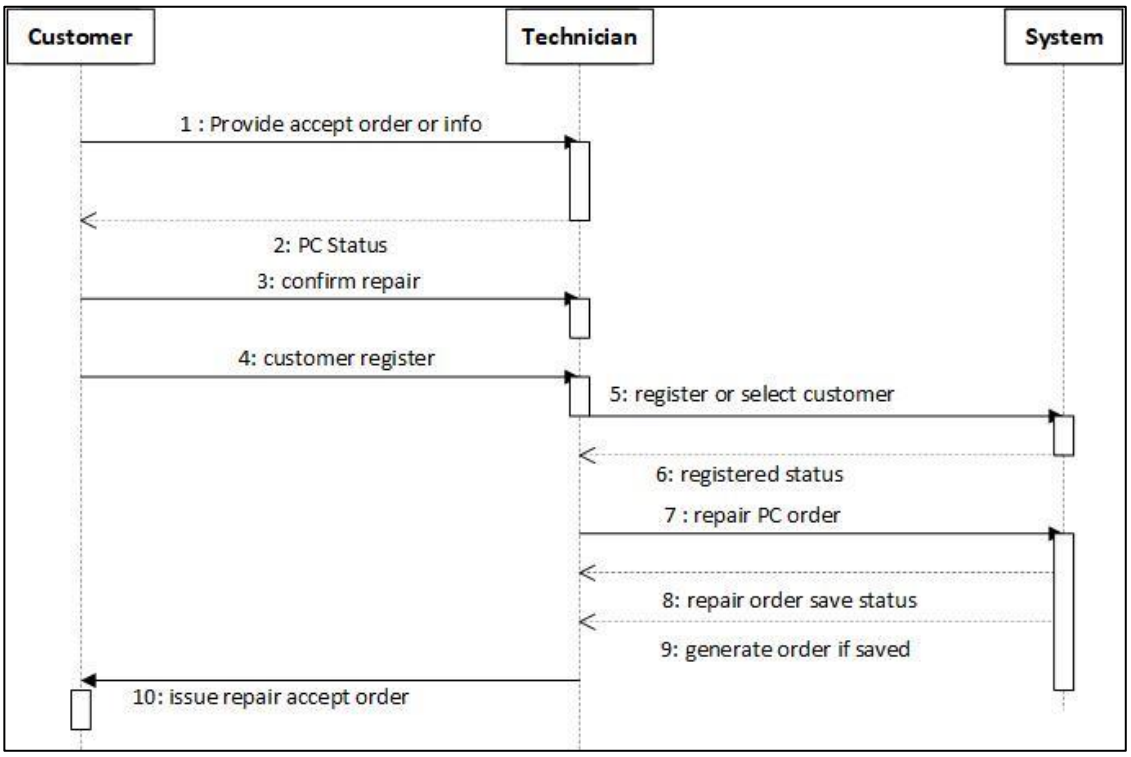


Figure 3.7 : Sequence diagram for create new repair PC order accept

Following figure 3.8 shows sequence diagram for create invoice for repair PC order

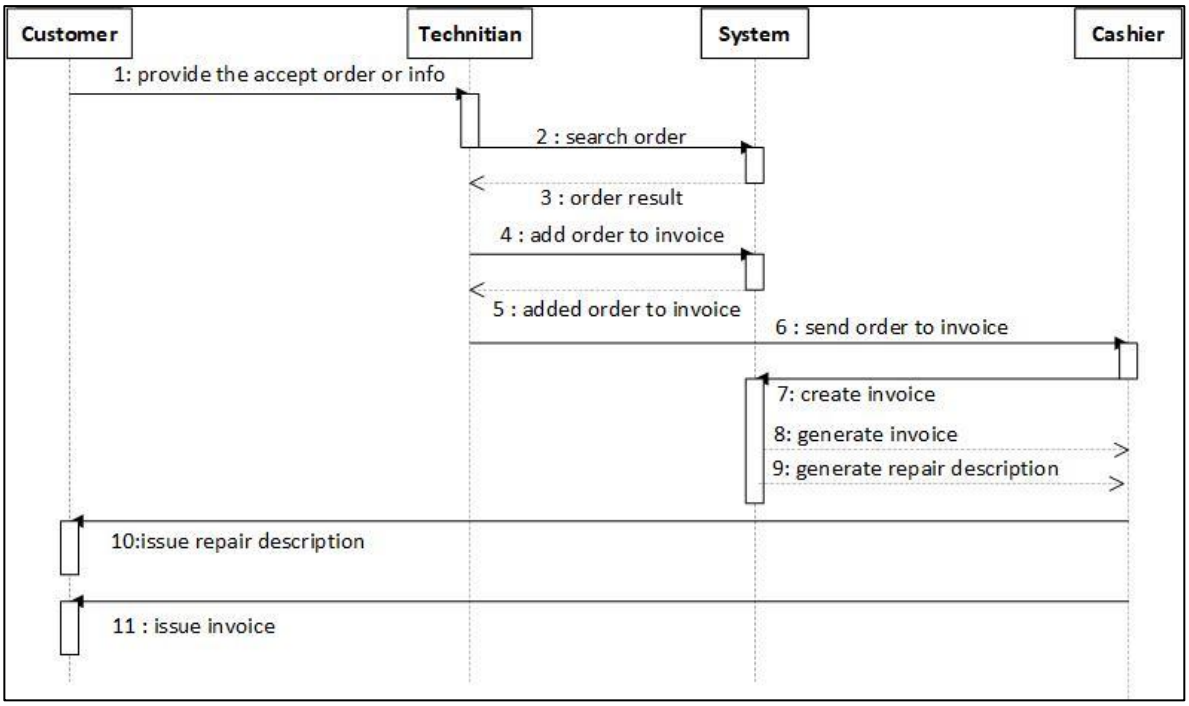


Figure 3.8 : Sequence diagram for create invoice for repair PC order

3.3 Data modeling

Database design is a collection of actions or procedures that aid in the creation, development, implementation, and upkeep of data management systems. A smart database architecture reduces maintenance costs, enhances data consistency, and impacts cost-effective disk storage solutions.

Following figure 3.9 shows EER of the system generated by MYSQL Workbench. Following figure 3.10 shows ER diagram for getting logical idea of this system.

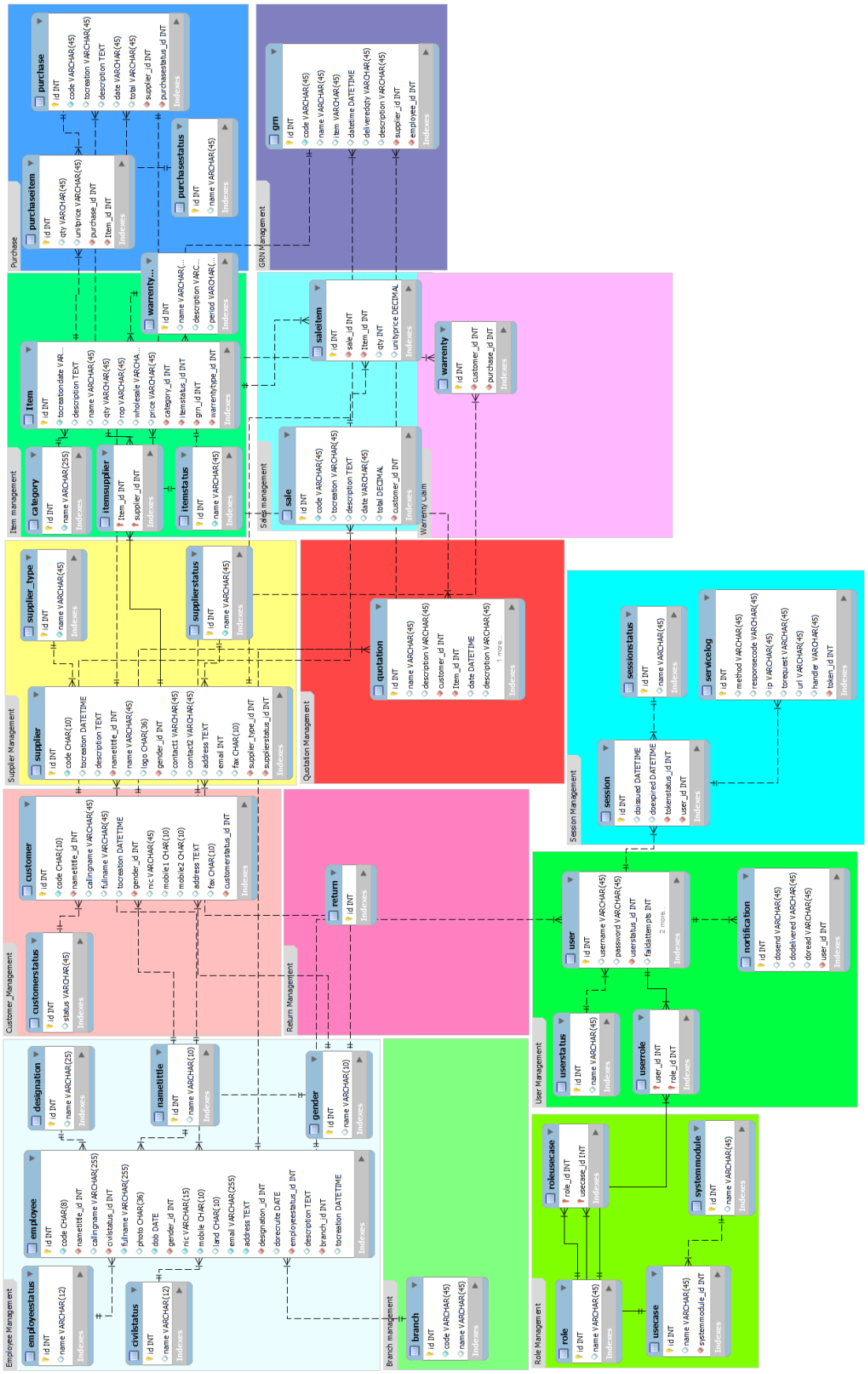


Figure 3.9 : EER Diagram I

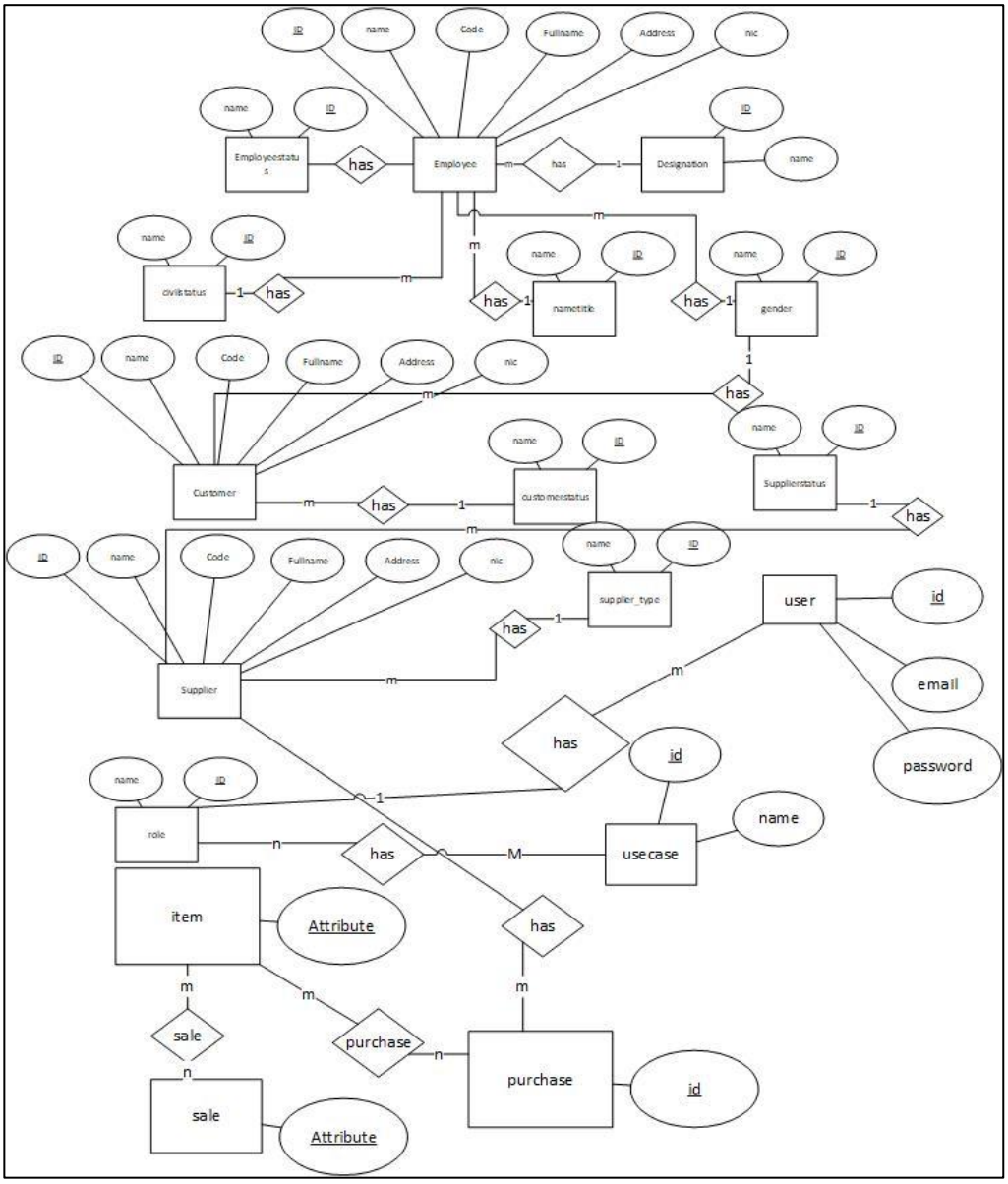


Figure 3.10 : EER Diagram 2

3.4 User interface design

The design of software, websites, or apps is referred to as user interface design. It is about programming the appearance of objects to improve usability and user experience.

3.4.1 Login Page for Users

User experience (UX) is crucial for login page design. A design that is effective attracts and turns fresh visitors into leads. It also helps to return users to log in fast. The following is the UI design for code Technology's sales and inventory management system.

The following figure 3.11 shows the user login interface of the developed system.

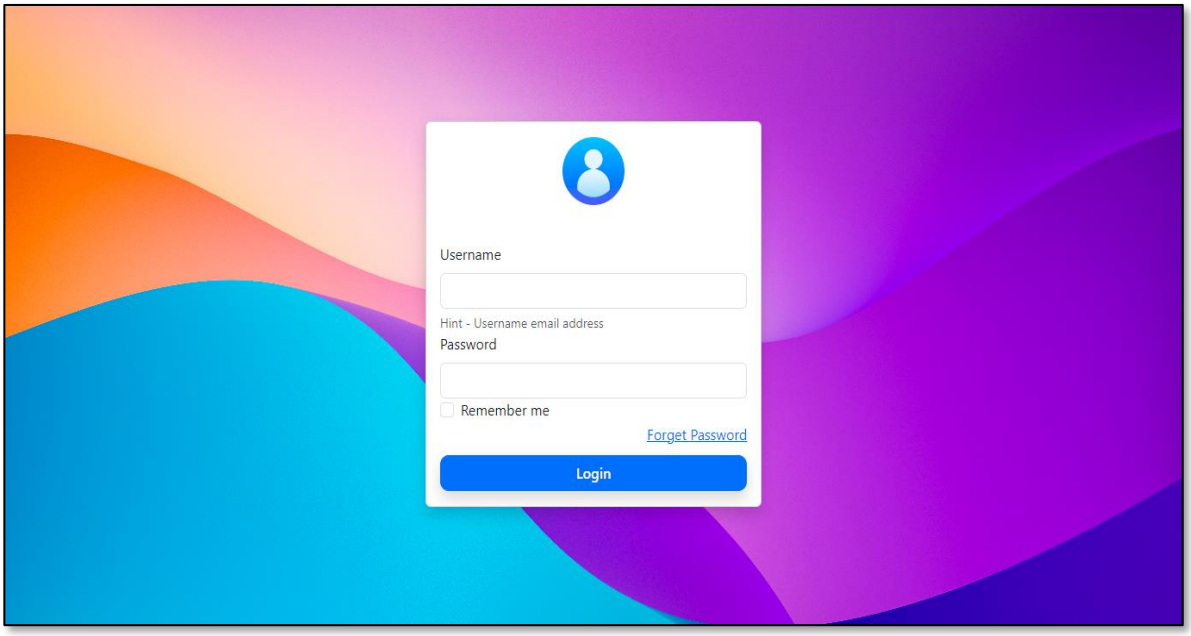


Figure 3.11 : Screenshot of the login page

3.4.2 Dashboard

This is the dashboard, often known as the administrator's home page. The menus on the left display goods, today's sales, categories, today's purchases, suppliers, and system users based on the client's preferences.

The figure 3.12 shows the admin dashboard of the developed system.

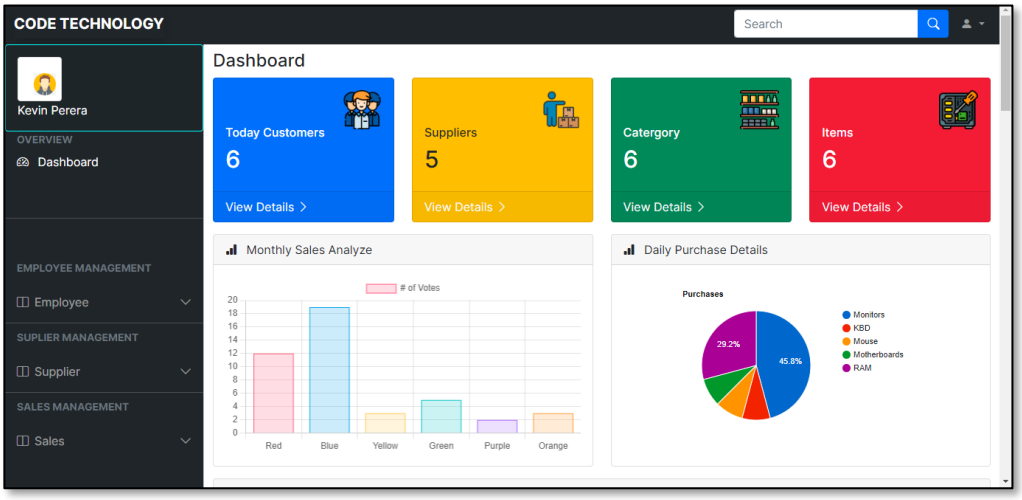


Figure 3.12 : Screenshot of the dashboard page

3.4.3 Customer Registration

Customer registration can be done only administrator and cashier. The figure 3.13 shows the customer registration of the developed system.

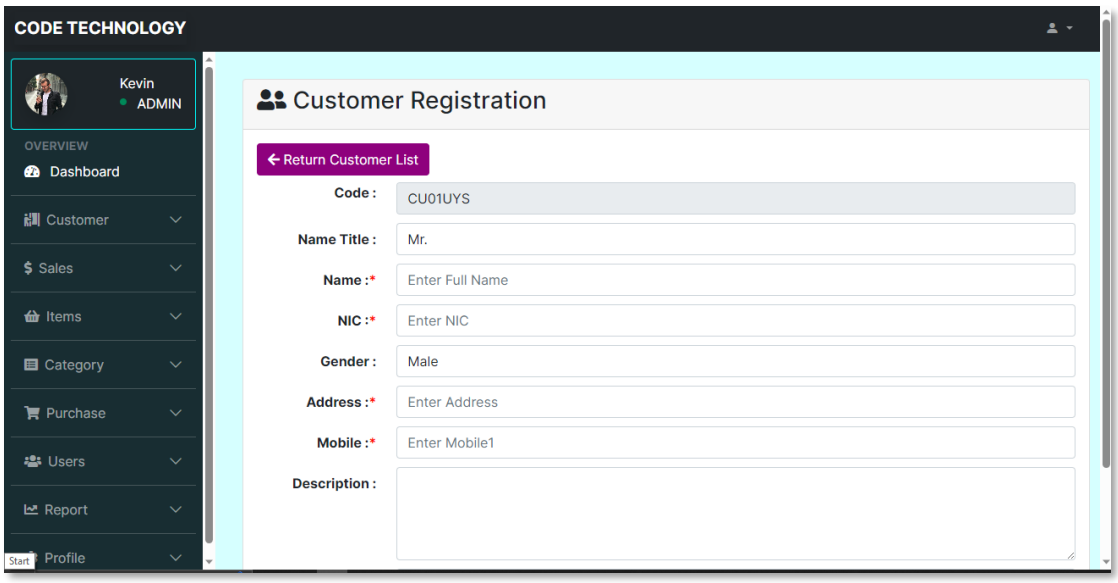


Figure 3.13 : Screenshot of customer registration

3.4.4 Customer Registration

List of customers can see administrator, technician and cashier. The figure 3.14 shows the list of customers. Only administrator can do all CRUD operations, but cashier only can add and edit customers only

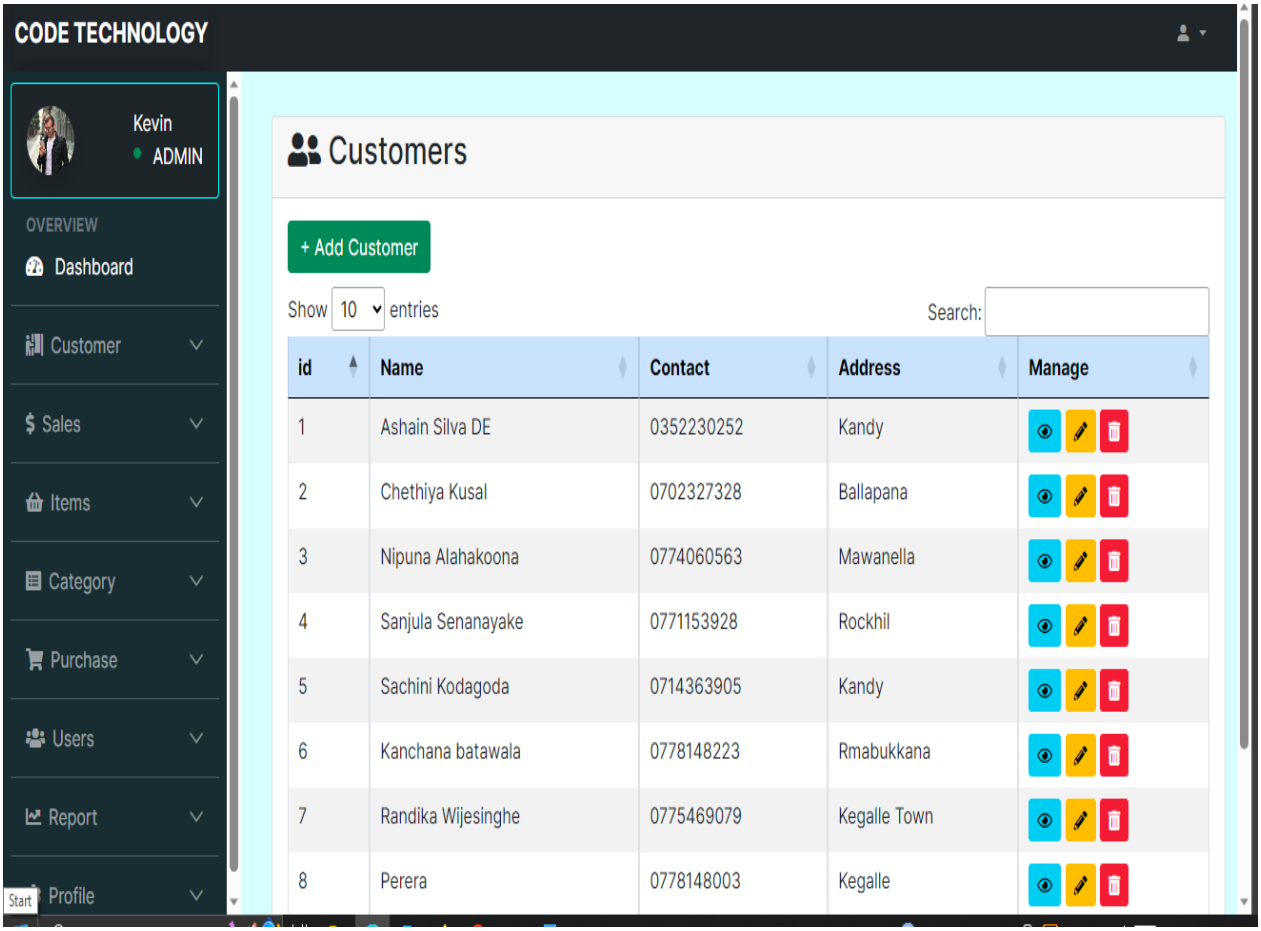


Figure 3.14 : Screenshot of list customers

3.4.5 Add New Supplier

This is the section where the user may add new providers. To add a new supplier, the user must provide a Supplier CODE, Supplier name, address, Contact Number, Supplier status, and Supplier type.

Figure 3.15 shows supplier registration form

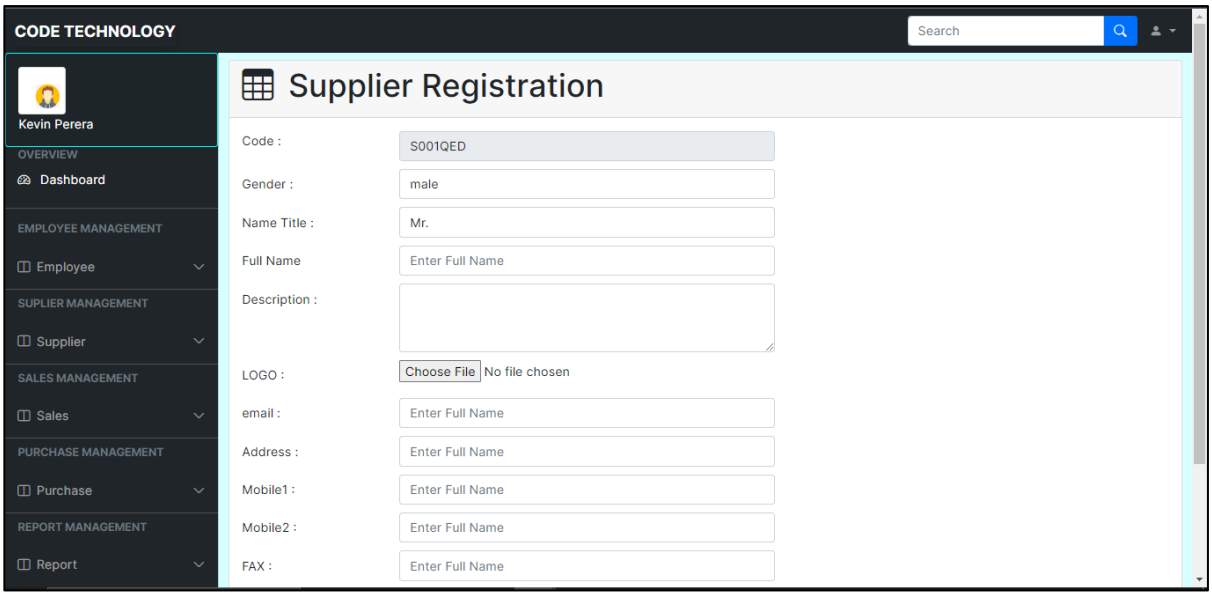


Figure 3.15 : Screenshot of supplier registration

3.4.6 View Supplier List

This is the section where the user may view suppliers. Figure 3.16 shows supplier list

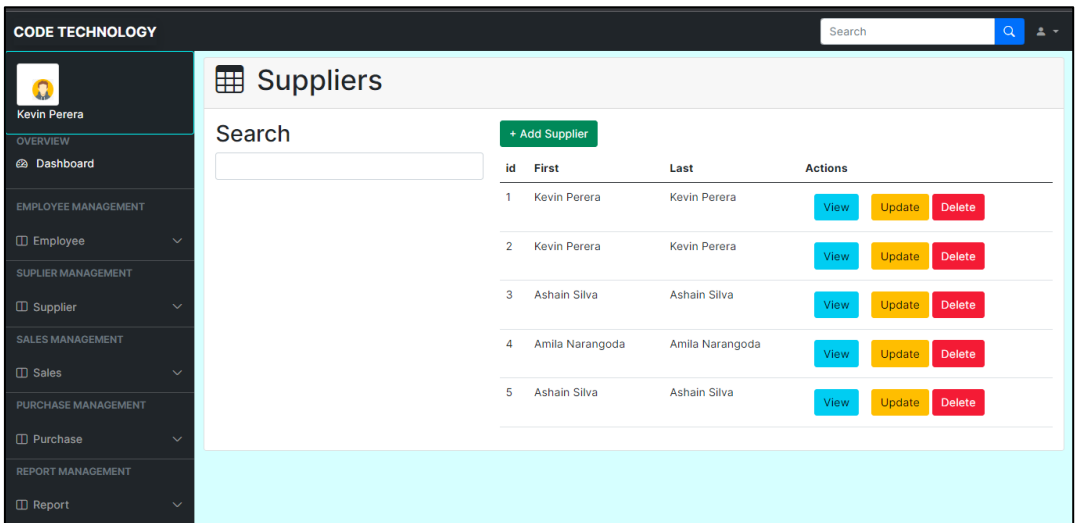


Figure 3.16 : Screenshot of list of suppliers

3.4.7 Popup Alert Messages

The system's message boxes were designed using user interface design concepts. Use colors, text, and pictures that are easy to grasp and recognize for system users.

Figure 3.17 depicts the system's success message boxes.

This popup message used to show when operation done by user, then user wants get an idea about this therefore if the process success, then user can get the idea by seeing the success message popup displayed in the system

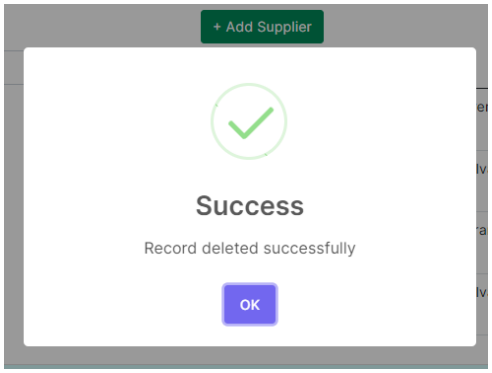


Figure 3.17 : Screenshot of success message

Figure 3.18 depicts the system's success message boxes. This Popup message used to show confirming this user decision before performing the operation. Example delete of specific data record

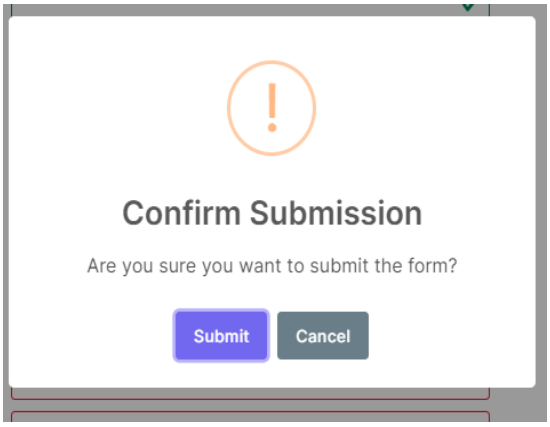


Figure 3.18: Screenshot confirmation message

3.4.8 Mobile Responsive creative design

The system was designed to work on any device, including computers, tablets, and smartphones. The main screen and other system components may be run on devices with varying display sizes.

Figure 3.19 depicts the system's responsive menu.

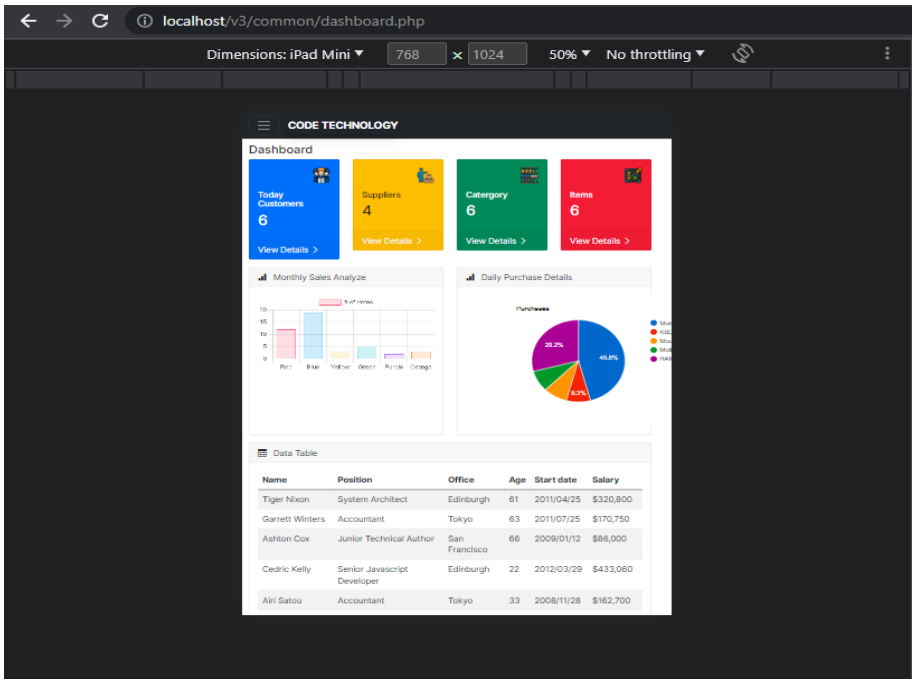


Figure 3.19 : Screenshot of the Mobile Responsive

Figma UI PrototypeDashboard Design for proposed system

Figure 3.20 shows prototype of the pre development stage concept it will give more ideas to implement the system. Examples Figma always give inspect of CSS

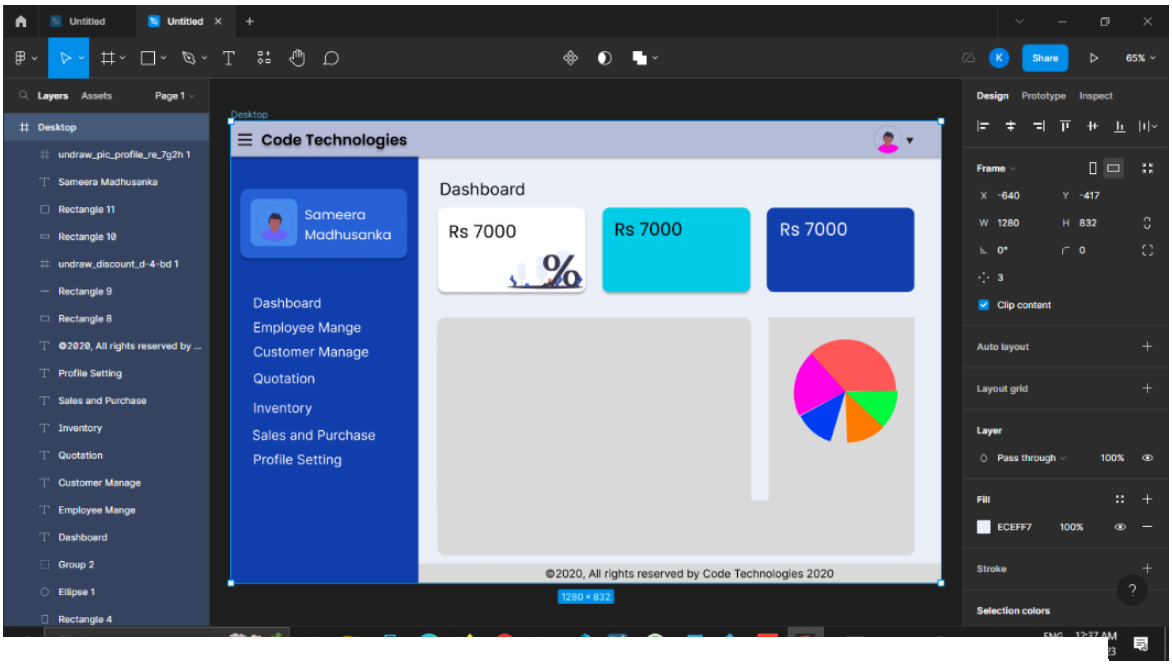


Figure 3.20 : Screenshot Figma Design

Chapter 4 – Implementation

Implementation is an extremely time-consuming and crucial stage of the entire project. During this step, the system's design is turned into executable software by developing the system. The tools, methodologies, architectures, and implementation environment utilized for the development of the Sales and inventory Management System are described in this chapter. This chapter includes code samples for various important features as well as third-party reusable components.

4.1 Implementation Environment

The implementation environment covers all of the tools and technologies required to construct this system, including both hardware and software.

4.1.1 Hardware and Software Requirements

For the creation of this system, the following software and hardware assets were employed.

Hardware Specification

- Core i3 6th generation Processor
- 8GB DDR4 2133Mhz
- 128 SSD 6Gbps
- Stable Internet Connection

Software Resources

- Windows 10 64bit OS
- MySQL Server with Workbench
- WAMP Server
- Microsoft Visual Studio code
- Microsoft Visio
- Microsoft Powerpoint
- Microsoft Words
- Apache web Server
- Mozilla Firefox
- Google Chrome

4.2 Implementation environment

For Client Side

HTML

HTML, or HyperText Markup Language, is the recognized markup language for texts intended to be displayed in a browser for the internet. **[4]**

BOOTSTRAP

Bootstrap is an open-source, free CSS framework for front-end web development that is responsive and mobile-first. It includes templates for layout, buttons, forms, navigation, and other UI components in HTML, CSS, and JavaScript. **[5]**

For Server side

PHP

PHP is a general-purpose programming language that is mostly used for web development. PHP is used to manage dynamic content, databases, session monitoring, and other similar tasks. **[6]**

MYSQL

MySQL is a well-known relational database management system that is frequently used in applications such as data warehousing, e-commerce, and logging. It is, however, most typically utilized as a database solution. **[7]**

AJAX

Ajax is a set of web development approaches that construct asynchronous online applications by utilizing multiple web technologies on the client side. Ajax allows web applications to transmit and get data from the server asynchronously without disrupting the existing page's appearance and behavior. **[8]**

jQuery

jQuery is used in the user UI of this web-based application for handling events and other validations. **[9]**

Apache

Apache is a popular open-source web server program. It is in charge of receiving HTTP requests from users and delivering the requested information in the form of website pages and files. **[10]**

4.2.1 Justification for Using Software

Following the completion of the client's requirements, the appropriate development technologies were carefully selected to begin the construction of the Sales and Inventory Management System. The technology was chosen for cross-platform web application development.

As a result, the most frequently used web development tools and frameworks in the industry were picked for the system's development.

MySQL was utilized for backend programming, and Bootstrap was used for front-end development. Other technologies, such as AJAX, were also leveraged to improve the system's interactivity and flexibility while reducing response time. Section 4.2 contains more information on the chosen technology. Choosing commonly used technologies with care was beneficial since they have adequate documentation and additional assets on the internet. It has aided development by making it easier to find answers to numerous development difficulties.

4.3 Module Structure

This software was created using common, industry-level programming standards and architectures. When the codebase grew, it became incredibly valuable and relatively straightforward to manage. The software architecture used for development, as well as some of the primary code parts of this system, are given here.

4.3.1 Folder Architecture

When managing a huge codebase, adhering to a suitable software architecture is quite beneficial. The MVC architecture was used to create the Sales and Inventory Management System. MVC is an abbreviation for model-view-controller, and it is one of the most popular software structures in the whole software engineering business. It aids in the correct organization of the codebase by dividing it up into different components, increasing the system's manageability. It also improves the reusability of those individual components.

The figure 4. 1 below depicts a graphical illustration of the structure of MVC [7].

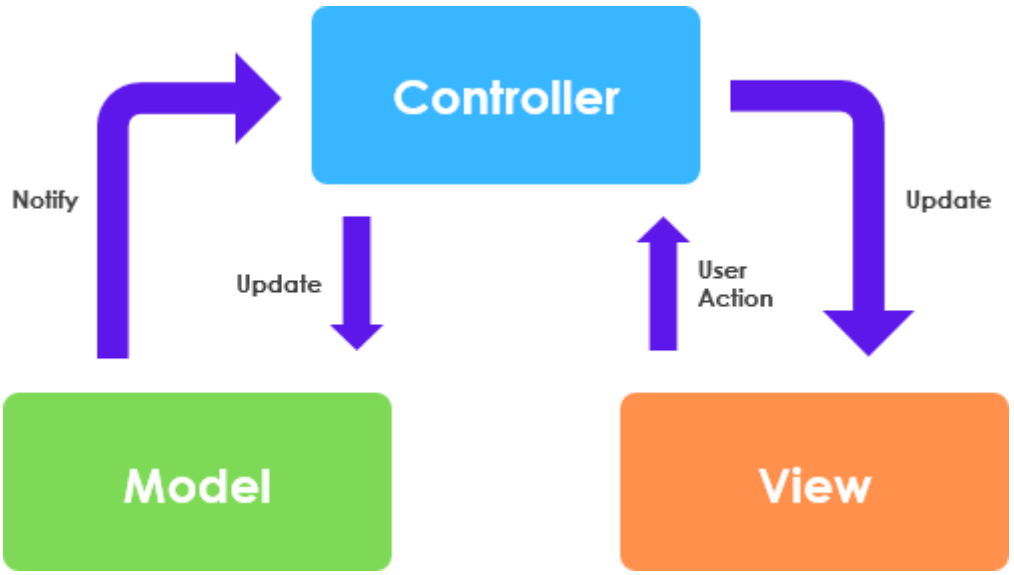


Figure 4.1: Screenshot of the MVC Architecture

4.4 Code explanation

The primary code segments used to construct the system's core are given below.

Database Connection

The figure 4.2 show database connectivity configurations code. It is used to connect the MYSQL database.

Common > config > dbconnection.php > ...

```php
<?php

// Database configuration
$host = 'localhost';
$dbName = 'v3';
$user = 'root';
$password = '1234';

// Create a new MySQLi instance
$con = new mysqli($host, $user, $password, $dbName);

// Check connection
if ($con->connect_error) {
    die("Connection failed: " . $mysqli->connect_error);

}

?>
```

Figure 4.2 : Screenshot of the Database Connection

Dashboard View

Common > Dashboard.php

```php
<?php
include('Pages/Header.php');

require('Top_nav.php');

require('Side_nav.php');

include('Pages/offcanvas.php');

include('Pages/Footer.php');

?>
```

Figure 4.3 : Screenshot of the Dashboard code

Login Page Code

```php
<?php
require('Pages/Login_Header.php');
?>
<style>···
</style>
<div class="container">
    <div class="row">
        <div class="d-flex align-items-center justify-content-center">
            <div class="card shadow">
                <div class="card-body">
                    <form method="post" action="<?php $_SERVER['PHP_SELF']; ?>">
                        <img src="../Common/Assets/images/Login/user.png" alt="usr_logo" srcset="" class="logo1">
                        <h1 class="head">
                            <div class="s1" style="margin-top:12px">
                                <span style="background-color: #ffffff; padding: 0 10px;font-weight:900;  font-weight: bold;font-size:xx-large; m
                                    Login
                                </span>
                            </div>
                        </h1>
                        <div>
                            <label for="exampleInputEmail1" class="form-label">Username</label>
                            <input type="email" class="form-control" id="exampleInputEmail1" aria-describedby="emailHelp" name="username">
                            <div id="emailHelp" class="form-text">Hint - Username email address</div>
                        </div>
                        <div>
                            <label for="exampleInputPassword1" class="form-label">Password</label>
                            <input type="password" class="form-control" id="exampleInputPassword1" name="password">
                        </div>
                        <div class="form-check">
                            <input type="checkbox" class="form-check-input" id="exampleCheck1">
                            <label class="form-check-label" for="exampleCheck1">Remember me</label>
                        </div>
                        <div class="forget">
                            <a href="#">Forget Password</a>
```

⊗ 0 ⚠ 0 Ln 1, Col 1 Spaces: 4 UTF-8 CRLF PHP Go Live

Figure 4.4 : Screenshot of the Login page code

```php
                        </div>
                        <div class="forget">
                            <a href="#">Forget Password</a>
                        </div>

                        <button type="submit" class="btn btn-primary shadow">Login</button>
                    </form>
                </div>
            </div>
        </div>
    </div>
</div>
<?php

require('config/dbconnection.php');

if ($_SERVER['REQUEST_METHOD'] === 'POST') {

    // Access the submitted values
    $username = $_POST['username'];
    $password = $_POST['password'];
    $userEnteredPassword = $password; // Assuming the user-entered password is received via a form post.

    $query = "SELECT password FROM user WHERE username = '$username'";


    $result = mysqli_query($con, $query);


    if ($result && mysqli_num_rows($result) > 0) {
        $row = mysqli_fetch_assoc($result);
        $hashedPasswordFromDB = $row['password'];
```

Figure 4.5 : Screenshot of the Login page code II

```
$result = mysqli_query($con, $query);


if ($result && mysqli_num_rows($result) > 0) {
    $row = mysqli_fetch_assoc($result);
    $hashedPasswordFromDB = $row['password'];

    // User input password
    $userPassword = $_POST['password']; // Assuming the password is submitted via a form

    // Verify the user's password
    if (password_verify($userPassword, $hashedPasswordFromDB)) {
        // Passwords match
        $_SESSION['user_role'] = 'admin';
        echo '
                <script>
                    window.onload = function() {
                     swal({
                         title: "Click to load dashboard",
                        icon: "info",
                    buttons: {
                    confirm: {
                        text: "Login",
                        value: true,
                        visible: true,
                        className: "btn btn-primary",
                        closeModal: true
                        }
                    }
                    }).then(function() {
                        window.location.href = "common/Dashboard.php";
                     });
                    };
```

Figure 4.6 : Screenshot of the Login page code III

```
    $userPassword = $_POST['password']; // Assuming the password is submitted via a form

    // Verify the user's password
    if (password_verify($userPassword, $hashedPasswordFromDB)) {
        // Passwords match
        $_SESSION['user_role'] = 'admin';
        echo '
            <script>
                window.onload = function() {
                 swal({
                     title: "Click to load dashboard",
                     icon: "info",
                buttons: {
                confirm: {
                    text: "Login",
                    value: true,
                    visible: true,
                    className: "btn btn-primary",
                    closeModal: true
                    }
                }
                }).then(function() {
                    window.location.href = "common/Dashboard.php";
                 });
                };
            </script>
        ';
    //  "<script>
    //     swal('Welcome!', 'Verified User!', 'success');
    //     // .then(function ()=>{
    //     // window.location.href = 'dashboard.php';
    //     </script>";
```

Figure 4.7 : Screenshot of the Login page code IV

Supplier Save

```php
<?php
require('pages/footer.php');
?>

<?php
if ($_SERVER['REQUEST_METHOD'] === 'POST') {


  // Access the submitted values
  $sup_code = $_POST['sup_code'];

  //should get values from foreign tables id's
  $nametitle = $_POST['nametitle'];
  $fullname = $_POST['fullname'];


  $description = $_POST['description'];


  // $proimg = $_POST['proimg'];
  $image = $_FILES['logoimg']['name'];
  $image_tmp = $_FILES['logoimg']['tmp_name'];


  $gender = $_POST['gender'];
  $contact1 = $_POST['mobile1'];
  $contact2 = $_POST['mobile2'];
  $address = $_POST['address'];
  $email = $_POST['email'];
  $fax = $_POST['fax'];
  // Read the image content
```

Figure 4.8 : Screenshot of the Supplier page code

```
// Read the image content
// $imageData = file_get_contents($image);

$supplierstatus_id = $_POST['supplier_status'];
$suppliertype_id = $_POST['supplier_type'];


// $targetDir = "uploads/";
// $targetFile = $targetDir . basename($_FILES["image"]["name"]);

$sql = "INSERT INTO supplier ( code, nametitle_id, name, description, logo,gender_id, contact1, contact2,
  address, email, fax, supplierstatus_id,suppliertype_id) VALUES ('$sup_code',$nametitle,
   '$fullname','$description','$image','$gender','$contact1','$contact2','$address','$email','$fax',
   $supplierstatus_id,$suppliertype_id)";
$result = mysqli_query($con, $sql);

// Print the query statement
// echo "Query: " . $sql . "<br>";
if ($result) {
  // Display SweetAlert success message
  echo "
<script src='https://unpkg.com/sweetalert/dist/sweetalert.min.js'></script>
<script src='https://cdnjs.cloudflare.com/ajax/libs/jquery/3.7.0/jquery.min.js' ></script>
<script>
  swal({
      title: 'Success!',
      text: 'Query executed successfully.',
      icon: 'success',
  }).then(function() {
      // Redirect to view.php
      window.location.href = 'supplier_view.php';
  });
</script>
```

Ln 404 Col 2 Spaces: 2 UTF-8 CRLF PHP Go Live

Figure 4.9 : Screenshot of the Supplier page code II

Authentication

```
1  <?php
2  session_start();
3
4  // Check if the user is authenticated
5  if (!isset($_SESSION["loggedin"]) || $_SESSION["loggedin"] !== true) {
6    // User is not logged in, redirect to login page
7    header("Location: login.php");
8    exit();
9  }
10
```

Figure 4.10 : Screenshot of the Authentication page code

Chapter 5 – Evaluation

To ensure that the system operates as planned, the built software must be tested and reviewed. Following the delivery of the software, the customer conducts a test to ensure that the system satisfies their business objectives and expectations.

5.1 Testing

Testing was carried out at various phases of development using the following testing methodologies.

5.1.1 Testing Types

Unit Testing

Each item of software is tested independently during unit testing. The primary goal of unit testing is to isolation each unit of the system and check that it behaves as intended. This may be used to quickly discover any flaws. [11]

Integration Testing

By testing several components as a group, integration testing strategy is used to evaluate the interaction between separate software components. This procedure can identify any compatibility concerns between distinct components. [12]

System Testing

Once all the components have been developed, the overall system is tested as a whole. The entire system is tested in this manner to ensure that it operates as planned. This testing process can be classified as black-box testing. [13]

Acceptance testing

Before approving the completely constructed system, the customer does acceptance testing. The customer tests the system to make sure that it meets their expectations and satisfies their business needs. [14]

5.2 Test Plan

The test plan specifies how the entire system must be tested and what methodologies should be employed. A multi-stage test strategy was used to test the Sales and Inventory Management System during its development. To ensure that the system operates as intended, certain commonly used testing methodologies such as unit testing, integration testing, system testing, and acceptance testing were utilized.

Black-box and white-box testing were also used to ensure that the system works as expected and satisfies all the client's criteria. Black box testing is concerned with the general functionality of the system rather than the internal logic of the system. White box testing is more technical in nature, focusing on the codebase and system logic.

After all of the developments were finished, an acceptance test was conducted at the customer location. A varied group of testers worked together to ensure that the system met their business needs. A questionnaire was used to collect feedback from the testers, and further information about this can be found in section 5.3.

5.2.1 Test Cases

The system was tested using various pre-defined test scenarios, the results of which are presented below.

Test case for User Login

Following table 5.1 shows test case for user login.

Test No	Test Case	Expected Result	Status
01	User Login – User Enter Correct Username Password	Successfully Login in to the System	PASS
02	User Login – User didn't enter username and Password	Display Error Message "Please Enter Username and Password"	PASS
03	User Login – User Enter Correct Username and enter incorrect password	Display Error Message "Please Enter Username and Correct Password"	PASS
04	User Login – User Enter Invalid User Email	Display Error Message "User Not Found"	PASS

Table 5.1 : Test case for User Login

Off canvas Bar Test Case

Following table 5.2 shows off canvas bar test case.

Test No	Test Case	Expected Result	Status
01	Total Categories	Show the Total Categories currently available.	PASS
02	Total Supplier	Show the total active suppliers of this shop.	PASS

03	Today Total Customers	Show the Daily customers count down.	PASS
04	Total Items	Show the total item or stocks available.	PASS

Table 5.2 : Off canvas Bar Test Case

Sidebar Test Case

Following table 5.3 shows side bar test case.

Test No	Test Case	Expected Result	Status
01	Clicks the "Dashboard" button.	Back to the Dashboard webpage	PASS
02	Clicks the "Sales" button.	Back to the Sales webpage	PASS
03	Clicks the "Customers" button.	Back to the Customer webpage	PASS
04	Clicks the "Supplier" button.	Back to the Supplier webpage	PASS
05	Clicks the "Purchase" button.	Back to the Purchase webpage	PASS
06	Clicks the "Users" button.	Back to the Users webpage	PASS
07	Clicks the "Return" button.	Back to the Return webpage	PASS
08	Clicks the "Report" button.	Back to the Report webpage	PASS
09	Clicks the "Setting" button.	Back to the Setting webpage	PASS

Table 5.3 : Sidebar Test Case

User Registration Test Case

Following table 5.4 shows user registration test case.

Test No	Test Case	Expected Result	Status
01	Check username.	When entering a username that is already in use within the system, an error message in red text should appear, stating, "The username is already taken."	PASS
02	Visibility of users	The Users page is exclusively accessible to the admin, who possesses the sole authority to manage user data and access this page.	PASS
03	Visibility of profile page	Each system user has the capability to modify their own restricted personal information, including their username and password.	PASS

Table 5.4 : User Registration Test Case

Supplier Test Case

Following table 5.5 shows supplier test case.

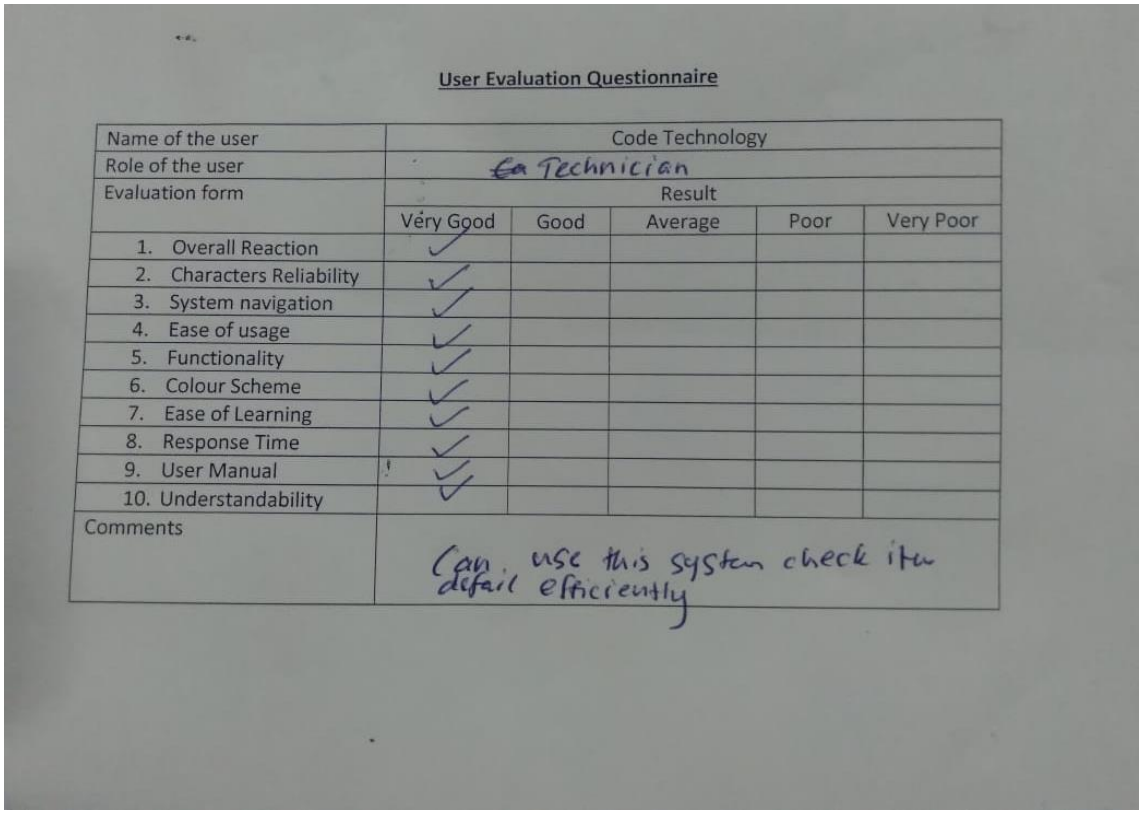
Test No	Test Case	Expected Result	Status
01	Active Supplier	When the supplier set as active and good as properties, then user can only purchases this supplier otherwise doesn't show in purchases phase	PASS
02	Blacklisted Supplier	If the supplier put as blacklisted, then user cannot purchases any item from that supplier. Administrator only can change this status.	PASS
03	Auto get last edited date	System user can able to view change of any supplier data who is change of particular data show changed time and logged user for enroll the to do particular scenario	PASS
04	Unique data	Getting supplier contact number that doesn't duplicate this number with another customer	PASS

Table 5.5 : Supplier Test Case

5.3 User evaluation

Some customer workers in various roles were recruited to test the system to see if it met their business requirements. The testers' feedback was gathered through interviews and the use of a questionnaire. The questionnaire results reveal that users were pleased with the system.

The figure 5.1 below depicts the Technician of Code Technology shop's responses to the evaluation questionnaire.

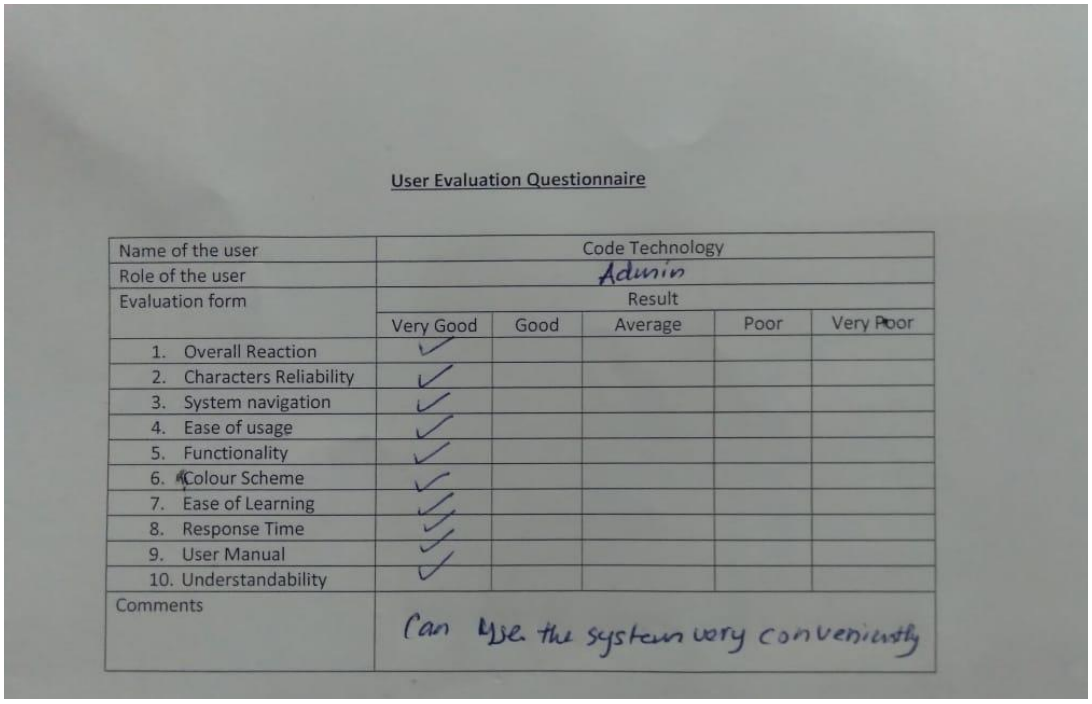


User Evaluation Questionnaire

Name of the user	Code Technology				
Role of the user	a Technician				
Evaluation form	Result				
	Very Good	Good	Average	Poor	Very Poor
1. Overall Reaction	✓				
2. Characters Reliability	✓				
3. System navigation	✓				
4. Ease of usage	✓				
5. Functionality	✓				
6. Colour Scheme	✓				
7. Ease of Learning	✓				
8. Response Time	✓				
9. User Manual	✓				
10. Understandability	✓				
Comments	Can use this system check item defail efficiently				

Figure 5.1 : User Evaluation Questionnaire I

The figure 5.2 below depicts the owner's responses to the evaluation questionnaire as the system administrator.

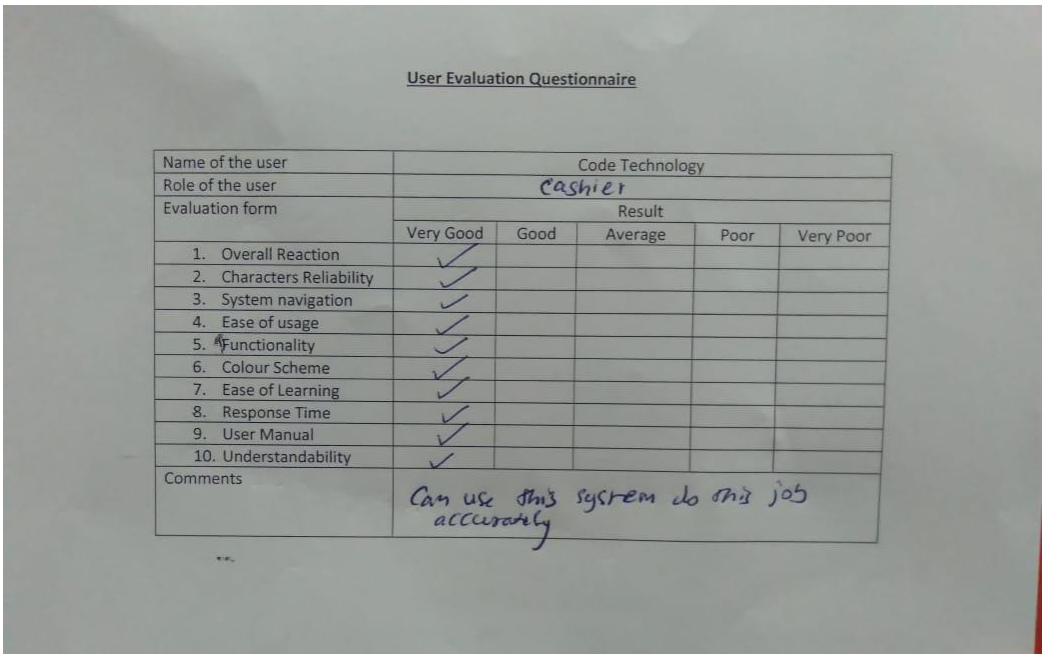


User Evaluation Questionnaire

Name of the user	Code Technology				
Role of the user	Admin				
Evaluation form	Result				
	Very Good	Good	Average	Poor	Very Poor
1. Overall Reaction	✓				
2. Characters Reliability	✓				
3. System navigation	✓				
4. Ease of usage	✓				
5. Functionality	✓				
6. Colour Scheme	✓				
7. Ease of Learning	✓				
8. Response Time	✓				
9. User Manual	✓				
10. Understandability	✓				
Comments	Can use the system very conveniently				

Figure 5.2 : User Evaluation Questionnaire II

The figure 5.3 below depicts the Cashier of Code Technology shop's responses to the evaluation questionnaire.



User Evaluation Questionnaire

Name of the user	Code Technology				
Role of the user	Cashier				
Evaluation form	Result				
	Very Good	Good	Average	Poor	Very Poor
1. Overall Reaction	✓				
2. Characters Reliability	✓				
3. System navigation	✓				
4. Ease of usage	✓				
5. Functionality	✓				
6. Colour Scheme	✓				
7. Ease of Learning	✓				
8. Response Time	✓				
9. User Manual	✓				
10. Understandability	✓				
Comments	Can use this system do this job accurately				

Figure 5.3 : User Evaluation Questionnaire III

Chapter 6 – Conclusion

This chapter outlines the entire project, from requirement gathering through handing over the fully built solution to the client. Some suggestions for future improvements to this system were also made.

6.1 Summary

6.1.1 Evaluation of the project

Code Technology is a well-known store in the Kegalle neighborhood. They offer all types of computers, components, and associated things directly to clients as retails and stocks, as well as hire buy. In their technical division, they also repair and service computers. Technicians, stockkeeper, cashier, and other personnel report to the shop manager.

From the beginning until the present, the wisdom carried out their daily tasks in a paper-based manual manner. They encountered several issues when manually storing client, supplier, and employee data, as well as creating purchase and selling regions. It is more difficult to create reports and monitor current store availability using the present manual approach. It has a detrimental impact on their business by making decisions more difficult and squandering time and money. To successfully address those concerns and problems in this company, a proposed new sales and inventory system was designed.

Gather the client's needs for the proposed system over a longer period of time. To properly determine stakeholder requirements, multiple types of fact seeking procedures were applied depending on the circumstance. Interviews are conducted with the person who owns stock keeper, cashiers, technicians, and other staff. Other times, ask basic questions and thoroughly examine their paperwork.

When doing system analysis, RUP approach was used. To clearly capture the client's requirements, many forms of UML diagrams were created, included diagrams for use cases, class diagrams, activity diagrams, and sequence diagrams. The ER diagram was used to guide the system database design process.

After considering the client's needs, the suggested system was constructed as a web-based system. The CODE technologies' personnel is concerned with the computer shop management system. The system is accessible to all users over the internet. PHP is used on the server side of the system. As a database management system, an MSSQL server is employed. The system's client side is built with JS, jQuery, Ajax, HTML, CSS, and Bootstraps.

To reduce system complexity, the system is separated into modules. Prepare a test strategy for the system module by module. System testing is performed through module test cases. After finishing module testing, the complete system was tested.

When the system was completed, it met all the client's criteria and met all of the project objectives. Implemented a web-based system with stored data to store and access data from any device and location. The system has tools for handling various types of data and creating appropriate reports based on the client's specifications. Allow their clients, workers, and consumers to interact via email. The system is now more secure for all operations performed on it.

Creating different types of reports based on user levels is quite important for making more effective judgments in this organization. The system history allows the firm manager or administrator to track users' activity in the system via login to logout. It is critical to provide distinct user levels and rights to clearly identify the users' workspace of the system. The system minimizes the user's involvement in each action. In addition, to maintain high activity efficiency, limit typing and data entry. Most of the time, system automated features offer for potential actions to eliminate user engagement. The system is powered by a well-designed database. Databases are designed with the goal of managing vast amounts of data efficiently and effectively in the future.

In this organization, the new system replaces manual systems with computerized systems, paper-based systems with computer web-based systems, and most manual processes with automation. The system archived the client's needs once they had been completed functionally and non-functionally. Considering user comments, overall system test results show that the system met the customers' expectations and successfully fulfilled the client satisfaction level.

6.1.2 Problems Encountered

The requirements collection phase of the project begins to get increasingly complex. It takes extra time to interview people, analyze documentation, validate scopes, and explicitly define system limits. In addition, additional weight was assigned to clearly specify requirements to reduce the project's failure rate.

Analyze and test essential project functionalities that have been prioritized. Because the success of the software project is dependent on the effective completion of the project's important functions. Some novel real-world activities are converting to system processes and developing them by confronting and addressing challenges.

6.2 Lesson learned

This project provided valuable practical experience, from the original business study through the effective delivery of the sales and inventory Management System.

- Using the MVC software design will provide you with even more information in the future.
- Throughout this project, several project management ideas studied in the BIT degree course were effectively used.
- Using several industry-level software frameworks and tools for the creation of this system provided a comprehensive technical experience.
- Effective interaction with the Owner has become critical in understanding their business needs and expectations. As a result, valuable expertise in effective communication was obtained.
- Time management abilities were also developed by successfully completing this assignment within the UCSC's pre-defined timeline.

6.3 Suggestions for future works

As a future enhancement to this system, a personnel management module may be created. This will be beneficial because CODE Technologies employs many people. This module can contain features such as attendance tracking, payment of salaries management, and short leave management.

References

[1] zoho.com, "zoho inventory Management System," zoho.com, 4 March 2023. [Online]. Available: https://www.zoho.com/inventory/inventory-software-demo/#/vendors. [Accessed 4 6 2023].

[2] "Zap Inventory Management System," [Online]. Available: https://www.zapinventory.com/. [Accessed 18 12 2022].

[3] Wikipedia, "Wikipedia.org," Wikipedia.org, 4 6 2023. [Online]. Available: https://en.wikipedia.org/wiki/Rational_unified_process#/media/File:Development-iterative.png. [Accessed 4 6 2023].

[4] "MDN Web Docs," Mozilla, 18 12 2022. [Online]. Available: https://developer.mozilla.org/en-US/docs/Web/HTML. [Accessed 18 12 2022].

[5] "Bootstrap," [Online]. Available: https://getbootstrap.com/docs/5.3/getting-started/introduction/. [Accessed 18 12 2022].

[6] "PHP Documentation," PHP, 18 12 2022. [Online]. Available: https://www.php.net/manual/en/. [Accessed 18 12 2022].

[7] "MySQL Documentation," SQL, 18 12 2022. [Online]. Available: https://dev.mysql.com/doc/refman/8.0/en/. [Accessed 18 12 2022].

[8] "JQUERY AJAX," 18 12 2022. [Online]. Available: https://api.jquery.com/category/ajax/. [Accessed 18 12 2022].

[9] "JQuery," 18 12 2022. [Online]. Available: https://api.jquery.com/. [Accessed 18 12 2022].

[10] "Apache Foundations," Apache.org, 18 12 2022. [Online]. Available: https://httpd.apache.org/docs/2.4/. [Accessed 18 12 2022].

[11] "Javatpoint," javatpoint.com, 18 12 2022. [Online]. Available: https://www.javatpoint.com/unit-testing. [Accessed 18 12 2022].

[12] "Javatpoint," javatpoint.com, 18 12 2022. [Online]. Available: https://www.javatpoint.com/integration-testing. [Accessed 18 12 2022].

[13] "Javatpoint," javatpoint.com, 18 12 2022. [Online]. Available: https://www.javatpoint.com/system-testing. [Accessed 18 12 2022].

[14] "Javatpoint," javatpoint.com, 18 12 2022. [Online]. Available: https://www.javatpoint.com/acceptance-testing. [Accessed 18 12 2022].

Appendix A – System manual

A.1 Hardware and Software Requirements

Hardware Requirements

Following Table A.1 shows Hardware Requirements

Hardware	Minimum Requirements
Computer Processor	Core i3 6th generation processor
Motherboard	H110 Chipset Motherboard
RAM	8GB DDR4 2133MHz Speed
Storage Device	128 SSD 6Gbps Internal storage freed Space up to 5GB
Network	Stable Internet Connection
Graphics	AMD- HD6770 1GB Graphic Card

Table A.1 : Hardware Requirements

Software Requirements

Following Table A.2 shows Software Requirements

Software	Minimum Requirements
Operating System	Windows 10 or Windows 11
IDE	Microsoft Visual Code, PHP-Storms or any other IDE supported PHP Language
Web Server Software (WAMP)	WAMP Software 3.3.0 PHP Version 8.1

Web browser	Mozilla Firefox, Google Chrome or any other JavaScript enabled browser
SQL Server	MYSQL Server and MYSQL Workbench

Table A.2 : Software Requirements

Program Installation

Step 1 – Install Microsoft Redistributable Kit

Download the Microsoft Visual C Redistributable Kit Zip file by visiting below URL. It contains BAT file no need to worry about find and install separate version files. This BAT file automatically installs all Redistributable version 2005 to 2022. Microsoft Redistributable Kit is essential for run WAMP Server Software. If its skip this step, you unable to run WAMP Server software.

https://www.techpowerup.com/download/visual-c-redistributable-runtime-package-all-in-one/

Step 2 – Install WAMP Server Software

Download the WAMP Server software by given below URL. After Downloading the software, run the setup file and follow the instructions given by the WAMP software, then you can successfully install this software.

https://sourceforge.net/projects/wampserver/postdownload

Once completed the above process, open the WAMP Application in computer desktop. After opening the application, you can see nothing happen, then you can see small icon on right hand side in taskbar. Then click the small icon, then you can see WAMP Server Menu. Figure A.1 shows the WAMP server control panel icon shown in the taskbar.

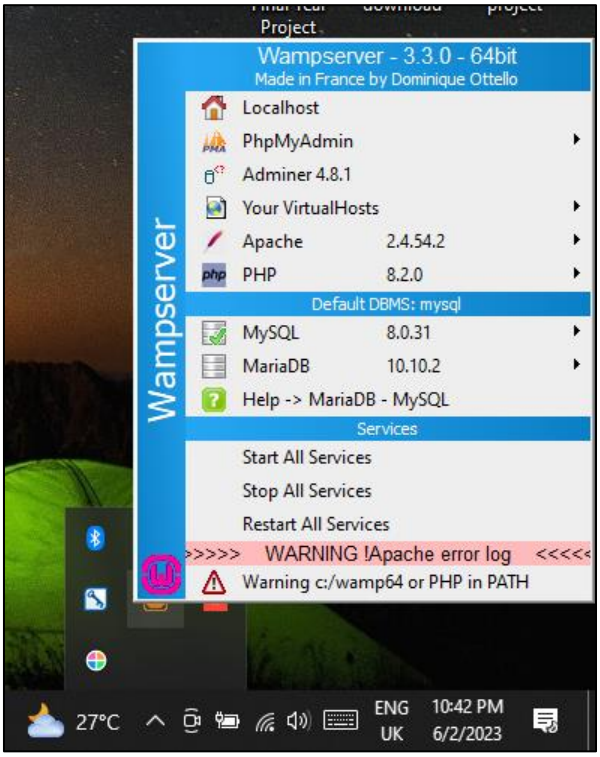


Figure A.1 : WAMP Software

Step 3 – Install MYSQL Server Software

Download the MySQL Server and Workbench Software by visiting below URL. After Downloading the software, run the setup file and follow the instructions given by the MYSQL Server software, then you can successfully install this software.

https://dev.mysql.com/downloads/workbench/

Figure A.2 shows the user interface of MYSQL Workbench software.

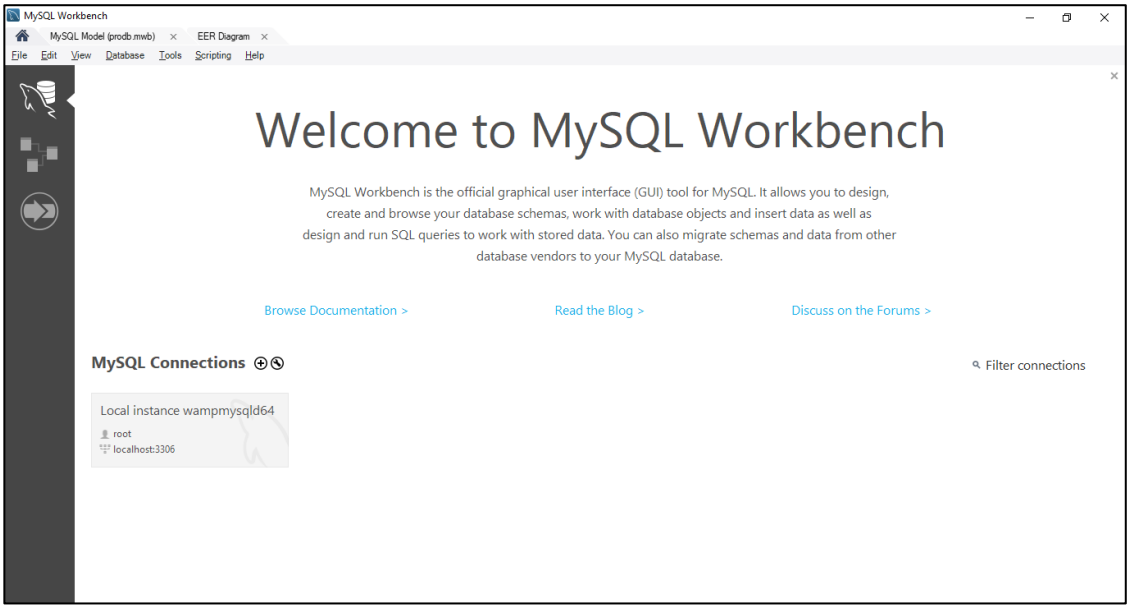


Figure A.2 : MYSQL Workbench Software

Step 4 – Install Web Browser

Download and install the JavaScript enabled Web browser visiting below URL as you wish.

Mozilla Firefox: - https://www.mozilla.org/en-US/firefox/new/

Google Chrome: - https://filehippo.com/download_google-chrome/

Opera Mini:- https://www.opera.com/computer/thanks?ni=stable&os=windows

Step 5 – Setup the database

Open the MYSQL Workbench Software

In the MYSQL Workbench click the MYSQL Local Connection tab available in your computer. After Click this tab this software asks database Username Password. Then give Username as root and Password as 1234. Figure A.3 shows above scenario where enter the password of the MYSQL Server.

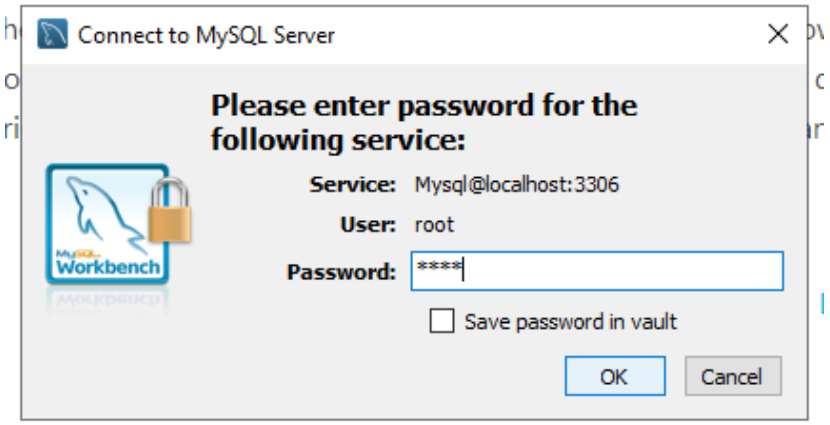


Figure A.3 : MYSQL Workbench Software 1

After creating a database name, it as V3.

After creating a database, in the schema category click the Data import menu. After clicking the menu, appears data import wizard select the SQL file in the database folder located in the project and click execute/import button.

Step 5 – Copy project files into WAMP project location

Copy project folder into root folder directory of the WAMP server.

WAMP Server Root folder C: /wamp/www

Step 5 – Execute the application.

Open the Web Browser type localhost/v3 and press enter. Then automatically redirect the index page of the system.

Appendix B – User manual

This section contains the user manual required for new users of the Sales and Inventory Management System for computer shop management system for computer shop to understand how the system's primary processes work. This involves logging in, viewing the dashboard, adding a new item, managing suppliers, and making a new purchase. Users will be provided a step-by-step demonstration of how particular procedures are performed, allowing them to quickly become accustomed to the system and do their daily tasks.

Login Page

The user can log in to the system by providing the proper login information. When the login button is pushed, the user is taken to the dashboard. The system will display an error message if the user enters an invalid username or password.

Figure B.1 shows the developed system login page.

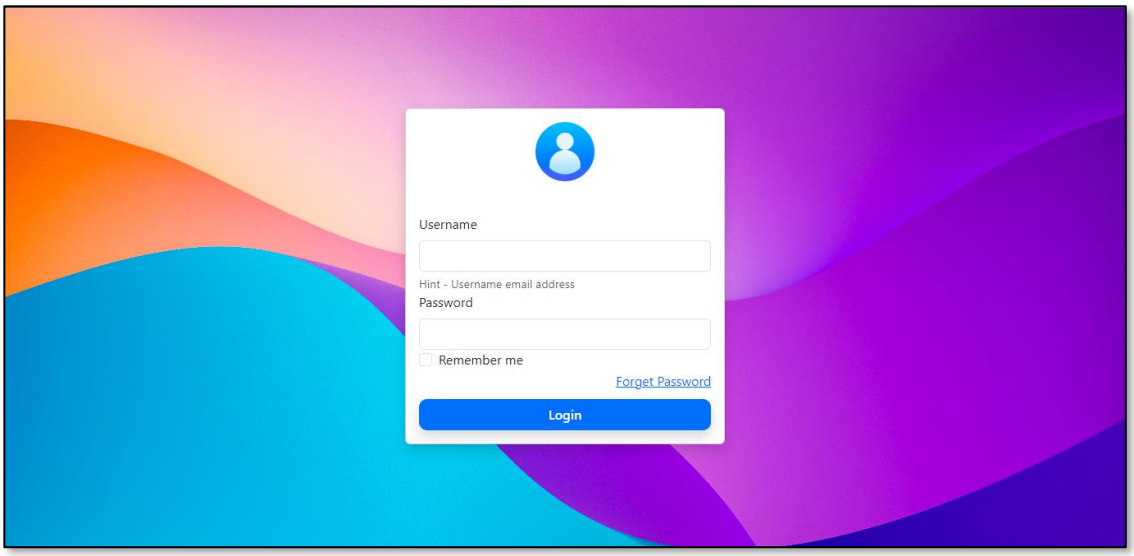


Figure B.1 : System login page

Figure B.2 shows the success message when user correctly enter the username and password.

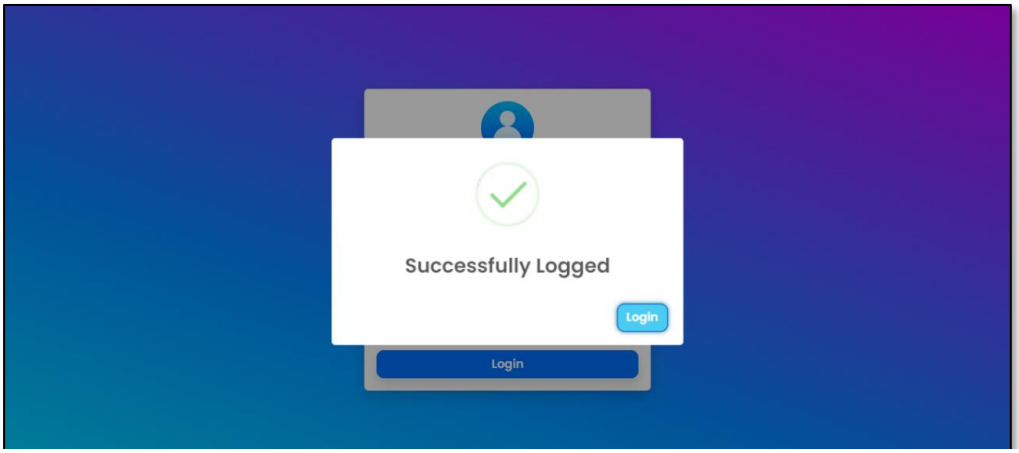


Figure B.2 : Login alert 2 - Success Login

Figure B.3 shows the popup message when the user entered email doesn't exist then show user does not found.

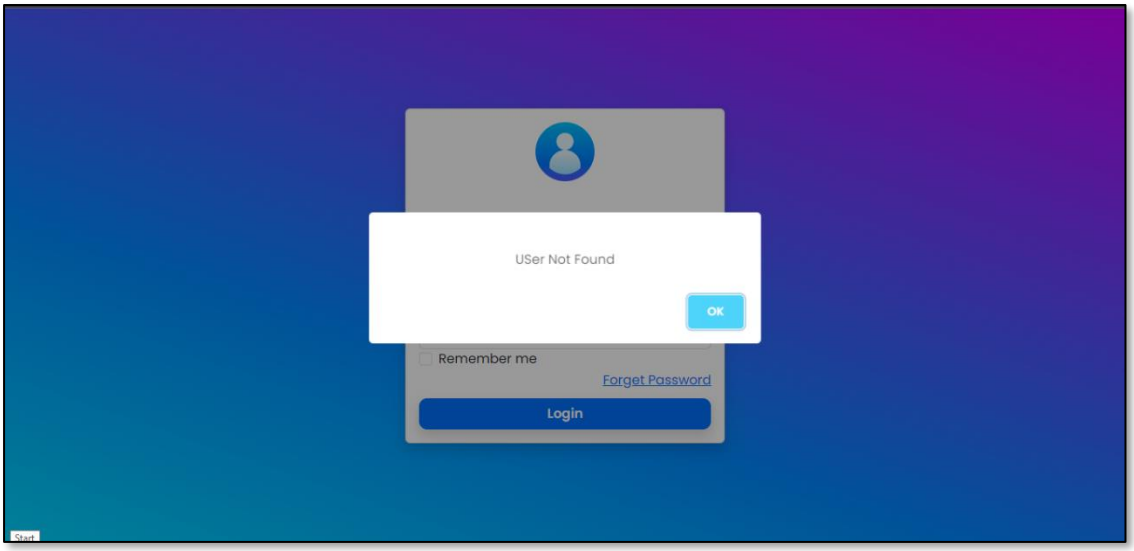


Figure B.3 : Login alert 2 - Invalid Login

Dashboard

When a user logs in, an intuitive dashboard is shown. The dashboard includes information on the overall number of items, Today's Sales, Categories, overall Purchases, Suppliers, and System Users.

Figure B.4 show overview dashboard calculating number of customers of particular day, total number supplier total number of sales and other etc.

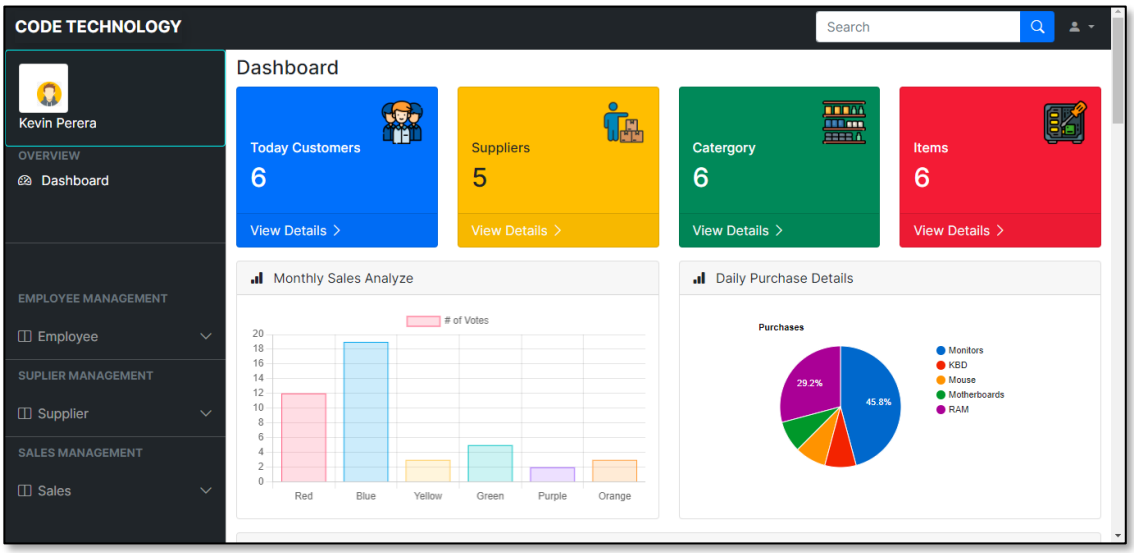


Figure B.4 : Dashboard 1

Side Navigation Bar

An interactive sidebar is offered to help the user navigate the system, and all the important management, including supplier, item, sales, and purchases, can be found here.

Following figure B.5 shows the Side Navigation Bar 1

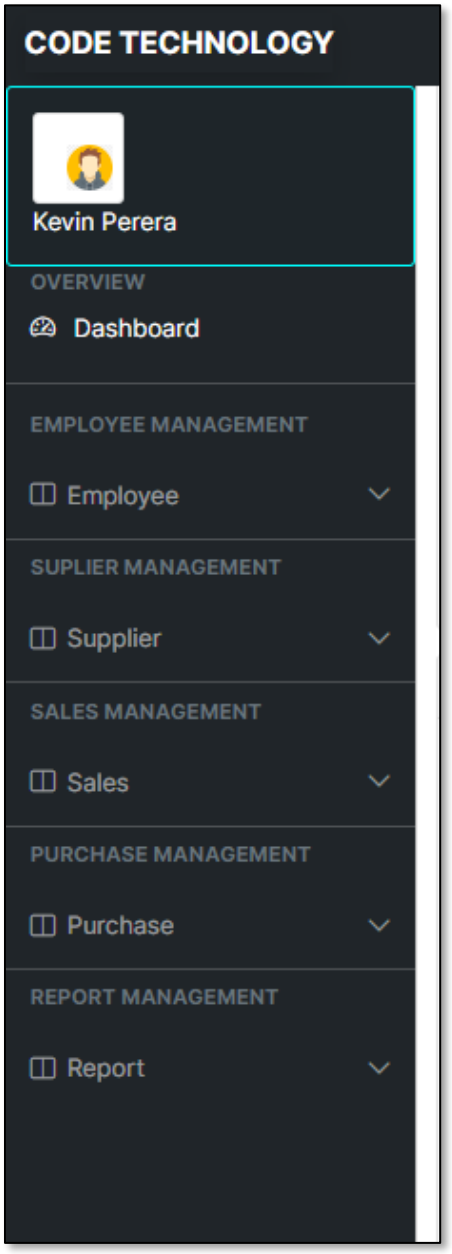


Figure B.5 : Side Navigation Bar 2

Appendix C – Management reports

This section offers a list of the system's numerous reports.

Supplier Report

A report may be created under the Supplier Details view area, which the administrator can print and carry with them. The report may be seen by the administrator, and clicking the print option will begin downloading a pdf file.

An invoice may be created from the system with the required items included. Displayed in Figure C.1 and Figure C.2 are screenshots depicting the overall of active suppliers within this shop.

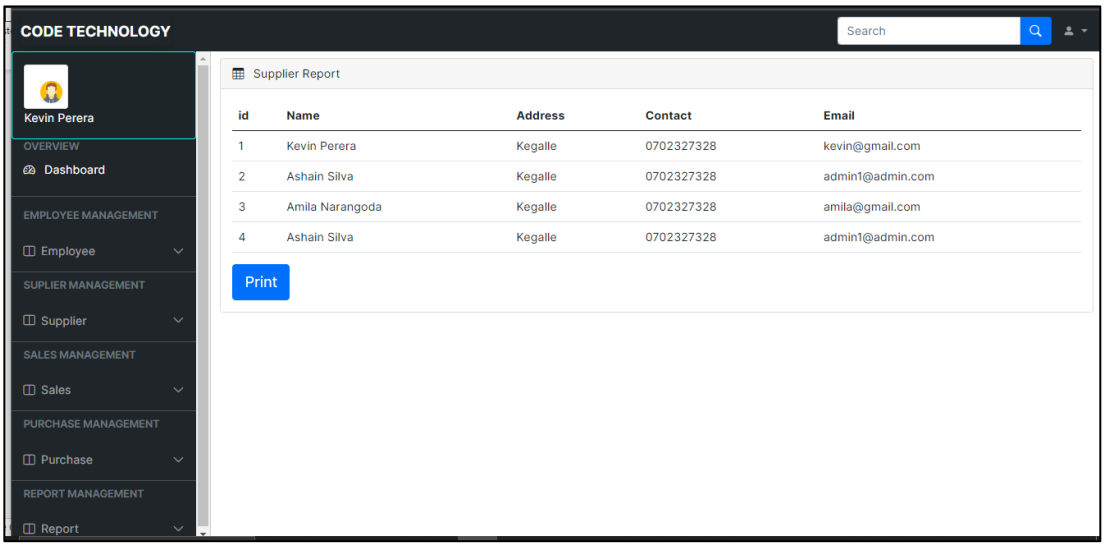


Figure C.1 : Supplier Report

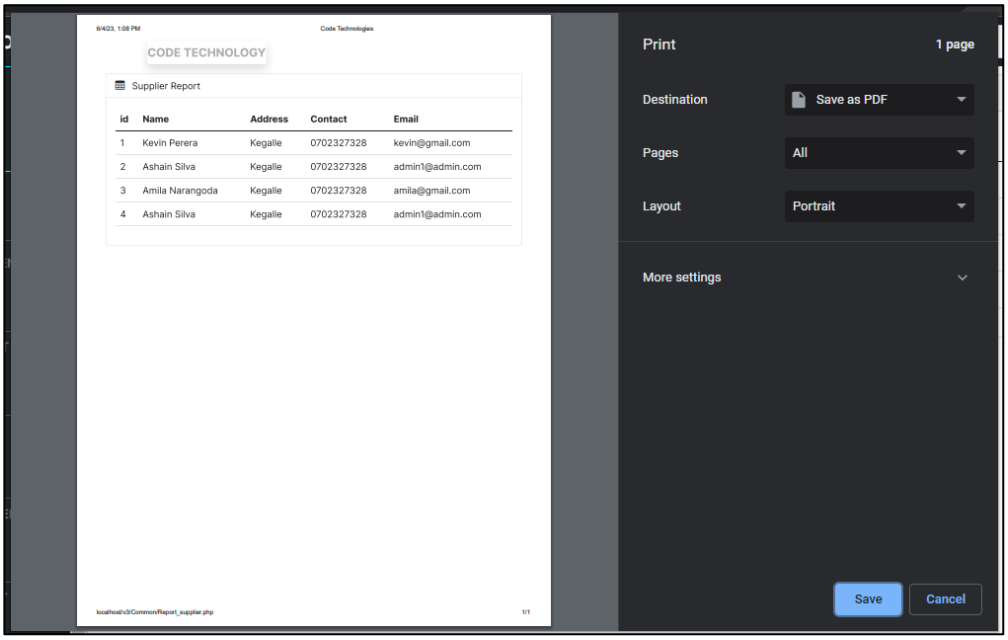


Figure C.2 : Supplier Report Print Preview

Report on Purchase

Within this sales and inventory management system, users with the "admin" privilege can generate comprehensive reports encompassing various purchase types. These reports will encompass aggregate quantities for each purchase of computer parts.

Displayed in Figure C.3 and Figure C.4 are screenshots depicting the all the purchases done in this shop.

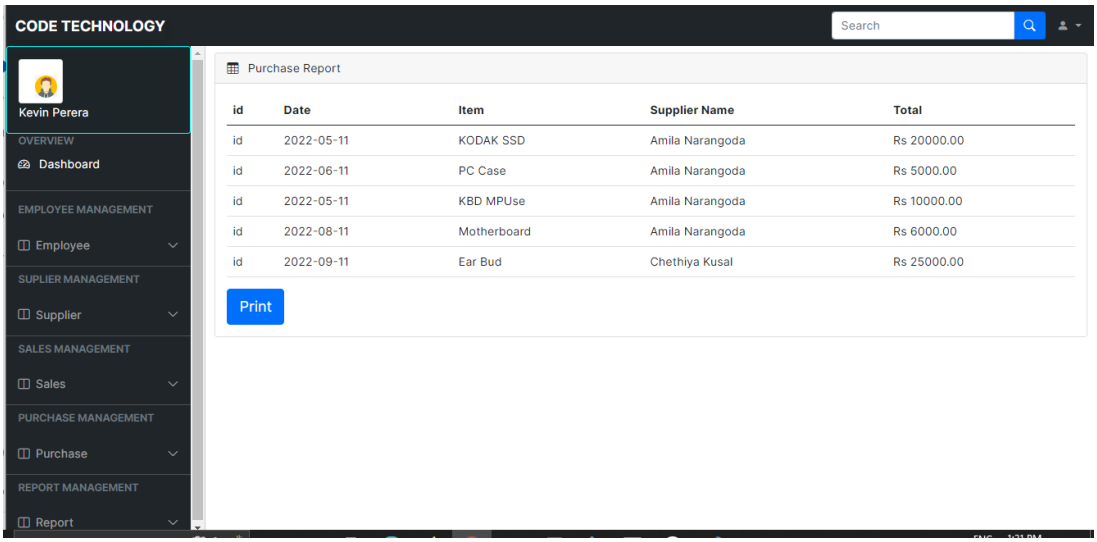


Figure C.3 : Purchase Report

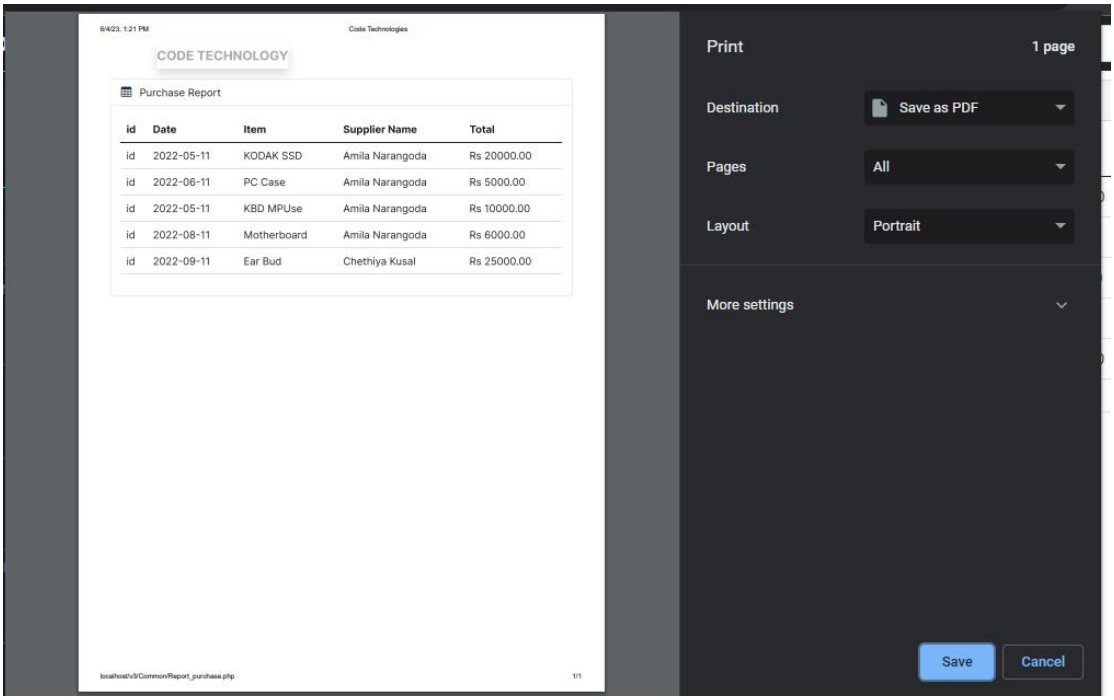


Figure C.4 : Purchase Report print preview.