STOCK CONTROL MANAGEMENT SYSTEM FOR M3 FARMA MARKETING

College of Technology and Allied Sciences
BOHOL ISLAND STATE UNIVERSITY
Zamora, Bilar, Bohol

KIMBERLY C. CURAY
JUNREY O. LIGTAS
LARRY C. QUILARIO

June 2022

STOCK CONTROL MANAGEMENT SYSTEM
FOR M3 FARMA MARKETING

———————————————

A Thesis
Presented to the Faculty of the
College of Technology and Allied Sciences
BOHOL ISLAND STATE UNIVERSITY
Zamora, Bilar, Bohol

———————————————

In Partial Fulfillment
Of the Requirements for the Degree
BACHELOR OF SCIENCE IN COMPUTER SCIENCE

———————————————

Kimberly C. Curay
Junrey O. Ligtas
Larry C. Quilario

June 2022

APPROVAL SHEET

This thesis entitled **"Stock Control Management System for M3 Farma Marketing"** prepared and submitted by *Junrey O. Ligtas, Kimberly C. Curay, and Larry C. Quilario* in partial fulfillment of the requirements for the degree Bachelor of Science in Computer Science has been examined and recommended for acceptance and approval for oral defense.

THE THESIS COMMITTEE
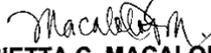
ARLEN B. GUDMALIN, PhD
Dean, CTAS
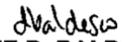
DANIE D. BALDESCO
Thesis Adviser

MAE P. BAS, PhD
Editor

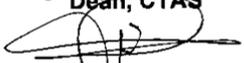**SHEILA G. TABUNO**
Chairperson, DCoS

Approved by the Examining Panel during the Oral Examination conducted on June 02, 2022 with rating __1.9__ .

EXAMINING PANEL

MARIETTA C. MACALOLOT, PhD
Campus Director

ARLEN B. GUDMALIN, PhD
Dean, CTAS

JOEL A. PIOLLO
Panel Expert

DARREL A. CARDAÑA
Panel Member

SHEILA G. TABUNO
Panel Member

Accepted and approved as partial fulfillment of the requirement for the degree Bachelor of Science in Computer Science.

June 02, 2022
Date of Oral Defense

MARIETTA C. MACALOLOT, PhD
Campus Director

ii

iii

ACKNOWLEDGMENTS

First and foremost, we want to thank God for allowing us to participate in the conduct of this study. Everything is possible with Him.

We, the developers, would want to express our gratitude and appreciation to the many people who have provided us with invaluable assistance and enthusiastic participation in helping us complete this study.

Mr. Danie D. Baldesco, Thesis Adviser, for allocating time and wisdom for the researcher to pave the way in bringing the system more productive.

Dr. Arlen B. Gudmalin, Mr. Joel A. Piollo, Ms. Sheila G. Tabuno, and Mr. Darrel A. Cardaña, the panelists throughout the defense, we acknowledge them for sharing their ideas and sincere assistance in bringing this effort to fruition despite the challenges we faced.

Ms. Sheila G. Tabuno, Subject Adviser, for her tireless efforts and support in providing suggestions, advice and guidance.

Dr. Marietta C. Macalolot, Campus Director, for her approval of the project.

Mr. Marino P. Omac, the owner of the establishment and the client, we express our sincere gratitude for giving his utmost support towards the development of the system and completion of the project.

Last but not least, we express our gratitude to our family members who have contributed to the successful execution of this project.

THE RESEARCHERS

ABSTRACT

The study aimed to design and develop a Stock Control Management System for M3 Farma Marketing in Riverside, Bilar, Bohol. Personal interviews with guide questions, real involvement, and observation of the flow of activity were used to collect enough data and information for the study. The researchers discovered that the current system uses manual process in the establishment's stock control management. The manual system presents problems including data and record loss, time consuming transactions, and supply shortages brought on by lack of real-time stock availability monitoring. The researchers developed a solution in response to the identified problems and issues that addresses the drawbacks of the manual system, secures information, and provide an easier process of stock control management. Based on the identified need, Stock Control Management System was designed using the modules administration, acquisition, release and reports to cover the operations of the establishment. Furthermore, the system would be developed using an offline system or a window-based system. The rapid application development model is chosen when designing the system. This technique would carry out the development stage in accordance with the project's scope of work. During the testing and implementation, standard system usability questionnaire was used for evaluation. The system usability questionnaire's average weighted mean is 6.2, which means "Agree." The interpretation indicates that the implementation of the system based on the system's usability result is possible. Thus, implementation of the Stock Control Management System for M3 Farma Marketing is highly recommended.

TABLE OF CONTENTS

Chapter

1   THE PROBLEM AND ITS SCOPE

THE PROBLEM

RESEARCH METHODOLOGY

LIST OF FIGURES

LIST OF TABLES

LIST OF PREVIEWS

Chapter 1

THE PROBLEM AND ITS SCOPE

Rationale

According to the study of Mrabu (2020), the issue of inefficient inventory management systems has been noticed globally. It makes sense that many organizations in developing nations have not yet given it the importance it merits. Consequently, the inability to regulate inventory may be the cause of poor-quality output. Sales business management has to give importance to stock management to supply the need of its customers.

Inventory management relies on having the appropriate stock on hand at the appropriate time to meet consumer demand (Jenkins, 2022). Companies with best-in-class inventory management techniques avoid making educated guesses about how much inventory to purchase and maintain a steady flow of raw materials, items that are still being manufactured, and finished goods as it moves from the manufacturing process to the consumer through a variety of distribution channels.

Many small and medium-sized businesses in the Philippines according to Rosario, have created their inventory controls using standard inventory control management (Rosario, 2022). The analysis of small and medium-sized businesses in the Philippines revealed several process changes to satisfy customer needs. The integration of a stock control system in their business would maintain inventory inflows and outflows by making better stock management

decisions, managing warehouse capacity, avoiding overspending, and cutting costs by making the best use of capital. Good inventory management balance the amount of inventory that enters and exits to maximize profitability. The application of stock management software is very significant for effective inventory control.

M3 Farma Marketing is a small pharmaceutical business that is gradually growing and is solely managed by the owner. The business manually does its physical counting of stocks and is dependent on the manual listing of products and inputs of data. This manual system may cause duplication of documents and work redundancy. No proper records of stock released are observed. It also occurs that there is no proper listing of products and in some instances, lost items are ignored. Additionally, the business stores its stocks across multiple warehouses and this added complexity to managing the stocks efficiently and effectively.

The Stock Control Management System would serve as a comprehensive solution for the business's stock control and management needs. Not only did the owner have oversight and control over the goods he had on hand, but he also used this to track and monitor the needs for both quality control and stock management. The stocking and product management of items is organized and eases the manual labour work the owner has to do. A better insight into which items are most popular allows for enhanced management of stock and quality control. This would help the owner in enhancing the operation of his business.

Literature Background

The development of the computerized system was anchored on Article XIV Section 12 of the Republic of the Philippines Constitution of 1987, which states:

> *"The State shall regulate the transfer and promote the adaptation of technology from all sources for the national benefit. It shall encourage the widest participation of private groups, local governments, and community-based organizations in the generation and utilization of science and technology."*

According to this article, the state should promote the adaptation of technology to bring greater convenience and benefits to people or organizations in the generation and use of science and technology. The Stock Control Management System is a computerized system developed to help promote the integration of technology in M3 Farma Marketing to enhance its operation.

The study of Ogbo & Ukpere (2014) indicated that a key strategy for achieving organizational success is flexibility in inventory control management. It was discovered that businesses gain benefits from inventory control management through simpler material storage and retrieval, increased sales effectiveness, and lower operating costs. The study also discovered a connection between operational viability, the usefulness of inventory control management in the organization's customer-related problems, and the use of cost-effective techniques to improve the organization's return on investment. One of the areas in which management of any firm should develop capability is effective inventory control management. It is recommended that organizations should adopt the inventory-keeping method that best suits their operations.

According to Tom Jose (2013), every firm needs inventory to keep its operations functioning smoothly. It connects the distribution and production processes. In the majority of businesses, the largest portion of current assets and working capital is invested in inventories. Therefore, having good inventory control and management is crucial. Inventory management aims to reduce investment in stocks while simultaneously guaranteeing the availability of goods in appropriate quantities as and when needed.

According to Pandya & Thakkar (2016), inventory management is the activity of developing and managing the inventory level of raw materials, work-in-process inventory and finished goods to ensure that there are enough supplies available and that the cost of the stocks is kept to a low. Inventory management is important for making an organization successful and efficient. Organizations deal with a variety of cost-related inventory, such as holding costs, ordering costs, and shortfall costs. The increase in each one brought on by a lack of an inventory control system would have a detrimental effect on the company's profitability. Therefore, effective inventory control allows firms to save a significant portion of their overall investment in inventories.

According to Milano (2020), the deterministic continuous review inventory theory is the one put into practice by most retailers. This essentially means keeping items in stock and ordering more when levels begin to drop. If it takes a week for items to ship to a store, there is a need for an enough inventory to last a week. Otherwise, you will have to turn customers away if the item is out of stock. Deterministic models of inventory control are used to determine the optimal

inventory of a single item when demand is mostly largely obscure. Under this model, inventory is built up at a constant rate to meet a determined or accepted demand. The researchers cited this theory because the created system includes processes for monitoring in-demand items and figuring out when to reorder them.

Some related open-source systems that might be used as a reference include the following:

1. STOCK MANAGEMENT SYSTEM (Ros, 2022). The idea behind the project is to manage stock products, together with their quantities and prices. When it comes to the project, a user must go through a login process to gain access and utilize its capabilities. The CRUD operations required for managing stock items are easily performed by a user. The Stock Management System project aids the user in the quick and simple management of stock items, quantities, and amounts. The developed system includes the log-in and managing stock products. The developed system which is supported by a relational database also uses the Create-Read-Update-Delete procedures.

2. BRONZE INVENTORY SYSTEM (Executive Property Investment, 2002). Barcoded product labelling, order transfers across sites, and a variety of reports for business management are all supported by the Bronze Inventory System. Though the developed system does not support barcoded product labelling and transfer of orders, the developed system supports the integration of quality control in the system. The owner can view notifications on near expiry and expired products.

3. SILVERNET INVENTORY SYSTEM (Executive Property Investment, 2002). SilverNet is a web-based inventory system that allows seeing information in any place about stocks the user registered. An internet connection is all that is required. The user can monitor the receiving and shipment of products, client payments, and product transfers between warehouses. The developed system supports a client-server network to access resources from a central computer called a server. The client uses the network as a way to connect to the server.

4. FUZZY INVENTORY CONTROL SYSTEM (Phruksaphanrat & Tanthatemee, Thailand, 2012). The study suggests using a fuzzy logic control to handle supply and demand uncertainty in a continuous inventory control system. The continuous inventory control system was represented using the fuzzy logic tool, and the demand, supply, order, and reorder points were described using linguistic terms. The main goals were to assess the order quantity and reorder point in each period while accounting for demand and supply uncertainties. The developed system incorporates procedures that involve keeping track of items in demand and determining when to reorder them.

The above-mentioned related studies have an impact on how the established system is configured. The existence of the related systems will serve as the basis of the developers to improve the present system. It will support and direct the developers on how to enhance M3 Farma Marketing's present Stock Control Management in Bilar, Bohol.

THE PROBLEM

Statement of the Problem

The study sought to design and develop a Stock Control Management System for M3 Farma Marketing. It sought to answer the following questions:

1. How are stocks managed in M3 Farma Marketing?

2. What are the problems or constraints that occur in the control management process?

3. What are the ways to improve the efficiency of stock control management in M3 Farma Marketing?

Scope and Delimitation

The study is focused on the stock control management of M3 Farma Marketing, within its timeframe and boundary, the system is designed with the following features:

1) Stock Control – Stock control includes the process of optimizing stock levels in a warehouse to stabilize inventory storage costs while retaining adequate stock.

2) Database – Database for storing and converting data into information to aid in decision-making. This module includes the accounts, stock handling and release.

3) Report – The function that provides electronic reporting to serve the strategic management of the system. It provides a tabular report for the item released and a graphical report for the item/s with the most released.

4) Quality Control – Quality control is a vital aspect of stock control as this affects the safety of customers or the quality of the product. This feature facilitates the checking and monitoring of the expiration date of the products.

Significance of the Study

The study would help the company develop its policy on stock control management system and procedures. The study would be beneficial for the following:

Owner. The owner of the business would have a comprehensive overview of their stock control, be better equipped to satisfy consumer demand, and be able to lower any costs related to holding stock. The system would also help the owner to prevent product obsolescence and spoilage, and ensure the products are never out of stock;

Staff. The staff who handles the release of items can be assured that the transactions are stored in the database. Thus, the staff would gain an advantage upon the implementation of the system as there is smooth and improved record-keeping in the release of the item;

Readers. The study would help the readers understand the actual processing of stock control and management, and view the procedures and other methods related to the system;

Researchers. This project would be used as a reference in case researchers decide to carry out a similar investigation. Researchers while using the system may also see the possibilities of upgrading the system;

Pharmaceutical Businesses. The attempt to get other establishments to adopt automation would use this study as a model and foundation. To give them the idea of upgrading from a manual to a computerized system.

RESEARCH METHODOLOGY

Development Framework

The study's conceptual diagram depicts the study's model, which is based on the input-process-output principle. The inputs are coming from the owner and staff. The processes include acquisition, release, and administration. The output provides decision support to the owner. It would also represent the work of the owner and staff.
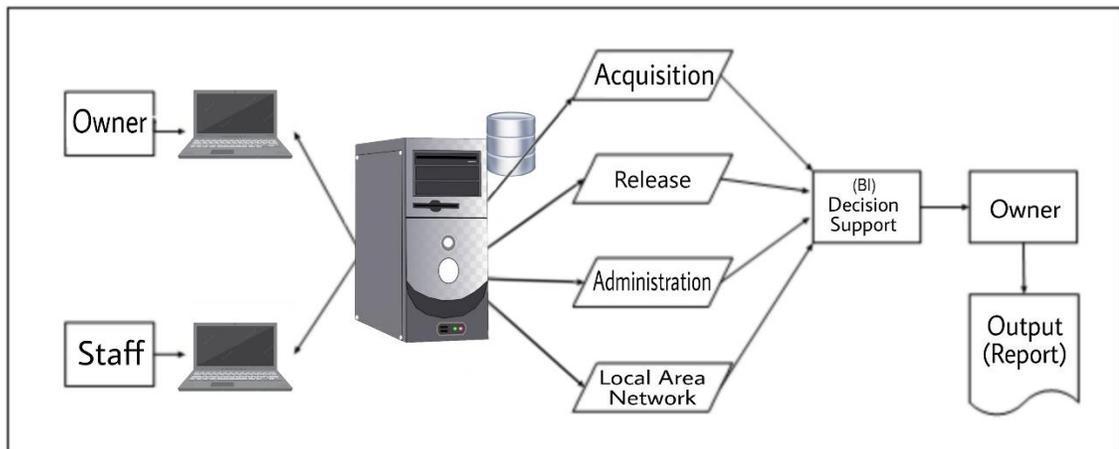
Figure 1. Conceptual Framework of the Developed System

The block diagram of M3 Farma Marketing's Stock Control Management System is shown in Figure 2. It covers the basic functionality of the system that represents the entities who are involved in using and maintaining the system.
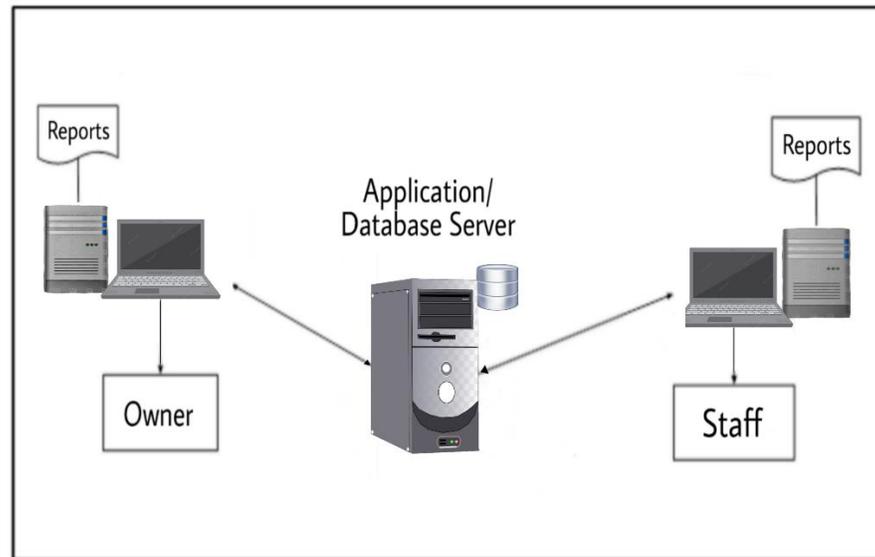
Figure 2. Block Diagram of the Proposed System

The staff and owner of M3 Farma Marketing were the major entities in charge of processes and services relating to individual end-users. The staff's interface can manage the release of items, and the owner's interface has access and authority to all system activities. The reports are generated in a form of tabular and graphical formats.

Development Model and Approaches

The Rapid Application Development (RAD) Model is applied in the development of the M3 Farma Marketing Stock Control Management System. RAD often referred to as adaptive software development is an incremental prototyping approach to software development that end users can produce better feedback when examining a live system. The RAD-based methodology enables the Software Development Life Cycle phases to be adjusted to get specific parts of the system produced rapidly and into the hands of the users.

The advantage of phased development-based methodologies is that the developers can quickly get a useful system into the hands of users, allowing them to gain business value sooner. Furthermore, because users start using the system beforehand, the users are more likely to notice important additional requirements. These are some of the factors that influenced the author's decision to use this methodology in the development process.

The rapid application development processes are divided into four phases:

Requirement Planning. The researchers identified the problem and constraints faced by the business and solutions were proposed. The researchers described the processes for dealing with issues, objectives, scope, risk, and so on. And project activities are scheduled in accordance with the timeline;

Prototype Cycles. During this phase, the client discussed the project requirements that are met at every step of the design process. The developer designs a prototype, and the client and users test it. Then, all the issues are discovered and comments are considered in the nest design version;

Rapid Construction. The primary goal of this phase is to put the design prototypes into beta systems and then into a working model. The developers can then build the final working model more quickly. After final numerous iterations, additional software development tasks are completed and then tested; and

Cutover. This is the implementation phase when the finished product is launched. Until the system is completed, the coding of the project is gradually

initiated. Testing is done to ensure that the project is usable. And the completely

functional software that has been built is deployed at the user's end.

Adopted from https://www.softermii.com/blog/rapid-application-development-model-how-and-when-to-use-it-in-your-software-project
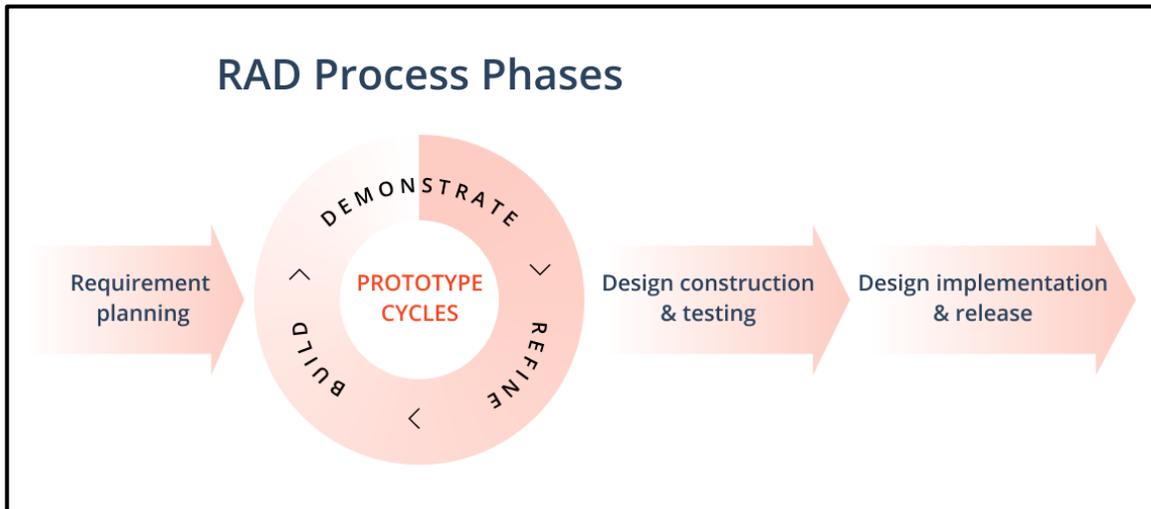
Figure 3. Rapid Application Development Model (RAD) Diagram

Development Tools

The following tools were utilized in the system development process:

Microsoft Visual Basic. IDE used to create console and GUI programs. It has a graphical user interface that enables the developer to manually create program code as well as drag and drop objects into the application.

MySQL. An open-source relational database management system which enables cost-effective delivery of reliable, high-performance and scalable embedded database applications. It was utilized in the programming of the proposed system to connect databases.

Adobe Photoshop 2020. An image editing program is going to be utilized to create a unified layout and theme for the built system.

Wamp Server. A software stack for the Microsoft Windows Operating System. It is used to create a database that can store data from the developed system using MySQL.

Crystal Report. A popular windows-based report writer solution that allows a programmer to create reports and dashboards from a variety of data sources with the least amount of written code. It is used to generate and view reports from the system that has been developed.

Environment and Participants

The research was carried out at M3 Farma Marketing in Riverside, Bilar, Bohol. The respondents were the owner and the two staff in the establishment. The researchers conducted a personal interview with the respondents, who handled the stock control management of the business, and questions concerning the establishment's processes were posed.

Data Collection

The developers obtained authorization from the owner of M3 Farma Marketing to perform the research. Personal interviews with guide questions, real involvement, and observation of the flow of the activity were used to collect enough data and information for the study.

In the system usability adopted by Lewis (1995), questionnaires were used as an instrument to determine the perception of the target users as to the system's acceptability. The target clientele was presented with the system's usability, operations and features.

The system was tested and evaluated to assess its general acceptance and functionality. Furthermore, the system's performance was measured in terms of its usability. The system usability involved two respondents, the owner and the staff of M3 Farma Marketing as shown in Table 1.

Table 1

Summary of Respondents in the System Usability

Respondents	No. of Respondents
Owner	1
Staff	2
Total	3

A questionnaire was presented after the testing and hands-on activities of the owner and staff to measure the level of system acceptability. The respondents were given some time to rate the numbered items in the questionnaire. The guidelines for interpreting the findings in the system usability are shown in Table 2 below.

Table 2

Interpretative Guide of the System Usability

Weight	Range	Description	Interpretation
7	6.4 – 7.0	Strongly Agree	The respondents strongly believe and are confident that the system is very usable

6	5.5 – 6.3	Agree	The respondent believes and is confident that the system is very usable
5	4.6 – 5.4	Tend to Agree	The respondents tend to agree that the system is usable
4	3.7 – 4.5	Neither Agree or Disagree	The respondents are neutral in trusting that the system is usable
3	2.8 – 3.6	Tend to Disagree	The respondents tend not to trust the system is usable
2	1.9 – 2.7	Disagree	The respondents believe that the system is not usable
1	1.0 – 1.8	Strongly Disagree	The respondents are strongly confident that the system is not usable

To determine the acceptability of the system, the average weight mean score was computed to evaluate and assess the system acceptability level using the formula below.

$$WMS = \frac{1f_1 + 2f_2 + 3f_3 + 4f_4 + 5f_5 + 6f_6 + 7f_7+}{n}$$

Where:

WMS = Weighted Mean Score
f_1 = frequency of respondents given a rate of 1
f_2 = frequency of respondents given a rate of 2
f_3 = frequency of respondents given a rate of 3
f_4 = frequency of respondents given a rate of 4
f_5 = frequency of respondents given a rate of 5
f_6 = frequency of respondents given a rate of 6
f_7 = frequency of respondents given a rate of 7
n = total number of respondents
1,2…7= constant (rating to the service provided)

OPERATIONAL DEFINITION OF TERMS

In the course of this research, the following terms are defined conceptually:

Critical Stocks. Stocks that are almost out and need restocking. An exact stock level at which the stock needs to be replenished.

Inventory. This is the complete list of the item that is in the warehouse that the company maintains during its normal course of business.

Manual. The current technique the owner utilize before the system was proposed. The owner essentially just counts the inventory he has. When stocks are getting low, the owner order from their suppliers. It utilizes simple calculations to determine when inventory is running short.

Out of Stock. An inventory state that indicates that a product is currently unavailable for purchase or use. These are the stocks that are already out in the warehouse and need restocking.

Staff. The employee who assessed the release of items in the warehouse.

Stock. The products held in the warehouse available for sale or distribution.

Stock Control. The set of operations performed by the business, aimed at controlling the movement of items through retail environments, is commonly referred to as inventory management.

Stock Control Management System. A stock control system is the mechanism used by the business to efficiently manage the flow of information, resources, and the transportation and storage of its products.

Chapter 2

PRESENTATION OF FINDINGS, ANALYSIS, AND INTERPRETATION OF DATA

Existing Operation and Processes

M3 Farma Marketing utilizes manual methods, with the use of a manual system to keep track of their inventory. Human error is inevitable, and the process and tracing of item losses were impossible as there were no efficient logging systems. Though a manual system is inexpensive, the use of a computerized system helps save time and money.

The research focuses on M3 Farma Marketing stock control management procedures. The owner is in charge of managing and overseeing the entire operation. Together with his staff, each has their roles in the company where the owner can do all processes and activities in his business. The staff's job is to control the available stock in the warehouse and assess its release.

A. **Recording of Delivered Items**

The owner checks the newly delivered stock to make sure the stocks and details in the receipt issued by the supplier are matched before making any price modifications. Then it would be recorded in a record book as product description and unit price.

B. **Inventory and Quality Control**

Now and then, the owner or the staff counts the items in the warehouse. The owner/staff is monitoring for the availability of stocks. Also,

occasionally, the owner/staff individually check for expired and close to expiry products. The inventory and quality control of items are recorded in the record book.

C. Release

The release of items is handled by the staff/owner for distribution to retail outlets; or disposal if the item is expired. The staff would look for available items and prepared them for release. The list of items prepared for release is recorded in the record book.

D. Generation of Reports

The recorded close-to-expiry, expired, critical and out-of-stock items report in the record book is reviewed manually by the owner and the owner then decides what to do afterwards.
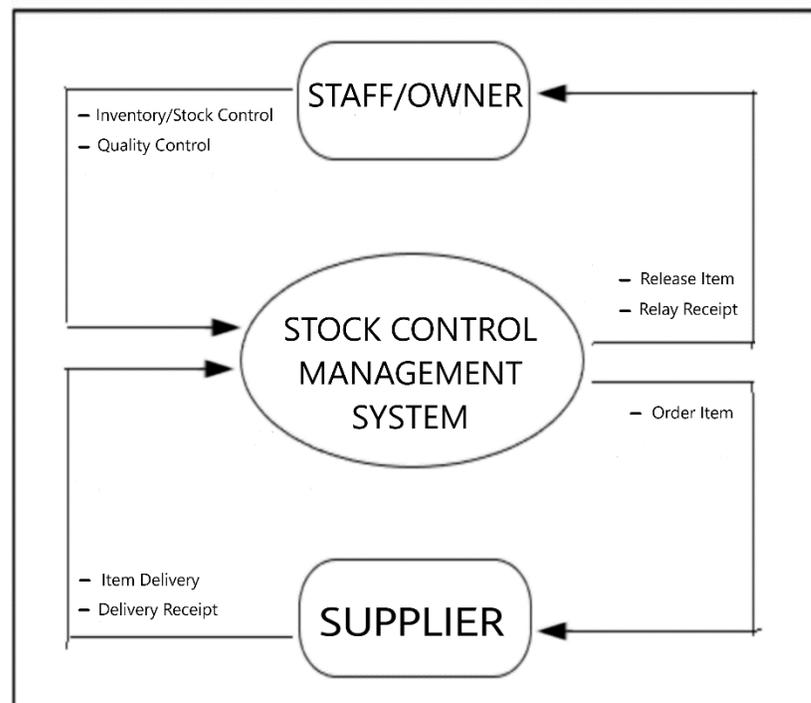
Figure 4. Contextual Diagram of the Present System

Present Event List Diagram

1. Process item recording.

2. Process inventory.

3. Process stock control.

4. Releasing of item.

5. Generates report.
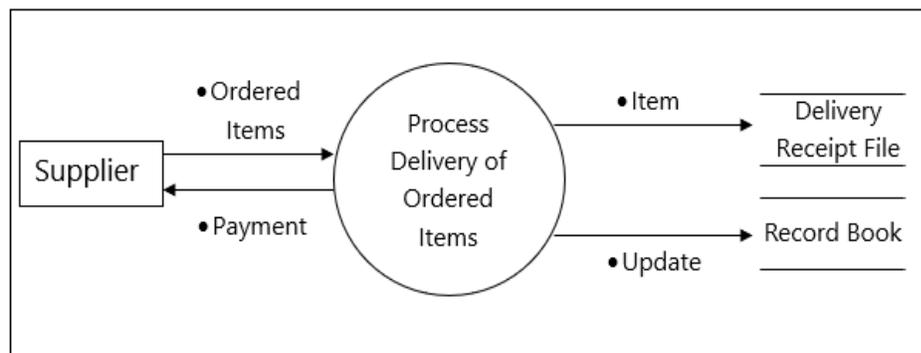
Figure 5. Process Item Recording (Event 1)
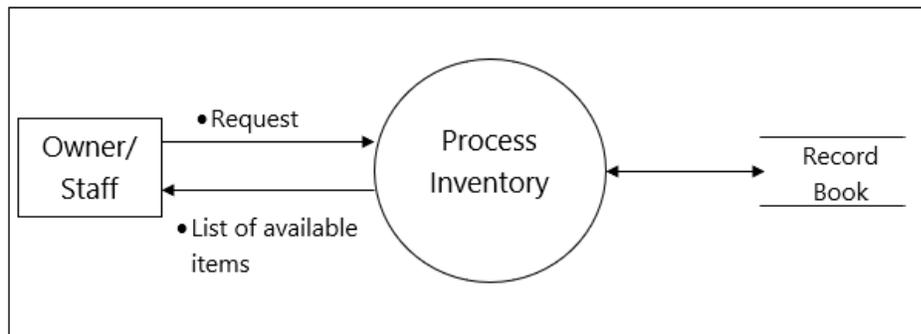
Figure 6. Process Inventory (Event 2)
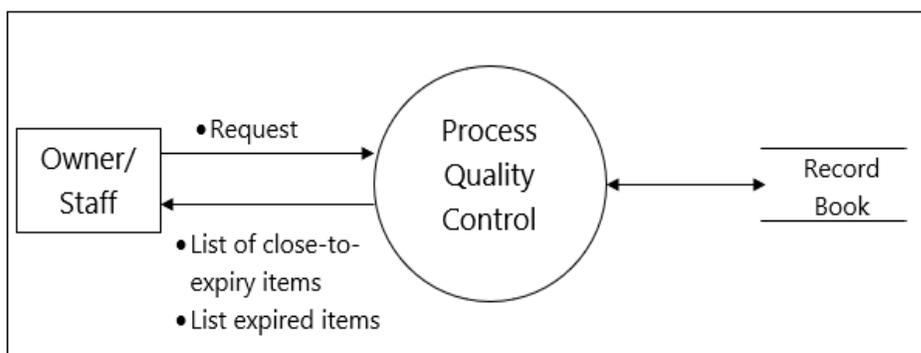
Figure 7. Process Quality Control (Event 3)

Figure 8. Process Release of Stock (Event 4)

Figure 9. Report Generation Process (Event 5)

Figure 10. Top Level of the Present System
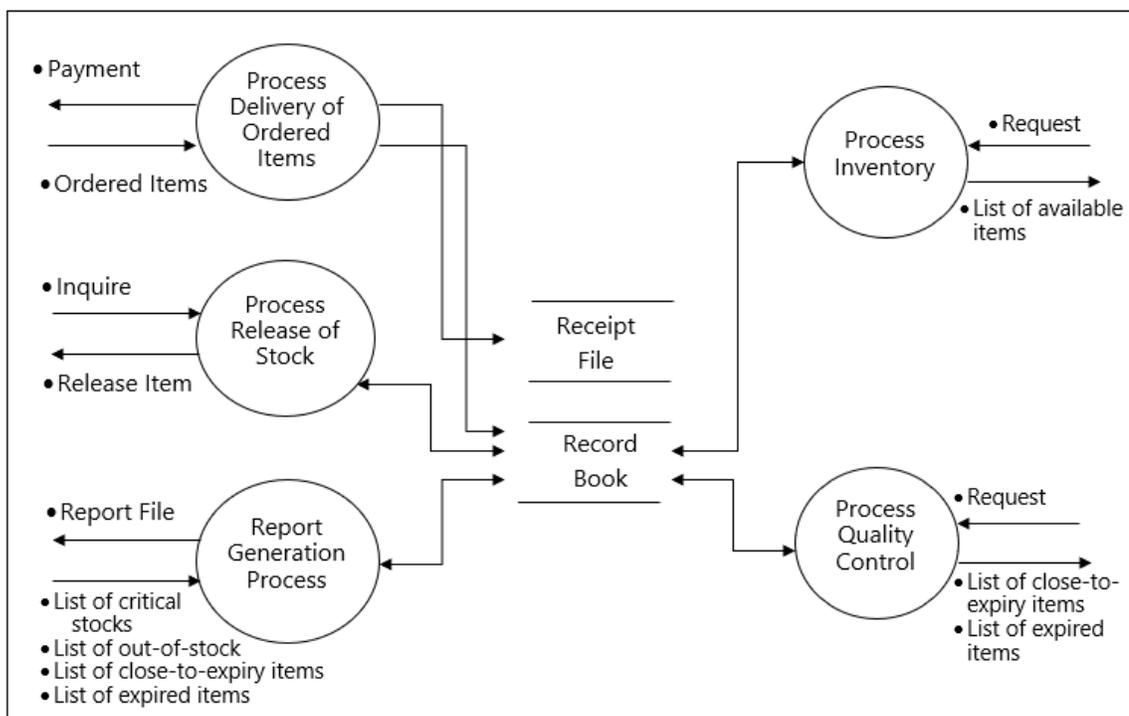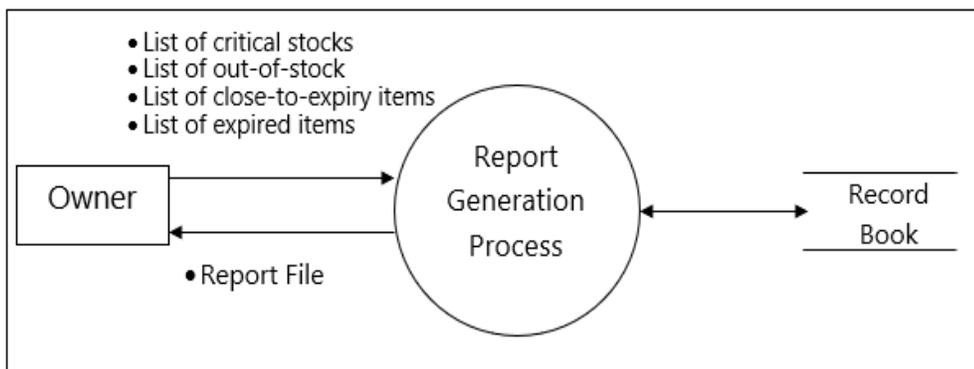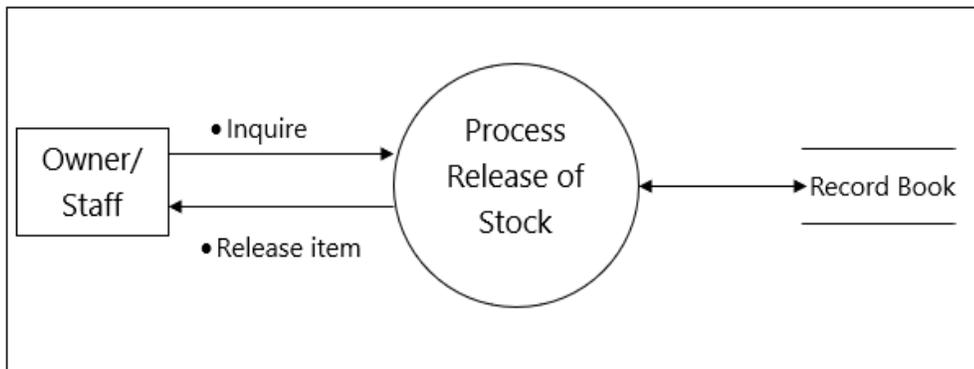
Needs of the Existing Operation

Based on the identified problems with consideration of the existing operations and processes of M3 Farma Marketing, the developers have identified the following needs to facilitate and enhance the processing of the Stock Control Management System of M3 Farma Marketing in Riverside, Bilar, Bohol. The following are the needs of M3 Farma Marketing for a system:

1. There is no appropriate file storage which raises the risk of misplacing or losing files. The system might replace this with a database system with security features.

2. A computerized system to view and handle item records to give conscious to the owner when an item is about to expire or stock is running low. The current stock control procedure is completed manually and human errors are inevitable.

3. Provide a system that records automatically each transaction of staff upon the release of stock.

4. The owner should be able to see the reports of the employee who performed the release and reports of the item with the most release.

5. To accommodate more distribution, stock monitoring must be upgraded.

Stock Control Management System for M3 Farma Marketing

Stock Control Management System delivers a customizable and efficient functionality to assist business people. The system is used to create a product list and several types of reporting for managing the business. The following lists below

are the representation of a user's interactions with the system that shows the connection between the users and the different interactions the users are involved.

A. Administration

To access the individual modules, the owner and staff must log in with their username and password. The owner has access to all of the system's procedures, including the management of accounts and adding of staff. The staff is assigned to stock control and the release of stock.

B. Acquisition

The acquisition comprises of insertion of newly purchased stock as well as stock information. The list of items is saved in the database when the stocks are added. The owner additionally adds the supplier's information, where the owner inputs the supplier's details. The supplier's information includes the company name, address, and contact number.

C. Release

The staff who use the system must log in to an account to select an item for release. The staff also select the customer to whom the stock is released. If the customer is not registered, the staff must record the customer's information first in the customer form. The selected item with its quantity reduced the stocks available in the warehouse and a report is generated on the transaction performed by the user.

D. Reports

A report can be viewed on the items that are close to expiry, expired, critical stocks and out of stock. The transaction performed by the staff on

the release of stock also has a report on which item the staff released and the quantities, lot number, and expiration date of each item. The report is selected on a monthly or yearly basis. A chart can also be viewed on which items have the greatest number of releases and this might help the owner to assess which product to order next.

Use Case Diagram

A TechTarget contributor (2020), states that Use Case Diagram is a way to summarize details of a system and the users within that system. It is generally shown as a graphic depiction of interactions among different elements in a system. Use case diagrams specifies the events in a system and how those events flow, however, use case diagram does not describe how those events are implemented.
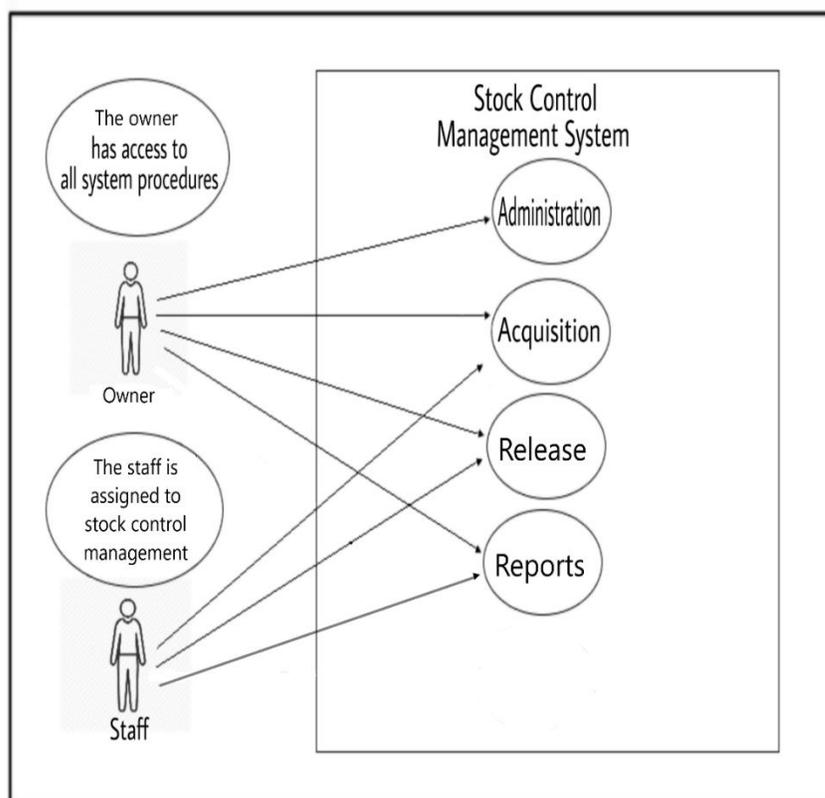
Figure 11. Use Case Diagram of the Developed System

Use Case Narrative

A use case narrative is a largely text-based description of a use case that could be supplemented with decision trees or other easily understood notations. The description should be written in the user's language, and thus provides an important communication tool between developers of systems and intended users. Narratives follow a structured format, typically using a numbered sequence of steps for the main activity accompanied by preconditions, post-conditions, alternative or exception paths, etc. (Dobing & Parsons, 2009).

Table 1. Use Case Narrative – Administration

Administration Process (UC1)	
Level	User Goal
Goal in Context	To use the system.
Primary Actor	Staff/Owner
Stakeholder	Staff/Owner: Wants to Log-in

Preconditions
   1. Must log in using their username and password.
   2. The username and password must match to log in to the system.
   3. The log-in button must be clicked to proceed.	

Triggers	Staff/Owner: Log-in button on the main page.
Success Guarantee	Wait for inputs.

Main Success Scenario
   1. Staff/Owner: Run the software.
   2. Staff/Owner: Clicks the log-in button on the main page.
   3. Staff/Owner: Input the username and password.
   4. System: Validates every field and checks for the user's account.
   5. System: Accepts the user's privilege.	

Extension: None	

Table 2. Use Case Narrative – Acquisition

Staff Recording Process (UC2-A)	
Level	User Goal
Goal in Context	To add staff
Primary Actor	Owner
Stakeholder	Owner: Wants to add information about staff.
Precondition 　　1. Required fields must be filled up completely. 　　2. Should have not recorded existing accounts.	
Triggers	Owner: Clicks the "Staff" menu in the main form.
Success Guarantee	The system saves the data inputted to the database.
Main Success Scenario 　　1. System: Displays the staff information form. 　　2. Owner: Input the staff information. 　　3. Owner: Clicks the "Save" button. 　　4. System: Saves the staff's information to the database.	
Extension: None	

Table 3. Use Case Narrative – Acquisition

Supplier Recording Process (UC2-B)	
Level	User Goal
Goal in Context	To add supplier
Primary Actor	Staff/Owner
Stakeholder	Staff/Owner: Wants to add information of supplier.
Preconditions 　　1. Required fields must be filled up completely. 　　2. Should have not recorded existing accounts.	
Triggers	Staff/Owner: Clicks the "Supplier" menu in the main form.
Success Guarantee	The system saves the data inputted to the database.
Main Success Scenario 　　1. System: Display the supplier's information form. 　　2. Staff/Owner: Input the supplier's information. 　　3. Staff/Owner: Clicks the "Save" button. 　　4. System: Save the supplier's information.	
Extension: None	

Table 4. Use Case Narrative – Acquisition

Item Recording Process (UC2-C)	
Level	User Goal
Goal in Context	To add item/s.
Primary Actor	Owner
Stakeholder	Owner: Wants to add a new item in stock.
Preconditions	
1. Required fields must be filled up completely. 2. Should have not recorded existing stock with the same batch number.	
Triggers	Owner: Clicks the "Add New" option in the "Stock" menu in the main form.
Success Guarantee	The system saves the data inputted to the database.
Main Success Scenario	
1. System: Display the Item's information form. 2. Staff/Owner: Input the Item's information. 3. Staff/Owner: Clicks the "Save/Update" button. 4. System: Save the Item's information.	
Extension: None	

Table 5. Use Case Narrative – Release

Stock Disposal Process (UC3-A)	
Level	User Goal
Goal in Context	To add customer
Primary Actor	Staff/Owner
Stakeholder	Staff/Owner: Wants to add customer's information.
Preconditions	
1. Required fields must be filled up completely.	
Triggers	Staff/Owner: Clicks the "Customer" menu in the main form.
Success Guarantee	The system saves the information in a database.
Main Success Scenario	
1. System: Display the Customer form. 2. Staff/Owner: Input the customer's information. 3. Staff/Owner: Clicks the "Save" button. 4. System: Save the customer's information.	
Extension: None	

Table 6. Use Case Narrative – Release

Stock Disposal Process (UC3-B)	
Level	User Goal
Goal in Context	To view and record item release processes.
Primary Actor	Staff/Owner
Stakeholder	Staff/Owner: Wants to release stock.
Preconditions 1. Select stock for disposal.	
Triggers	Staff/Owner: Clicks the "Release" menu in the main form.
Success Guarantee	The system saves the released item in a database.

Main Success Scenario
    5. System: Display the Release form.
    6. Staff/Owner: Select the item to be released.
    7. Staff/Owner: Clicks the "Release" button.
    8. System: Saves the Item released.	

Extension: None	

Table 7. Use Case Narrative – Reports

Stock Disposal Process (UC4)	
Level	User Goal
Goal in Context	View stock overview and report of staff's release transactions in the system.
Primary Actor	Owner/Staff
Stakeholder	Owner/Staff: Wants to view reports.
Preconditions 1. Log in to view reports.	
Triggers	Owner/Staff: Clicks the "Reports" menu in the main form.
Success Guarantee	The system displays reports of the selected menu.

Main Success Scenario
    1. System: Display the Reports form.
    2. Owner: Select options for filter and display data.
    3. Owner: Clicks the "OK" button.
    4. System: Display reports based on options.	

Extension: None	

Database Design

Database Design according to Peterson (2022), refers to a set of procedures for planning, developing, implementing, and maintaining business data management systems. A well-designed database is simple to manage, increases data consistency, and saves money on disc storage space. The database designer determines how data pieces relate to one another and what information must be saved. The following database tables are used to store the information that is processed in the system, as well as a set of operations that make searching, sorting, saving, and recombination of similar actions easier.

Table 1

Data Structure for Accounts used to Log-In

FIELD	NAME	TYPE	DESCRIPTION
1	UserType	varchar(5)	Account User Type
2	UserName	varchar(30)	Account User Name
3	Password	varchar(20)	Account Password
4	RecoveryQuestion	varchar(50)	Account Recovery Question
5	RecoveryAnswer	varchar(20)	Account Recovery Answer

Table 2

Data Structure for Staff

FIELD	NAME	TYPE	DESCRIPTION
1	ID	int	Primary Key
2	Name	varchar(50)	Staff Name
3	Address	varchar(50)	Staff Address
4	Contact	varchar(40)	Staff Contact
5	Image	blob	Staff Image

Table 3

Date Structure for Supplier

FIELD	NAME	TYPE	DESCRIPTION
1	SupplierID	int	Primary Key
2	BusinessName	varchar(70)	Supplier's Name
3	Owner	varchar(50)	Owner
4	Contact	varchar(40)	Contact

Table 4

Data Structure for Supplier's Bank Account

FIELD	NAME	TYPE	DESCRIPTION
1	SupplierID	int	Foreign Key
2	BankID	int	Primary Key
3	BankName	varchar(50)	Bank Name
4	AccountName	varchar(40)	Account Name
5	AccountNumber	varchar(30)	Account Number

Table 5

Date Structure for Customer

FIELD	NAME	TYPE	DESCRIPTION
1	ID	int	Primary Key
2	Name	varchar(50)	Customer's Name
3	Address	varchar(50)	Address
4	Contact	varchar(40)	Contact

Table 6

Data Structure for Stocks

FIELD	NAME	TYPE	DESCRIPTION
1	ID	int	Primary Key
2	GenericName	varchar(60)	Generic Name
3	BrandName	varchar(30)	Brand Name
4	Manufacturer	varchar(50)	Manufacturer
5	Strength	varchar(15)	Strength
6	DosageForm	varchar(30)	Dosage Form
7	Unit	varchar(10)	Unit
8	NosOrVol	varchar(9)	Nos. or Volume
9	LotOrBatchNum	varchar(20)	Lot/ Batch No.
10	ExpirationDate	date	Expiration Date
11	Supplier	varchar(50)	Supplier
12	OriginalPrice	double(7,2)	Acquisition Price
13	SellingPrice	double(7,2)	Unit Price
14	Quantity	int	In-Stock
15	Description	varchar(300)	Description
16	DateAdded	date	Date Added

Table 7

Data Structure for Cart

FIELD	NAME	TYPE	DESCRIPTION
1	ID	int	Primary Key
2	Quantity	int	Releasing Stock Quantity
3	Unit	varchar(20)	Releasing Stock Unit
4	Description	varchar(300)	Releasing Stock Description
5	LotOrBatchNum	varchar(20)	Releasing Stock Lot/Batch No.
6	ExpirationDate	DateTime	Releasing Stock Expiration Date

Table 8

Data Structure for the Released Stock

FIELD	NAME	TYPE	DESCRIPTION
1	ID	int	Primary Key
2	StockID	int	Released Stock ID
3	AuthorizedRep	varchar(50)	Representative
4	DateReleased	DateTime	Released Date
5	Description	varchar(300)	Description
6	LotOrBatchNum	varchar(20)	Lot/Batch Num
7	Quantity	int	Quantity

ER Diagram
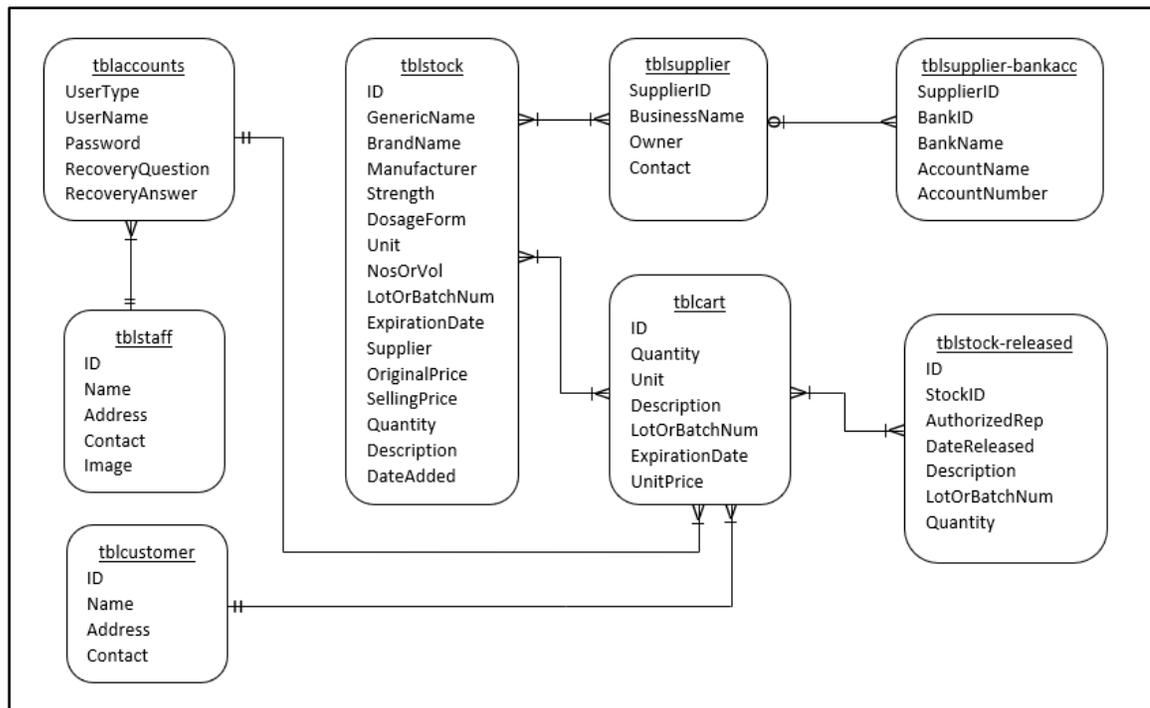
Figure 12. ER Diagram

Program Hierarchy

The hierarchy chart shows the relationship between various modules, according to Busbee (2018). It does indicate that the function's organization is calling on a subordinate function, created by a programmer to aid in the documentation of a program and to give the overall image of the modules utilized in it. The program hierarchy chart contains eight (8) modules, each with the appropriate action to be executed. All of these occurrences contributed to the researchers' analysis, design, and development of the system.
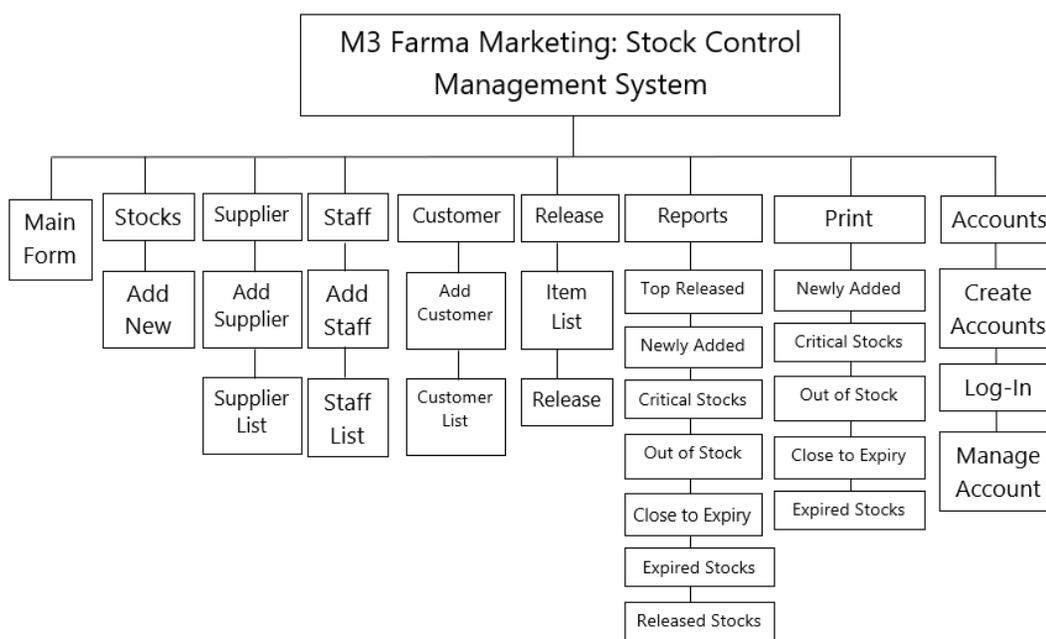
Figure 13. Program Hierarchy of the Developed System

Functional Requirements

A functional requirement according to Martin (2022) describes the service that the software must provide. It refers to a software system and its component. A function is nothing but the inputs, behaviour, and outputs of a software system.

Calculation, data, manipulation, business process, user interaction, or any other specific functionality defines what function a system is likely to perform.

Furthermore, A functional software requirement aids in capturing the system's intended behaviour. It describes the actions taken on each screen. The logic for handling data should be programmed into the system. It should include descriptions of system reports or other outputs, as well as detailed information about the system's workflows. It should explicitly describe who are authorized to produce, alter, and delete data in the system, as well as how the system meets applicable regulatory and compliance requirements.

The following features are based on the existing standard requirements of an assessment tool, with the respondent's agreement and collaboration.

Log-in Authentication Process (Administration)

FREQ 1: Access to the developed system username, password, and user level should be matched

FREQ 2: Access to the system must be password protected with hidden characters.

FREQ 3: The system should unlock features for each user's privileges when the log-in credentials are correct.

Changing the User's Password (Administration)

FREQ 4: The system should be able to recover the user's password in the log-in process when forgotten.

FREQ 5: The system should be able to change the user's password when the two-confirmation password field is matched.

FREQ 6: The system should accept the password in the log-in process.

Recording of Staff's (Acquisition)

FREQ 7: The system should allow the owner to add employees' information and create their respective accounts.

FREQ 8: The system should check for existing employee records.

FREQ 9: The system should accept/save the employee's information.

Recording of Supplier's (Acquisition)

FREQ 10: The system should allow adding supplier information.

FREQ 11: The system should check for existing supplier records.

FREQ 12: The system should accept/save the supplier's information.

Recording of Item/s (Acquisition)

FREQ 13: The system should allow user to add/register item/s information.

FREQ 14: The system should have monitoring of stocks to avoid shortages.

FREQ 15: The system should check for existing item records.

FREQ 16: The system should update the quantity for the existing item.

Inventory

FREQ 17: The system should allow the users to search for available item/s.

FREQ 18: The system should associate with retrieving, sorting, filtering and displaying the specific data.

Recording of Customer's (Release)

FREQ 19: The system should allow adding of customers' information.

FREQ 20: The system should check for existing customers' records.

FREQ 21: The system should accept/save the customer's information.

Release

FREQ 22: The system should allow the owner/staff to search for available items.

FREQ 23: Automatically displays the available item.

FREQ 24: The system should quantify the total amount of stock/s released.

FREQ 25: The system should save the release transaction including the user, and the date and time to their respective tables.

FREQ 26: The system should provide a report of the released stocks.

FREQ 27: The system should allow cancelling of item releases.

Reports

FREQ 28: The system should allow the owner to view the release transaction performed by users.

FREQ 29: The system should associate with searching, sorting, filtering and displaying the specific data.

FREQ 30: The system should allow the deletion of records.

FREQ 31: The system should allow the owner to view and print reports.

FREQ 32: The system should allow the owner to generate the top released item report in graphical form.

User Account Management

FREQ 33: The system should allow the admin to add accounts.

FREQ 34: The system should allow the admin to modify accounts.

FREQ 35: The system should allow the admin to delete accounts.

Administrative Log-out Process

FREQ 36: The system should execute the removal of the logged-in account

and reset user privileges.

Non-Functional Requirements

The definition of a non-functional requirement from a ReQtest author (2020) explains that it essentially specifies how a system should behave and that it is a constraint upon the behaviour of the system. It includes all criteria that aren't addressed by the functional requirements. It establishes criteria for judging a system's operation rather than specific actions.

Non-functional requirements have an impact on the user experience since it defines a system's behaviour and features. The system would be easier to use and perform better if it is adequately defined and executed. As product attributes, non-functional requirements focus on user expectations and satisfaction. On the following page, you'll find the quality attributes:

NFREQ 1: The developed system should be adaptable to the available technology

that the establishments provide.

NFREQ 2: The developed system should be accessible at all times.

NFREQ 3: The developed system must be user-friendly.

NFREQ 4: The developed system should maximize the productivity of the clientele.

NFREQ 5: The developed system should achieve reliability for the end-users

NFREQ 6: The developed system must sustain high-security measures.

Test Case

As defined by Brush (2020), a test case is a set of actions done on a system to check if it satisfies software requirements and functions correctly. A test case is used to assess whether various aspects inside a system are working as intended and to ensure that the system meets all applicable standards, guidelines, and customer requirements. The act of writing a test case can also aid in the discovery of system problems or defects.

These are the scenarios that were tested during acceptance testing. In each test case of the developed system, the test plan is to allow users to use the system and follow the instructions. To be regarded as successful, the system must produce the intended outcome in each test case. Each test case's specifics are as follows:

Log-in Authentication Process (Administration)
Test Case 1

Module: Log-in Form (Admin Account)

Severity 1

Instruction:

1. In the log-in form, enter the username and password.
2. Click the "Log-in" button.

Expected Result:

    1. The owner can access all the modules of the system.

Test Case 2

Module: Log-in Form (User Account)

Severity 1

Instruction:

    1. In the log-in form, enter the username and password.
    2. Click the "Login" button.

Expected Result:

    1. The staff can access stock-in, manage stock modules, and release form.

Forgot Password Process (Administration)

Test Case 1

Module: Forgot Password Form

Severity 1

Instruction:

    1. In the log-in form, enter the username and password.
    2. Click the "Forgot Password" link.
    3. Answer recovery question.

Expected Result:
    1. Password would show if the answer matches.

Manage Account Process (Administration)

Test Case 1

Module: Manage Account Form

Severity 1

Instruction:

    1. Log in and click the manage account link in the main menu.
    2. Change desired information.
    3. By changing the password, confirm the password.
    4. Then click the "Update" button.

Expected Result:

1. The updated account would be saved.

Test Case 2 (Remove Account)

Module: Manage Account Form

Severity 1

Instruction:

1. Log in and click manage the account.
2. Click the "Remove" button then click "Yes"

Expected Result:

1. The account is removed and the user would be logged out.

Create Account Process (Administration)

Test Case 1

Module: Manage Account Form

Severity 1

Instruction:

1. Click the "Log-in" button in the main form.
2. Click the "Create Account" link.
3. Fill in the required fields.
4. Then click the "Save" button.

Expected Result:

1. The created account would be saved.

Recording of Staff's Process (Acquisition)

Test Case 1 (Adding Staff information)

Module: Staff's Form

Severity 1

Instructions:

1. On the main menu click "Staff".
2. Input detailed employee information
3. Then click the "Save" button

Expected Result:

1. The Staff information should be saved.

2. Automatically displays the Staff information.

Test Case 2 (Updating Staff's Information)

Module: Staff Form

Severity: 1

Instructions:

1. On the main menu click "Staff".
2. Double-click the staff in the Data Grid View for updating.
3. Edit the staff's information.
4. Click the "Update" button.
5. Then click the "Yes" command for confirmation.

Expected Result:

1. The staff's information is successfully updated.

Test Case 3 (Deleting Staff's Information)

Module: Staff's Form

Severity: 1

Instructions:

1. On the main menu click "Staff".
2. Double-click the staff in the Data Grid View for deleting.
3. Click the "Delete" button.
4. Then click the "Yes" command for confirmation.

Expected Result:

1. Staff's information is successfully deleted.

Recording of Supplier's Process (Acquisition)

Test Case 1 (Adding Supplier's Information)

Module: Supplier's Form

Severity: 1

Instructions:

1. On the main menu click "Supplier".
2. Input detailed supplier information.
3. Click the "Save" button.

Expected Result:

1. The supplier information should be saved.

2. Automatically displays the supplier information.

Test Case 2 (Updating Supplier's Information)

Module: Supplier's Form

Severity: 1

Instructions:

1. On the main menu click "Supplier".
2. Double-click the supplier in the Data Grid View for updating.
3. Edit the supplier's information.
4. Click the "Update" button.
5. Then click the "Yes" command for confirmation.

Expected Result:

1. The supplier's information is successfully updated.

Test Case 3 (Deleting Supplier's Information)

Module: Supplier's Form

Severity: 1

Instructions:

1. On the main menu click "Supplier".
2. Double-click the supplier in the Data Grid View for deleting.
3. Click the "Delete" button.
4. Then click the "Yes" command for confirmation.

Expected Result:

1. The supplier's information is successfully deleted.

Recording of Item/s Process (Acquisition)

Test Case 1 (Adding New Item)

Module: Add Stock Form

Severity: 1

Instructions:

1. On the main menu click "Stock".
2. Click "Add New" in the dropdown option.
3. Input detailed item information.
4. Click the "Save" button.

Expected Result:

1. The item information should be saved.
2. Automatically displays the item information.

Test Case 2 (Updating Item/s Information)

Module: Add Stock Form

Severity: 1

Instructions:

1. Double-click the item in the Data Grid View in the main form.
2. Edit the item information.
3. Click the "Update" button.
4. Then click the "Yes" command for confirmation.

Expected Result:

1. Item information is successfully updated.

Test Case 3 (Deleting Item/s Information)

Module: Add Stock Form

Severity: 1

Instructions:

2. Double-click the item in the Data Grid View in the main form.
3. Click the "Delete" button.
4. Then click the "Yes" command for confirmation.

Expected Result:

1. Item information is successfully deleted.

Recording of Customer's Process (Release)

Test Case 1 (Adding Customer's Information)

Module: Customer's Form

Severity: 1

Instructions:

4. On the main menu click "Customer".
5. Input detailed customer information.
6. Click the "Save" button.

Expected Result:

3. The customer information should be saved.
4. Automatically displays the customer information.

Test Case 2 (Updating Customer's Information)

Module: Customer's Form

Severity: 1

Instructions:

6. On the main menu click "Customer".
7. Double-click the customer in the Data Grid View for updating.
8. Edit the customer's information.
9. Click the "Update" button.
10. Then click the "Yes" command for confirmation.

Expected Result:

2. The customer's information is successfully updated.

Test Case 3 (Deleting Customer's Information)

Module: Customer's Form

Severity: 1

Instructions:

5. On the main menu click "Customer".
6. Double-click the customer in the Data Grid View for deleting.
7. Click the "Delete" button.
8. Then click the "Yes" command for confirmation.

Expected Result:

2. The customer's information is successfully deleted.

Item Release Process (Release)

Test Case 1 (Adding Stock for Release)

Module: Release Form

Severity: 1

Instructions:

1. On the main menu, click "Release".
2. Double-click the item in the Data Grid View in the main form.
3. Click the "Release" button.
4. "ReaseTo" form would appear and click the "Confirm" button.
5. Click the "Yes" command for confirmation.

Expected Result:

1. It should display the user who released the item.

2. It should display the date and time of release.
3. It should display the correct total quantity of the items.
4. It should display the correct change of the items.
5. A report is generated for the item released.
6. Can print the report.

Test Case 2 (Cancel/ Return Items in Cart)

Module: Release Form

Severity: 1

Instructions:

1. Double-click the item in the Data Grid View in the Release form to deduct one from the item.
2. Long-click the item in the Data Grid View in the Release form to completely remove the item.

Expected Results:

1. It should display the correct total quantity of the items.
2. It should display the correct change of the items.

Stock Monitoring Process

Test Case 1 (Monitoring of Available Stock)

Module: Home/Main Form

Severity: 1

Instructions:

1. Run the application.

Expected Result:

1. Automatically displays all available item/s.

Test Case 2 (Monitoring of Newly Added Stocks)

Module: Home/Main Form

Severity: 1

Instructions:

1. On the main menu, click "Report".
2. Select "Newly Added Stock" in the option.

Expected Result:

1. Automatically displays all newly added item/s.

Test Case 3 (Monitoring of Critical Stocks)

Module: Home/Main Form

Severity: 1

Instructions:

3. On the main menu, click "Report".
4. Select "Critical Stock" in the option.

Expected Result:

2. Automatically displays all critical-stock item/s.

Test Case 4 (Monitoring of Out-of-Stock)

Module: Home/Main Form

Severity: 1

Instructions:

1. On the main menu, click "Report".
2. Select "Out of Stock" in the option.

Expected Result:

1. Automatically displays all Out-of-Stock item/s.

Test Case 5 (Monitoring of Close to Expiry Items)

Module: Home/Main Form

Severity: 1

Instructions:

1. On the main menu, click "Report".
2. Select "Close to Expiry" in the option.

Expected Result:

1. Automatically displays all Close to Expiry item/s.

Test Case 6 (Monitoring of Expired Stock)

Module: Home/Main Form

Severity: 1

Instructions:

1. On the main menu, click "Report".
2. Select "Expired Stocks" in the option.

Expected Result:

1. Automatically displays all expired item/s.

Report Process (Report)

Test Case 1 (View Released Transactions of Users)

Module: Records Form

Severity: 1

Instructions:

1. On the main menu, click "Report".
2. Select "Released Records" in the option.
3. Search/Sort and filter data to view.
4. Click the "Ok" button.

Expected Result:

1. Automatically displays all records.

Test Case 2 (Top Release Item Report)

Module: Report Form

Severity: 1

Instructions:

1. On the main menu, click "Report".
2. Select "Top Released" in the option.

Expected Result:

1. Display Top 10 release items.
2. Can print report.

Administrative Log-out Process

Test Case 1

Module: Home/Main Form

Severity: 1

Instructions:

1. Click the "Log-out" link right below the "Log-in" button.

Expected Result:

1. User Log-out successfully.

Technical Requirements

All computers, to function properly, must have certain hardware components and software resources installed. The right selection of hardware and software components, as well as the identification of users involved in the operation, are critical for successful operation and maximizing the system's potential.

The hardware component describes the computer's physical components. The motherboard, microprocessor, hard disk, RAM (Random-Access Memory), UPS (Uninterruptible Power Supply), monitor, mouse, and keyboard are all included in this. However, there are only three (3) components that help in the data processing. The CPU, hard disk drive (HDD), and Random-Access Memory (RAM) are these components.

The programs that the computer system can understand and execute certain tasks are referred to as software. It is a series of instructions or data that the computer converts into a machine-readable format to operate it. Furthermore, the software generates an output that a person can comprehend.

"Peopleware" a term coined by Neuman, 1977 as cited in Computer Hope, 2020 refers to the human role in the development of computers and related devices, as well as the software and hardware components. It should be able to do the work appropriately to ensure the system's efficacy and produce accurate results in all data processing. It also refers to the system's users and the system user of our study is the M3 Farma Marketing owner and the staff.

The following specification is required for the deployment of the designed system: Stock Control Management System of M3 Farma Marketing.

Minimum Hardware Specification

Minimum hardware specification refers to a hardware device that can nevertheless meet the system's minimum requirements. This section details the hardware requirements for the system to perform as intended. These requirements were chosen based on the system development tool(s) utilized by the developers.

Component	Specification
Microprocessors	Intel Core I3
Hard Disk Drive	118 Gigabytes
RAM	4 Gigabytes

Minimum Software Specification

A software requirements specification seeks to provide a detailed description of the software's intended function and environment according to Rosencrance (2019). To work and run properly, it requires distinct software. This software was listed below, along with its associated requirements. These specs were created using the specifications of the computer units used throughout the system's development.

Item	Specification
Operating System	Windows 10
Wamp Server	Version 3.2.3
MySQL	Version 8.0.21

Economic Performance Evaluation

The analysis is one of the most important aspects of any system or design endeavour. This includes determining the estimated cost of the produced system as well as the benefits it provides. The time it takes for the new system to save enough money to cover its development costs is calculated. This section specifies the planned budget for the resources to be utilized.

Investment and Annual Operating Cost

Items	Quantity	Unit	Unit Price	Total
A. Initial Investment				
A.1. Software				
Software Development	1	Lot	₱ 10,000.00	₱ 10,000.00
Software Licensing	1	Lot	₱ 8,000.00	₱ 8,000.00
Installation	1	Piece	₱ 1,000.00	₱ 1,000.00
Sub-total				₱ 19,000.00
A.2. Hardware				
Desktop Computer	2	Set	₱ 25,000.00	₱ 50,000.00
Printer	1	Unit	₱ 10,000.00	₱ 10,000.00
Sub-total				₱ 60,000.00
Total Initial Investment Cost				₱ 79,000.00
B. Annual Operating Cost				
B.1. Utilities				
Electricity	12	Kwh	₱ 120.00	₱ 1,440.00
B.2. General Services				
System Maintenance	1	Lot	₱ 1,000.00	₱ 1,000.00
Total Annual Operating Cost				₱ 2,440.00

Business Intelligence

Business intelligence, according to Stedman & Burns (2020), is a technology-driven process for analyzing data and delivering actionable information that helps executives, managers and workers make informed business decisions. The goal is to increase the consistency and quality of the data used in decision-

making. Business intelligence is used to understand the firm's capabilities, as well

as the state of the art, trends, and future directions for the market, technologies,

and regulatory environment in which the company competes, as well as competitor

actions and the implications of those actions.

Preview 1 below shows the notifications form/report.

Preview 2 below shows the top release item report in graphical form.
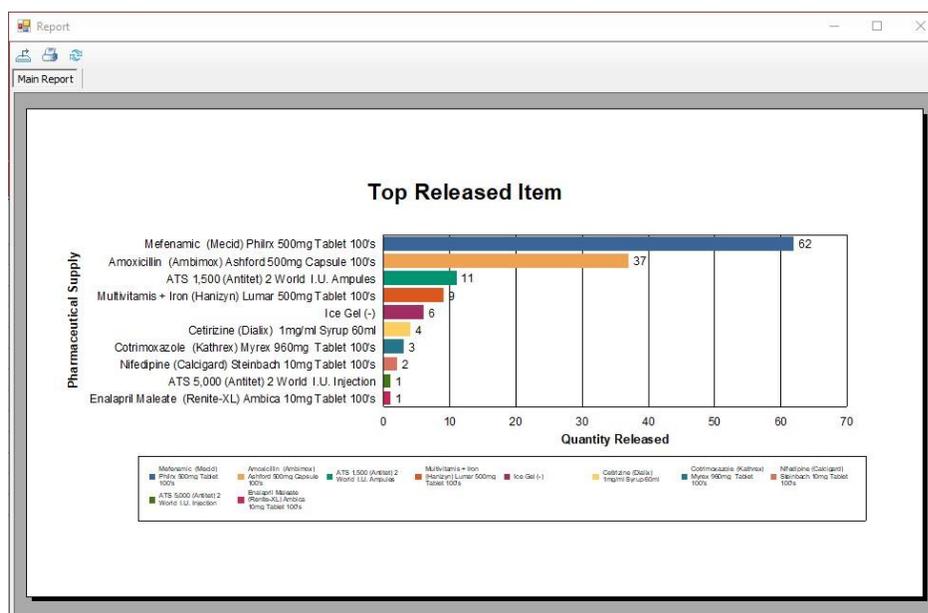
Screen Layout

The design of graphical user interfaces is described by the screen layout. It encompasses a wide range of applications in which screens or displays can be utilized as a part of human-machine interaction, and should be distinguished from graphical user interface activities (Ryte Wiki, 2021). One of the numerous aspects of the system's user-friendliness is the screen layout. It should be constructed in such a way that users can navigate the system quickly and easily, with a clear awareness of the tasks it must do.

The screen layout of the created system is shown in the following preview.

Preview 1 below shows the log-in process for users.

Preview 1. Log-in Form

Preview 2 below shows the recover password form.

Preview 2. Recover Password Form

Preview 3 below shows the create or manage account form.

Preview 3. Manage Account Form

Preview 4 below shows the main menu form and the available stocks.

Preview 4. Main Form

Preview 5 below shows the item release form.

Preview 5. Release Form

Preview 6 below shows the release to form.

Preview 6. Release To Form

Preview 7 below shows the supplier form.

Preview 7. Supplier Form

Preview 8 below shows the add stock form.

Preview 8. Add Stock Form

Preview 9 below shows the staff form.

Preview 9. Staff Form

Preview 10 below shows the customer form.

Preview 10. Customer Form

Preview 11 below shows the critical stock report.

Preview 11. Report Form

Preview 12 below shows the out-of-stock report.

Preview 12. Report Form

Preview 13 below shows the close-to-expiry report.

Preview 13. Report Form

Preview 14 below shows the expired stock report.

M3 FARMA MARKETING
PHARMACEUTICAL SUPPLIES DISTRIBUTOR
Prop - MARINO P. OMAC   Non-VAT Reg. TIN-210-680-620-000

EXPIRED STOCK   Date: 18/07/2022

DESCRIPTION	EXPIRATION DATE	LOT/BATCH NO.	QTY.
Nifedipine2 (Calcigard) Steinbach 10mg Tablet 100's	08/2020	01219292	17

Preview 14. Report Form

Preview 15 below shows the item released records form.

Released Records

Authorized Representative

Enter Year   OK
Select Month

Customer	Address	Authorized Representative	Date Released	Description	Lot/Batch No.	Quantity
		Kimberly C. Curay	25/05/2022 5:23 am	Multivitamis + Iron (...	132025	2
		Kimberly C. Curay	25/05/2022 5:23 am	Mefenamic (Mecid) ...	211195	5
Calape Pharmacy	Calape, Bohol	Marino	19/07/2022 2:22 am	Multivitamis + Iron (...	132025	1
Calape Pharmacy	Calape, Bohol	Marino	19/07/2022 2:22 am	Amoxicillin (Ambimo...	706211252	1
Calape Pharmacy	Calape, Bohol	Marino	19/07/2022 2:22 am	Cotrimoxazole (Kath...	YAN082	2
		Marino	18/07/2022 12:10 pm	ATS 5,000 (Antitet) 2 ...	J200201	1
		Marino	19/07/2022 9:53 am	Amoxicillin (Ambimo...	706211252	1
		Marino	19/07/2022 9:53 am	Enalapril Maleate (R...	F803	1
		Marino	19/07/2022 10:35 am	Cotrimoxazole (Kath...	YAN082	4
		Marino	19/07/2022 10:35 am	Enalapril Maleate (R...	F803	1
		Marino	19/07/2022 10:59 am	Enalapril Maleate (R...	F803	1
		Marino	19/07/2022 10:59 am	Amoxicillin (Ambimo...	706211252	1
		Marino	19/07/2022 10:59 am	Cotrimoxazole (Kath...	YAN082	1
		Marino	19/07/2022 11:05 am	Cotrimoxazole (Kath...	YAN082	2
		Marino	19/07/2022 11:07 am	Cotrimoxazole (Kath...	YAN082	2
BTP Pharmacy	Dauis, Pharmacy	Marino	19/07/2022 11:59 am	Mefenamic (Mecid) ...	211195	2
Botica Rezel	Talibon, Bohol	Marino	19/07/2022 12:12 pm	Mefenamic (Mecid) ...	211195	2
Botica Rezel	Talibon, Bohol	Marino	19/07/2022 12:12 pm	Nifedipine (Calcigar...	0121929	2

Preview 15. Records Form

Testing and Evaluation

Testing and Evaluation is the process of comparing a system or component to requirements and specifications through testing, according to The MITRE Corporation, 2015. The outcomes are assessed to determine design progress, performance, and supportability, among other things. Testing and assessment were carried out to verify the system's functionality, including the provision of the desired output, time/period of information processing, the volume of data processed, and proper response to user inputs. This is also the process of evaluating the system's overall performance. The usability of the system was assessed in this study to identify its technical performance as regarded by the targets.

System Usability

Software testing is a means of determining whether the software is performing as expected, such as providing accurate output, running quickly enough, handling expected loads, and appropriately responding to user inputs. Software evaluation is the process of determining whether the Software's original aims have been met. The poll had three participants: the owner and the staff.

The system usability questionnaire was used by the researcher. The researchers demonstrated and presented the module of the system. It was pilot tested for them to use and give feedback. The table below shows the pertinent respondents during the pilot testing.

The system usability result is shown in the table below, which includes the weighted mean and the meaning of each statement.

Criteria for System Usability	Rating	Interpretation
1. Overall, I am satisfied with how easy it is to this system.	5.7	Agree
2. It was simple to use this system.	6.3	Agree
3. I can complete my work quickly using this system.	6	Agree
4. I can efficiently complete my work using this system	6	Agree
5. I feel comfortable using this system	6.3	Agree
6. It was easy to learn to use this system	6.7	Strongly Agree
7. I believe I became productive using this system	6	Agree
8. It is easy to find the information I needed	6.3	Agree
9. The information provided for the system is easy to understand	6.3	Agree
10. The information is effective in helping me complete the tasks and scenarios	6.7	Strongly Agree
11. The organization of information on the system on screens is clear	6.7	Strongly Agree
12. The interface of this system is pleasant	6	Agree
13. I like using the interface of this system	6.3	Agree
14. This system has all the functions and capabilities I expect it to have	5.7	Agree
15. Overall, I can be satisfied with this system	6.3	Agree
AVERAGE WEIGHT MEAN	6.2	Agree

Chapter 3

SUMMARY OF FINDINGS, CONCLUSION AND RECOMMENDATION

Summary of Findings

The researchers discovered that the current system uses manual processes in M3 Farma Marketing's stock control management, after conducting an investigation and study. Furthermore, monitoring is tough, as is retrieving records concerning stock data, and the compilation of reports is a time-consuming task. Thus, records must be organized and kept secure. Releasing of stocks should also be monitored and recorded and the generation of reports is an important asset to help assess the owner in monitoring the status of his business.

M3 Farma Marketing Stock Control Management System was designed using the modules Administration, Acquisition, Release and Reports to cover the establishment's activities based on the specified needs. The developed system was pilot tested, and end-users were polled to score the system's usefulness using system usability standards. The system usability questionnaire's average weighted mean is 6.2, which means "Agree." The interpretation indicates that the implementation of the system based on the system's usability result is possible. The findings also imply that the method gives satisfaction to the respondents. It also demonstrates that the method is really basic and straightforward to use, as well as effective, instructive, simple to comprehend, and obvious.

Conclusion

In conclusion, the project is relevant to the objectives set. The project is based on a preliminary investigation conducted at M3 Farma Marketing. Thus, the planning and analytical operations of designing the system are based on the results obtained from the investigation. Not only that but because this would be the store's first computerized system, the functionality is limited to addressing one significant issue: Stock Control Management. Still, the computerization of the existing activities in M3 Farma Marketing is possible and attainable, according to its current procedures and operations.

Based on the functional evaluation of M3 Farma Marketing, the system's capacity and features somehow met the client's expectations for improving business operations. Furthermore, the system's adoption is reasonable and cost-effective considering the economic performance evaluation.

Recommendation

The developers have advised the following for efficient record keeping, seamless system adoption and operation, and future development based on observations made during the implementation.

1. M3 Farma Marketing must implement the system to improve stock control and inventory management, as well as uncover potential system problems and errors.

2. The owner and the staff of M3 Farma Marketing must be given a training or orientation sessions in order to familiarize and orient them with the features and functionality of the new system.

3. M3 Farma Marketing must regulate system maintenance to guarantee the stability of the system and the security of the records.

References

Abid, A. (2021). *The Role of Information Technology in Business Success*. LinkedIn. Retrieved April 03, 2021 from https://www.linkedin.com/pulse/role-information-technology-business-success-abid-afzal-butt/

Acopiado, S., Dabodabo, T., Gomez, A., Santiago, K. *Sales and Inventory.* Unpublished Thesis Document. Saint Vincent's College.

Brush, K. (2020). *Test Case.* SearchSoftwareQuality. Retrieved March 06, 2020 from https://www.techtarget.com/searchsoftwarequality/definition/test-case?amp=1

Busbee, K.L. (2018). *Hierarchy or Structure Chart - Programming Fundamentals.* Pressbooks. Retrieved December 15, 2018 from https://press.rebus.community/programmingfundamentals/chapter.hierarchy-or-structure-chart/

Computer Hope. (2020). *What is Peopleware?* Retrieved March 06, 2020 from https://www.computerhope.com/jargon/p/peopleware.htm

Crosby, T. (2020). *How Inventory Management System Work.* HowStuffWorks. Retrieved July 27, 2020 from https://money.howstuffworks.com/how-inventory-management-systems-work.html

Dobing, B., & Parsons J. (2009). *Use Cases in the UML.* IGI Global. Retrieved July 15, 2022 from https://www.igi-global.com/chapter/use-cases-uml/14160

Epa, K., Saludo, R., Busbus, R. (2020). *Sales and Inventory System of ClipMed Pharmacy.* Unpublished Undergraduate Thesis. Bohol Island State University, Zamora, Bilar, Bohol.

Executive Property Investment. (2002). *Bronze inventory system* (1.3.5) [Computer software]. https://executivpro.com/products/bronze

Executive Property Investment. (2002). *Silvernet inventory system.* https://executivpro.com/products/silvernet

Hayes, A. (2022). *How Point of Sale (POS) Works.* Investopedia. Retrieved May 29, 2022 from https://www.investopedia.com/terms/p/point-of-sale.asp/

Hindsman, A. (2020). *Guide to inventory management and stock control.* Sage Advice South Africa. Retrieved March 09, 2020 from https://www.sage.com/en-za/blog/inventory-management-guide/

Jenkins, A. (2020). *Must-Have Inventory Management System Features, Requirements & Modules.* Oracle NetSuite. Retrieved August 26, 2020 from https://www.netsuite.com/portal/resource/articles/inventory-management/inventory-management-system-features.shtml/

Jenkins, A. (2021). *Inventory Control vs. Inventory Management: What's the Difference?* Oracle NetSuite. Retrieved March 02, 2021 from https://www.netsuite.com/portal/resource/articles/inventory-management/inventory-management-inventory-control.shtml/

Jenkins, A. (2022). *14 Top Inventory Management Trends to Know in 2021*. Oracle NetSuite. https://www.netsuite.com/portal/resource/articles/inventory-management/inventory-management-trends.shtml

Lewis, J.R. (1995) *IBM Computer Usability Satisfaction Questionnaires: Psychometric Evaluation and Instructions for Use.* International Journal of Human – Computer Interaction, 7:1, 57-78.

Martin, M. (2022). *What is a Functional Requirement in Software Engineering? Specification, Types, Examples.* Guru99. Retrieved June 25, 2022 from https://www.guru99.com/functional-requirement-specification-example.html

Mehmedovic, D. (2020). *8 Benefits of Having Stock Control Management System in Your Company.* Sage 50 Wholesale Software. https://spire.today/8-benefits-of-having-stock-control-management-system/

Milano, S. (2020). *How to Figure Out Minimum Inventory Levels.* Small Business - Chron.Com. https://smallbusiness.chron.com/figure-out-minimum-inventory-levels-25863.html

Mrabu, L. (2020). *Assessment of the Effectiveness of Inventory Control System. A case of Tanzania Electric Supply Company Limited* (Doctoral dissertation, Mzumbe University)

Ogbo, A. I., & Ukpere, W. I. (2014). The impact of effective inventory control management on organisational performance: A study of 7up bottling company nile mile enugu, nigeria. *Mediterranean Journal of Social Sciences*, *5*(10), 109-109.

Mueller, D. R. (2021). *What Is Stock Control?* Bizfluent. https://bizfluent.com/info-8088223-stock-control.html

Pandya, B., & Thakkar, H. (2016). A review on inventory management control techniques: ABC-XYZ analysis. *REST Journal on Emerging trends in Modelling and Manufacturing*, *2*(3), 2016.

Phruksaphanrat, B. & Tanthatemee, T. (2012). *Fuzzy Inventory Control System for Uncertain Demand and Supply.* Lecture Notes in Engineering and Computer Science. 2196. 1224-1229.

Peterson, R. (2022). *Database Design in DBMS Tutorial. Learn Data Modeling.* Guru99. Retrieved June 18, 2022 from https://www.guru99.com/database-design.html/

ReQtest author (2020). *Functional vs Non-Functional Requirements – Understand the Difference*. ReQtest. Retrieved July 28, 2020 from https://reqtest.com/requirements-blog/functional-vs-non-functional-requirements/

Ros, F. (2022). *Stock Management System In VB.NET With Source Code*. Source Code & Projects. https://code-projects.org/stock-management-system-in-vb-net-with-source-code/

Rosario, J.M. (2022). *Implementation of Inventory Control Management and Repeat Purchase in Right Goods Philippines Incorporated: Inputs to Policy Reformulation.*

Rosencrance, L. (2019). *Software Requirements Specification (SRS).* SearchSoftwareQuality. Retrieved September 10, 2019 from https://www.google.com/amp/s/www.techtarget.com/searchsoftwarequality/definition/software-requirements-specification%3famp=1

Ryte Wiki. (2021). *Screen Design.* https://en.ryte.com/wiki/Screendesign

Stedman, C., & Burns, E. (2020). *Business Intelligence (BI).* SearchBusinessAnalytics. Retrieved September 17, 2020 from https://www.google.com/amp/s/www.techtarget.com/searchbusinessanalytics/definition/business-intelligence-BI%3famp=1

TechTarget contributors. (2020). *Use case diagram (UML use case diagram).* WhatIs.Com. Retrieved July 24, 2020 from https://www.google.com/amp/s/www.techtarget.com/whatis/definition/use-case-diagram%3famp=1/

The MITRE Corporation. (2015). *Test and Evaluation.* Retrieved April 10, 2015 from https://www.mitre.org/publications/systems-engineering-guide/se-lifecycle-building-blocks/test-and-evaluation

Tom Jose, V., Jayakumar, A., & Sijo, M. T. (2013). Analysis of inventory control techniques; a comparative study. *International Journal of Scientific and Research Publications*, *3*(3), 1.

What is a Software Process Model? Visual Paradigm. Retrieved from https://www.visual-paradigm.com/guide/software-development-process/what-is-a-software-process-model/

Zaman, N. (2013). *Efficient Stock Control Management System.* Unpublished Thesis Document. Malaysia.

APPENDICES

APPENDIX A

INTERVIEW GUIDE QUESTIONS

Owner:

1. What are the problems that the current system has?

2. How do you handle the arrival of new stocks?

3. What methods are used to save and retrieve records?

4. What is the current inventory system processes of the company?

5. How long does it take to record stock information?

6. How often do you conduct an inventory?

7. Who is authorized to take stock from the warehouse?

8. How do you monitor and record the disposal of stocks?

9. What kinds of reporting are required? How frequently is this required?

10. Would you agree to improve and enhance the present system through computer automation?

Staff:

1. Do you monitor a product's expiry date?

2. Who keeps track of stock availability?

3. How do you determine the stock's availability?

4. How often do you take stocks from the warehouse?

5. Do you provide a report?

LETTER OF INTENT

Republic of the Philippines
Bohol Island State University
Bilar Campus
Zamora, Bilar, Bohol

March 29, 2022

MARINO P. OMAC
Owner
M3 Farma Marketing
Riverside, Bilar, Bohol

Sir:

Good day!

We, the Senior Students of Bachelor of Science in Computer Science of Bohol Island State University – Bilar Campus, are to conduct a system development project (thesis) as requirement for graduation.

In this regard, we would like to ask your good office to grant us permission to conduct a system study (thesis) based on the Stock Control Management System of your establishment as basis for our proposed automation. This study would include an interview, observation of transactions and review of documents significant for the study and development of the system. We anticipate having an implementation testing (bench marking) of the system developed as part of the usability testing.

We assure you that we shall honor secrecy and privacy to all data and information we shall be handling as we go along with our study. Your approval will be a great help to the success of our study.

We anticipate your favorable response regarding this matter.

Thank you very much and more power!

Respectfully yours,

JUNREY O. LIGTAS

KIMBERLY C. CURAY

LARRY C. QUILARIO
Researchers

Noted:

SHEILA G. TABUNO
Subject Instructor

DANIE D. BALDESCO
Thesis Adviser

Recommended by:

SHEILA G. TABUNO
Chairperson, DCoS

Endorsed by:

ARLEN B. GUDMALIN, PhD
Dean, CTAS

Approved by:

MARINO P. OMAC
Owner, M3 Farma Marketing

LETTER OF IMPLEMENTATION

Republic of the Philippines
Bohol Island State University
Bilar Campus
Zamora, Bilar, Bohol

May 30, 2022

MARINO P. OMAC
Owner
M3 Farma Marketing
Riverside, Bilar, Bohol

Sir:

Good day!

It is our pleasure to inform you that the system "Stock Control Management System, Riverside, Bilar, Bohol" is now in its final phase. With this, we would like to conduct benchmarking activities as part of the implementation.

It will be conducted on May 30, 2022 in your office at any time of your convenience. This activity will allow you to assess our developed system and give feedback, as well.

But this time, we would like to express our gratitude for allowing us to conduct our thesis study. We are hoping for future collaborations with you, our dear client.

May the good Lord continually bless you and your business.

Thank you very much and more power!

Respectfully yours,

JUNREY O. LIGTAS

KIMBERLY C. CURAY

LARRY C. QUILARIO
Researchers

Noted:

SHEILA G. TABUNO
Subject Instructor

DANIE D. BALDESCO
Thesis Adviser

Recommended by:

SHEILA G. TABUNO
Chairperson, DCoS

Endorsed by:

ARLEN B. GUDMALIN, PhD
Dean, CTAS

Approved by:

MARINO P. OMAC
Owner, M3 Farma Marketing

APPENDIX B

SYSTEM USABILITY QUESTIONNAIRE

Instructions:

- Please rate the usability of the system

- Try to respond to all the items

- For items that are not applicable, use N/A

- Make sure those fields are filled in

Rating Scales:

7 - Strongly Agree

6 - Agree

5 - Tend to Agree

4 - Neither Agree or Disagree

3 - Ten to Disagree

2 - Disagree

1 - Strongly Disagree

Criteria for System Usability	Rating
1. Overall, I am satisfied with how easy it is to this system.	
2. It was simple to use this system.	
3. I can complete my work quickly using this system.	
4. I can efficiently complete my work using this system	
5. I feel comfortable using this system	
6. It was easy to learn to use this system	
7. I believe I became productive using this system	

8. It is easy to find the information I needed	
9. The information provided for the system is easy to understand	
10. The information is effective in helping me complete the tasks and scenarios	
11. The organization of information on the system in screens is clear	
12. The interface of this system is pleasant	
13. I like using the interface of this system	
14. This system has all the functions and capabilities I expect it to have	
15. Overall, I can be satisfied with this system	
AVERAGE WEIGHT MEAN	

APPENDIX C

USER'S MANUAL

Account Log-In

Owner/Admin Account Log-in

Steps:

1. In the log-in form, enter the username and password.
2. Click the "Log-in" button.
3. Confirmation "Log-in Successfully" will appear.

User Account Log-in

Steps:

1. In the log-in form, enter the username and password.
2. Click the "Login" button.
3. Confirmation "Log-in Successfully" will appear.

Forgot Password

Recover Password

Steps:

1. In the log-in form, enter the username and password.
2. Click the "Forgot Password" link.
3. Answer recovery question.
4. Password would show if the answer matches.

Manage Account

Manage Accounts

Steps:

1. Log in and click the manage account link in the main menu.
2. Change desired information.
3. By changing the password, confirm the password.
4. Then click the "Update" button.
5. Confirmation "Account Updated" will appear.

Remove Account

Steps:

1. Log in and click manage the account.

2. Click the "Remove" button then click "Yes"
3. Confirmation "Removed Successfully" will appear.

Create Account

Create New Account

Steps:

1. Click the "Log-in" button in the main form.
2. Click the "Create Account" link.
3. Fill in the required fields.
4. Then click the "Save" button.
5. Confirmation "Saved" will appear.

Recording of Staff Information

Adding Staff Information

Steps:

1. On the main menu click "Staff".
2. Input detailed employee information
3. Then click the "Save" button
4. Confirmation "Saved" will appear.

Updating Staff Information

Steps:

1. On the main menu click "Staff".
2. Double-click the staff in the Data Grid View for updating.
3. Edit the staff's information.
4. Click the "Update" button.
5. Then click the "Yes" command for confirmation.
6. Confirmation "Changes Added" will appear.

Deleting Staff Record

Steps:

1. On the main menu click "Staff".
2. Double-click the staff in the Data Grid View for deleting.
3. Click the "Delete" button.
4. Then click the "Yes" command for confirmation.

Recording of Supplier Information

Adding Supplier Information

Steps:

1. On the main menu click "Supplier".
2. Input detailed supplier information.
3. Click the "Save" button.
4. Confirmation "Saved" will appear.

Updating Supplier Information

Steps:

1. On the main menu click "Supplier".
2. Double-click the supplier in the Data Grid View for updating.
3. Edit the supplier's information.
4. Click the "Update" button.
5. Then click the "Yes" command for confirmation.
6. Confirmation "Changes Added" will appear.

Deleting Supplier Information

Steps:

1. On the main menu click "Supplier".
2. Double-click the supplier in the Data Grid View for deleting.
3. Click the "Delete" button.
4. Then click the "Yes" command for confirmation.

Recording of Stock Information

Adding New Item

Steps:

1. On the main menu click "Stock".
2. Click "Add New" in the dropdown option.
3. Input detailed item information.
4. Click the "Save" button.
5. Confirmation "Added Successfully" will appear.

Updating Item Information

Steps:

1. Double-click the item in the Data Grid View in the main form.
2. Edit the item information.
3. Click the "Update" button.
4. Then click the "Yes" command for confirmation.
5. Confirmation "Changes Added" will appear.

Deleting Item Record

Steps:

1. Double-click the item in the Data Grid View in the main form.
2. Click the "Delete" button.
3. Then click the "Yes" command for confirmation.

Recording of Customer Information

Adding Customer Information

Steps:

1. On the main menu click "Customer".
2. Input detailed customer information.
3. Click the "Save" button.
4. Confirmation "Saved" will appear.

Updating Customer Information

Steps:

1. On the main menu click "Customer".
2. Double-click the customer in the Data Grid View for updating.
3. Edit the customer's information.
4. Click the "Update" button.
5. Then click the "Yes" command for confirmation.
6. Confirmation "Changes Added" will appear.

Deleting Customer Information

Steps:

1. On the main menu click "Customer".
2. Double-click the customer in the Data Grid View for deleting.
3. Click the "Delete" button.
4. Then click the "Yes" command for confirmation.

Stock Release

Releasing Stock

Steps:

1. On the main menu, click "Release".
2. Double-click the item in the Data Grid View in the main form.
3. Click the "Release" button.
4. "ReaseTo" form would appear and click the "Confirm" button.
5. Click the "Yes" command for confirmation.

Cancel/Return Items in Cart

    Steps:

1. Double-click the item in the Data Grid View in the Release form to deduct one from the item.
2. Long-click the item in the Data Grid View in the Release form to completely remove the item.

Stock Monitoring

Monitoring of Available Stock

    Steps:

1. Run the application.
2. The system displays the list of available items.

Monitoring of Newly Added Stocks

    Steps:

1. On the main menu, click "Report".
2. Select "Newly Added Stock" in the option.

Monitoring of Critical Stocks

    Steps:

1. On the main menu, click "Report".
2. Select "Critical Stock" in the option.

Monitoring of Out-of-Stock

    Steps:

1. On the main menu, click "Report".
2. Select "Out of Stock" in the option.

Monitoring of Close to Expiry Items

    Steps:

1. On the main menu, click "Report".
2. Select "Close to Expiry" in the option.

Monitoring of Expired Stock

    Steps:

1. On the main menu, click "Report".
2. Select "Expired Stocks" in the option.

Reports

View Released Transactions of Users

    Steps:

1. On the main menu, click "Report".
2. Select "Released Records" in the option.
3. Search/Sort and filter data to view.
4. Click the "Ok" button.

View Top Release Item Report

    Steps:

1. On the main menu, click "Report".
2. Select "Top Released" in the option.

Log-out

Log-out Account

    Steps:

1. Click the "Log-out" link right below the "Log-in" button.

APPENDIX D

SOURCE CODE

Module Connect to Database

```vb
Imports MySql.Data.MySqlClient
Module DatabaseConnect
    Public con As New
MySqlConnection("Server=localhost;password=;user=root;database=m3farmamarketing;")
    Public Sub cQuery(ByVal query As String)
        Dim cmd As New MySqlCommand(query, con)
        con.Open()
        cmd.ExecuteNonQuery()
        con.Close()
    End Sub
    Sub setconnection()
        Try
            con.ConnectionString = "Server=" & My.Settings.host _
            & ";Database=" & My.Settings.dbname & ";User ID=" & My.Settings.usid _
            & ";Password=" & My.Settings.pwd & ""
            con.Open()
            con.Close()
        Catch ex As Exception
            ex.ToString()
        End Try
    End Sub
End Module
```

Log-in Form

```vb
Imports MySql.Data.MySqlClient
Public Class M3_LogIn
    Private Sub llForgotPass_LinkClicked(sender As Object, e As
LinkLabelLinkClickedEventArgs) Handles llForgotPass.LinkClicked
        If cbUserType.Text = Nothing And tbUserName.Text = Nothing Then
            MsgBox("Enter your User Type and User Name first!",
MsgBoxStyle.Information, "Sytem Checked")
            Return
        Else
            M3_ForgotPassword.Show()
        End If
    End Sub

    Private Sub btnLogIn_Click(sender As Object, e As EventArgs) Handles
btnLogIn.Click
        Dim theQuery As String = "SELECT Password FROM tblaccounts WHERE
UserType='" & cbUserType.Text & "' AND UserName='" & tbUserName.Text & "'"
        Dim cmd1 As MySqlCommand = New MySqlCommand(theQuery, con)
        If con.State = ConnectionState.Open Then
            con.Close()
        End If
        con.Open()
        Dim pass = cmd1.ExecuteScalar()
        con.Close()
```

```vb
        If tbPass.Text = Nothing Or cbUserType.Text = Nothing Or tbUserName.Text =
Nothing Then
            MsgBox("All fields required!", MsgBoxStyle.Information, "Wrong
Credentials")
        Else
            If tbPass.Text = pass Then
                MsgBox("Log-In Successfully!", MsgBoxStyle.Information, "Sytem
Checked")

                Dim del As String = "Delete from tblloggedin"
                Dim cmd6 As New MySqlCommand(del, con)
                If con.State = ConnectionState.Open Then
                    con.Close()
                End If
                con.Open()
                cmd6.ExecuteNonQuery()
                con.Close()

                Dim save As String = "Insert into
tblloggedin(UserType,UserName,Password)values('" & cbUserType.Text & "' " _
                                            & ",'" & tbUserName.Text & "',' " &
tbPass.Text & "')"
                Dim cmd As New MySqlCommand(save, con)
                con.Close()
                con.Open()
                cmd.ExecuteNonQuery()
                con.Close()
                Me.Close()

                M3_Home.llLogOut.Visible = True
                M3_Home.lblUserType2.Visible = True
                Dim theQuery2 As String = "SELECT UserType FROM tblloggedin"
                Dim cmd2 As MySqlCommand = New MySqlCommand(theQuery2, con)
                If con.State = ConnectionState.Open Then
                    con.Close()
                End If
                con.Open()
                M3_Home.lblUserType2.Text = cmd2.ExecuteScalar()
                con.Close()

                M3_Home.lblUserName2.Visible = True
                Dim theQuery3 As String = "SELECT UserName FROM tblloggedin"
                Dim cmd3 As MySqlCommand = New MySqlCommand(theQuery3, con)
                If con.State = ConnectionState.Open Then
                    con.Close()
                End If
                con.Open()
                M3_Home.lblUserName2.Text = cmd3.ExecuteScalar()
                con.Close()
                M3_Home.llManageAcc.Visible = True
                M3_Home.Show()
            Else
                MsgBox("Wrong Credentials!", MsgBoxStyle.Information, "Sytem
Checked")
            End If
        End If
    End Sub
```

```vbnet
    Private Sub llCreateAccount_LinkClicked(sender As Object, e As
LinkLabelLinkClickedEventArgs) Handles llCreateAccount.LinkClicked
        M3_ManageAccount.fillComboBox()
        M3_ManageAccount.Show()
    End Sub

    Private Sub tbPass_KeyDown(sender As Object, e As KeyEventArgs) Handles
tbPass.KeyDown
        If (e.KeyCode = Keys.Enter) Then
            btnLogIn.PerformClick()
        End If
    End Sub
End Class
```

Forgot Password Form

```vbnet
Imports MySql.Data.MySqlClient
Public Class M3_ForgotPassword
    Private Sub M3_ForgotPassword_Load(sender As Object, e As EventArgs) Handles
MyBase.Load
        Dim theQuery As String = "SELECT RecoveryQuestion FROM tblaccounts WHERE
UserType = '" & M3_LogIn.cbUserType.Text & "'" _
                                 & " AND UserName = '" & M3_LogIn.tbUserName.Text
& "'"
        Dim cmd1 As MySqlCommand = New MySqlCommand(theQuery, con)
        If con.State = ConnectionState.Open Then
            con.Close()
        End If
        con.Open()
        lblQuestion.Text = cmd1.ExecuteScalar()
        con.Close()
    End Sub

    Private Sub tbQuestion_TextChanged(sender As Object, e As EventArgs) Handles
tbAns.TextChanged
        Dim theQuery As String = "SELECT RecoveryAnswer FROM tblaccounts WHERE
UserType = '" & M3_LogIn.cbUserType.Text & "'" _
                                 & " AND UserName = '" & M3_LogIn.tbUserName.Text
& "'"
        Dim cmd1 As MySqlCommand = New MySqlCommand(theQuery, con)
        If con.State = ConnectionState.Open Then
            con.Close()
        End If
        con.Open()
        Dim ans = cmd1.ExecuteScalar()
        con.Close()

        If tbAns.Text = ans Then
            Dim theQuery2 As String = "SELECT Password FROM tblaccounts WHERE
UserType = '" & M3_LogIn.cbUserType.Text & "'" _
                                      & " AND UserName = '" &
M3_LogIn.tbUserName.Text & "'"
            Dim cmd2 As MySqlCommand = New MySqlCommand(theQuery2, con)
            If con.State = ConnectionState.Open Then
                con.Close()
            End If
            con.Open()
            lblPass.Text = cmd2.ExecuteScalar()
```

```vbnet
                con.Close()
            End If
        End Sub
End Class
```

Manage Account Form

```vbnet
Imports MySql.Data.MySqlClient
Public Class M3_ManageAccount
    Sub saveAcc()
        Dim theQuery As String = "SELECT * FROM tblaccounts WHERE
UserType=@UserType AND UserName=@UserName"
        Dim cmd1 As MySqlCommand = New MySqlCommand(theQuery, con)
        con.Open()
        cmd1.Parameters.AddWithValue("@UserName", cbUserName.Text)
        cmd1.Parameters.AddWithValue("@UserType", cbUserType.Text)
        Using reader As MySqlDataReader = cmd1.ExecuteReader()
            If reader.HasRows Then
                MsgBox("Already Registered!", MsgBoxStyle.Exclamation, "Error")
            Else
                If MessageBox.Show("Add Account?", "System Checked",
MessageBoxButtons.YesNo, MessageBoxIcon.Question) = Windows.Forms.DialogResult.Yes
Then
                    Dim save As String = "Insert into
tblaccounts(UserType,UserName,Password,RecoveryQuestion,RecoveryAnswer)values('" &
cbUserType.Text & "' " _
                                     & ",'" & cbUserName.Text & "','" &
tbPassword.Text & "', '" & convertQuotes(cbQuestion.Text) & "', '" & tbAns.Text &
"')"
                    Dim cmd As New MySqlCommand(save, con)
                    con.Close()
                    con.Open()
                    cmd.ExecuteNonQuery()
                    MsgBox("Saved!", MsgBoxStyle.Information, "System Checked")
                    con.Close()
                End If
            End If
        End Using
        con.Close()
    End Sub

    Sub updateAcc()
        Dim up As String = "Update tblaccounts set UserType ='" & cbUserType.Text
& "', UserName ='" & cbUserName.Text & "', Password ='" & tbPassword.Text & "'" _
                        & ",RecoveryQuestion='" &
convertQuotes(cbQuestion.Text) & "',RecoveryAnswer='" & tbAns.Text & "' WHERE
UserType = '" & cbUserType.Text & "'" _
                        & " AND UserName = '" & cbUserName.Text & "'"
        Dim cmd As New MySqlCommand(up, con)
        If con.State = ConnectionState.Open Then
            con.Close()
        End If
        con.Open()
        cmd.ExecuteNonQuery()
        MsgBox("Account Updated!", MsgBoxStyle.Information, "Success")
        con.Close()
        Me.Close()
    End Sub
```

```vbnet
    Sub accToFormForUpd()
        Dim theQuery As String = "SELECT Password FROM tblaccounts where
UserName='" & M3_Home.lblUserName2.Text & "'" _
                            & "AND UserType='" & M3_Home.lblUserType2.Text &
"'"
        Dim cmd1 As MySqlCommand = New MySqlCommand(theQuery, con)
        If con.State = ConnectionState.Open Then
            con.Close()
        End If
        con.Open()
        tbPassword.Text = cmd1.ExecuteScalar()
        con.Close()
        Dim theQuery2 As String = "SELECT RecoveryQuestion FROM tblaccounts where
UserName='" & M3_Home.lblUserName2.Text & "'" _
                            & "AND UserType='" & M3_Home.lblUserType2.Text &
"'"
        Dim cmd2 As MySqlCommand = New MySqlCommand(theQuery2, con)
        If con.State = ConnectionState.Open Then
            con.Close()
        End If
        con.Open()
        cbQuestion.Text = cmd2.ExecuteScalar()
        con.Close()
        Dim theQuery3 As String = "SELECT RecoveryAnswer FROM tblaccounts where
UserName='" & M3_Home.lblUserName2.Text & "'" _
                            & "AND UserType='" & M3_Home.lblUserType2.Text &
"'"
        Dim cmd3 As MySqlCommand = New MySqlCommand(theQuery3, con)
        If con.State = ConnectionState.Open Then
            con.Close()
        End If
        con.Open()
        tbAns.Text = cmd3.ExecuteScalar()
        con.Close()
        Dim theQuery4 As String = "SELECT UserType FROM tblaccounts where
UserName='" & M3_Home.lblUserName2.Text & "'" _
                            & "AND UserType='" & M3_Home.lblUserType2.Text &
"'"
        Dim cmd4 As MySqlCommand = New MySqlCommand(theQuery4, con)
        If con.State = ConnectionState.Open Then
            con.Close()
        End If
        con.Open()
        cbUserType.Text = cmd4.ExecuteScalar()
        con.Close()

        Dim lt As String = "select UserName from tblloggedin"
        Dim da As New MySqlDataAdapter(lt, con)
        If con.State = ConnectionState.Open Then
            con.Close()
        End If
        con.Open()
        Dim dt As New DataTable
        da.Fill(dt)
        cbUserName.DataSource = dt
        cbUserName.DisplayMember = "UserName"
        cbUserName.ValueMember = "UserName"
        con.Close()
```

```vb
    End Sub

    Sub fillComboBox()
        Dim lt As String = "select Name from tblstaff Order by Name"
        Dim da As New MySqlDataAdapter(lt, con)
        If con.State = ConnectionState.Open Then
            con.Close()
        End If
        con.Open()
        Dim dt As New DataTable
        da.Fill(dt)
        cbUserName.DataSource = dt
        cbUserName.DisplayMember = "Name"
        cbUserName.ValueMember = "Name"
        con.Close()
    End Sub

    Public Function convertQuotes(ByVal str As String) As String
        convertQuotes = str.Replace("'", "''")
    End Function

    Private Sub btnUpdate_Click(sender As Object, e As EventArgs) Handles
btnUpdate.Click
        If cbUserType.Text = Nothing Or cbUserName.Text = Nothing Or
tbPassword.Text = Nothing Or tbConfirmPass.Text = Nothing _
            Or cbQuestion.Text = Nothing Or tbAns.Text = Nothing Then
            MsgBox("All fields required!", MsgBoxStyle.Information, "Invalid")
            Return
        Else
            updateAcc()
            Dim del As String = "Delete from tblloggedin"
            Dim cmd6 As New MySqlCommand(del, con)
            If con.State = ConnectionState.Open Then
                con.Close()
            End If
            con.Open()
            cmd6.ExecuteNonQuery()
            con.Close()
            M3_Home.lblUserType2.Visible = False
            M3_Home.lblUserName2.Visible = False
            M3_Home.llManageAcc.Visible = False
        End If
    End Sub

    Private Sub tbConfirmPass_Leave(sender As Object, e As EventArgs) Handles
tbConfirmPass.Leave
        If tbConfirmPass.Text <> tbPassword.Text Then
            MsgBox("Passwords don't match!", MsgBoxStyle.Information, "System
Checked")
            tbConfirmPass.Text = Nothing
        End If
    End Sub

    Private Sub btnSave_Click(sender As Object, e As EventArgs) Handles
btnSave.Click
        If cbUserType.Text = Nothing Or cbUserName.Text = Nothing Or
tbPassword.Text = Nothing Or tbConfirmPass.Text = Nothing _
            Or cbQuestion.Text = Nothing Or tbAns.Text = Nothing Then
```

```vbnet
                MsgBox("All fields required!", MsgBoxStyle.Information, "Invalid")
                Return
            Else
                saveAcc()
            End If
        End Sub

        Private Sub M3_ManageAccount_Load(sender As Object, e As EventArgs) Handles
MyBase.Load
            cbUserType.SelectedIndex = 0
            Dim theQuery3 As String = "SELECT UserType FROM tblloggedin"
            Dim cmd3 As MySqlCommand = New MySqlCommand(theQuery3, con)
            If con.State = ConnectionState.Open Then
                con.Close()
            End If
            con.Open()
            Dim userType = cmd3.ExecuteScalar()
            con.Close()
            If userType = "Admin" Then
                cbUserName.DropDownStyle = ComboBoxStyle.DropDown
            End If
        End Sub

        Private Sub btnRemove_Click(sender As Object, e As EventArgs) Handles
btnRemove.Click
            If MessageBox.Show("Remove Account?", "System Checked",
MessageBoxButtons.YesNo, MessageBoxIcon.Question) = Windows.Forms.DialogResult.Yes
Then
                Dim lt As String = "select COUNT(UserType) from tblaccounts WHERE
UserType = 'Admin'"
                Dim cmd1 As MySqlCommand = New MySqlCommand(lt, con)
                If con.State = ConnectionState.Open Then
                    con.Close()
                End If
                con.Open()
                Dim count = cmd1.ExecuteScalar
                If cbUserType.Text = "Admin" And count <= 1 Then
                    MsgBox("Must have at least one Admin account!",
MsgBoxStyle.Information, "Cannot Remove!")
                    Return
                Else
                    Dim del As String = "Delete from tblaccounts WHERE UserType='" &
cbUserType.Text & "' AND UserName='" & cbUserName.Text & "'"
                    Dim cmd As New MySqlCommand(del, con)
                    cmd.ExecuteNonQuery()
                    MsgBox("Removed Successfully!", MsgBoxStyle.Information, "System
Checked")
                    Me.Close()
                    M3_Home.lblUserType2.Visible = False
                    M3_Home.lblUserName2.Visible = False
                    M3_Home.llManageAcc.Visible = False
                    M3_Home.llLogOut.Visible = False
                End If
                con.Close()
            End If
        End Sub
```

```vbnet
    Private Sub cbUserType_Leave(sender As Object, e As EventArgs) Handles
cbUserType.Leave
        Dim theQuery As String = "SELECT UserType FROM tblloggedin"
        Dim cmd As MySqlCommand = New MySqlCommand(theQuery, con)
        If con.State = ConnectionState.Open Then
            con.Close()
        End If
        con.Open()
        Dim userType = cmd.ExecuteScalar()
        con.Close()
        If userType <> "Admin" And cbUserType.Text = "Admin" Then
            MsgBox("Please log-in as Admin to create Admin account!",
MsgBoxStyle.Information, "No Privilege!")
            cbUserType.Text = Nothing
        End If
    End Sub

    Private Sub cbUserType_SelectedIndexChanged(sender As Object, e As EventArgs)
Handles cbUserType.SelectedIndexChanged
        If cbUserType.Text = "User" Then
            cbUserName.DropDownStyle = ComboBoxStyle.DropDownList
        Else
            cbUserName.DropDownStyle = ComboBoxStyle.DropDown
        End If
    End Sub
End Class
```

Notifications Form

```vbnet
Imports MySql.Data.MySqlClient
Public Class M3_Notifications
    Sub expiredStocksNotify()
        Dim lt As String = "select COUNT(ExpirationDate) from tblstock WHERE
DATEDIFF(ExpirationDate,CURDATE()) <= 0"
        Dim cmd1 As MySqlCommand = New MySqlCommand(lt, con)
        con.Open()
        Dim count = cmd1.ExecuteScalar
        lblExpired.Text = count
        con.Close()
    End Sub

    Sub closeToExpiryNotify()
        Dim lt As String = "select COUNT(ExpirationDate) from tblstock WHERE
DATEDIFF(ExpirationDate,CURDATE()) > 0 AND DATEDIFF(ExpirationDate,CURDATE()) <
90"
        Dim cmd1 As MySqlCommand = New MySqlCommand(lt, con)
        con.Open()
        Dim count = cmd1.ExecuteScalar
        lblCloseToExp.Text = count
        con.Close()
    End Sub

    Sub criticalStockNotify()
        Dim lt As String = "select COUNT(Quantity) from tblstock WHERE Quantity <
50 AND Quantity > 0"
        Dim cmd1 As MySqlCommand = New MySqlCommand(lt, con)
        con.Open()
        Dim count = cmd1.ExecuteScalar
```

```
            lblCritical.Text = count
            con.Close()
    End Sub

    Sub outOfStockNotify()
        Dim lt As String = "select COUNT(Quantity) from tblstock WHERE Quantity =
0"
        Dim cmd1 As MySqlCommand = New MySqlCommand(lt, con)
        con.Open()
        Dim count = cmd1.ExecuteScalar
        lblOS.Text = count
        con.Close()
    End Sub
    Private Sub M3_Notifications_Load(sender As Object, e As EventArgs) Handles
MyBase.Load
        closeToExpiryNotify()
        expiredStocksNotify()
        criticalStockNotify()
        outOfStockNotify()
    End Sub

    Private Sub llClosetoExp_LinkClicked(sender As Object, e As
LinkLabelLinkClickedEventArgs) Handles llClosetoExp.LinkClicked
        M3_Home.CloseToExpiryToolStripMenuItem2.PerformClick()
    End Sub

    Private Sub llExpired_LinkClicked(sender As Object, e As
LinkLabelLinkClickedEventArgs) Handles llExpired.LinkClicked
        M3_Home.ExpiredStocksToolStripMenuItem.PerformClick()
    End Sub

    Private Sub llCritical_LinkClicked(sender As Object, e As
LinkLabelLinkClickedEventArgs) Handles llCritical.LinkClicked
        M3_Home.CriticalStocksToolStripMenuItem2.PerformClick()
    End Sub

    Private Sub llOS_LinkClicked(sender As Object, e As
LinkLabelLinkClickedEventArgs) Handles llOS.LinkClicked
        M3_Home.OutOfStockToolStripMenuItem2.PerformClick()
    End Sub
End Class
```

Main Form

```
Imports MySql.Data.MySqlClient
Public Class M3_Home
    Sub listStock()
        Dim lt As String = "select ID as ID, GenericName as `Generic Name`,
BrandName as `Brand Name`, Manufacturer as `Manufacturer/ Distributor`," _
                            & "Strength as Strength, DosageForm as `Dosage Form`,
Unit as Unit, NosOrVol as `Nos./Volume`, LotOrBatchNum " _
                            & "as `Lot/Batch No.`, ExpirationDate as `Expiration
Date`, Supplier as Supplier, OriginalPrice as `Acquisition Price`," _
                            & "SellingPrice as `Unit Price`, Quantity as `In-
Stock`, Description as Description, DateAdded as `Date Added`" _
                            & "from tblstock WHERE
DATEDIFF(ExpirationDate,CURDATE()) > 0 AND Quantity > 0 order by GenericName"
        Dim da As New MySqlDataAdapter(lt, con)
```

```vbnet
        If con.State = ConnectionState.Open Then
            con.Close()
        End If
        con.Open()
        Dim ds As New DataSet
        da.Fill(ds, "tblstock")
        da.Dispose()
        dgvStocks.DataSource = ds.Tables(0)
        dgvStocks.Columns(0).Visible = False
        dgvStocks.Columns(6).Visible = False
        dgvStocks.Columns(10).Visible = False
        dgvStocks.Columns(11).Visible = False
        dgvStocks.Columns(14).Visible = False
        dgvStocks.Columns(15).Visible = False
        con.Close()
    End Sub
    Sub checkNewStocks()
        Dim lt As String = "select ID as ID, GenericName as `Generic Name`,
BrandName as `Brand Name`, Manufacturer as  `Manufacturer/ Distributor`, " _
                        & "Strength as Strength, DosageForm as `Dosage Form`,
Unit as Unit, NosOrVol as `Nos./Volume`, LotOrBatchNum " _
                        & "as `Lot/Batch No.`, ExpirationDate as `Expiration
Date`, Supplier as Supplier, OriginalPrice as `Acquisition Price`," _
                        & "SellingPrice as `Unit Price`, Quantity as `In-
Stock`, Description as Description, DateAdded as `Date Added` " _
                        & "from tblstock WHERE DATEDIFF(DateAdded,CURDATE()) <
14 AND DATEDIFF(DateAdded,CURDATE()) >= 0 order by GenericName"
        Dim da As New MySqlDataAdapter(lt, con)
        If con.State = ConnectionState.Open Then
            con.Close()
        End If
        con.Open()
        Dim ds As New DataSet
        da.Fill(ds, "tblstock")
        da.Dispose()
        dgvStocks.DataSource = ds.Tables(0)
        dgvStocks.Columns(0).Visible = False
        dgvStocks.Columns(6).Visible = False
        dgvStocks.Columns(8).Visible = False
        dgvStocks.Columns(9).Visible = False
        dgvStocks.Columns(10).Visible = False
        dgvStocks.Columns(11).Visible = False
        dgvStocks.Columns(14).Visible = False
        dgvStocks.Columns(15).Visible = False
        con.Close()
        chbLot.Checked = False
        chbExpDate.Checked = False
    End Sub

    Sub checkAllCheckBox()
        chbGenName.Checked = True
        chbBrandN.Checked = True
        chbManu.Checked = True
        chbStrength.Checked = True
        chbDosF.Checked = True
        chbDosF.Checked = True
        chbNosOrVol.Checked = True
        chbLot.Checked = True
```

```vbnet
            chbExpDate.Checked = True
            chbUPr.Checked = True
            chbQty.Checked = True
        End Sub

    Sub saveToCart()
        Dim ID As Integer
        ID = dgvStocks.CurrentRow.Cells(0).Value
        Dim inStock As Integer
        inStock = dgvStocks.CurrentRow.Cells(13).Value
        Dim inStockBal = inStock - 1
        Dim quantity As Integer
        quantity = 1
        Dim unit As String
        unit = dgvStocks.CurrentRow.Cells(6).Value
        Dim descr As String
        descr = dgvStocks.CurrentRow.Cells(14).Value
        Dim lot As String = dgvStocks.CurrentRow.Cells(8).Value
        Dim expDate As DateTime = dgvStocks.CurrentRow.Cells(9).Value
        Dim unitPrice As Double
        unitPrice = dgvStocks.CurrentRow.Cells(12).Value
        Dim totalPrice As Double
        totalPrice = dgvStocks.CurrentRow.Cells(12).Value

        If inStockBal < "0" Then
            MsgBox("Out of Stocks!", MsgBoxStyle.Information, "Sytem Checked")
            Return
        Else
            Dim save As String = "Insert into
tblcart(ID,Quantity,Unit,Description,LotOrBatchNum,ExpirationDate,UnitPrice,TotalP
rice)values('" & ID & "','" & quantity & "','" & unit & "','" &
convertQuotes(descr) & "','" & lot & "','" & Format(expDate, "yyyy/MM/dd") & "','"
& unitPrice & "','" & totalPrice & "') ON DUPLICATE KEY UPDATE
Quantity=Quantity+1, TotalPrice=UnitPrice*Quantity"
            Dim cmd1 As New MySqlCommand(save, con)
            If con.State = ConnectionState.Open Then
                con.Close()
            End If
            con.Open()
            cmd1.ExecuteNonQuery()
            con.Close()
        End If

        Dim updInStock As String = "UPDATE tblstock set Quantity = '" & inStockBal
& "' where ID = '" & ID & "'"
        Dim cmd As New MySqlCommand(updInStock, con)
        con.Open()
        cmd.ExecuteNonQuery()
        con.Close()
        chbSelectionState()
    End Sub

    Sub saveToCartForPartition()
        Dim ID As Integer
        ID = dgvStocks.CurrentRow.Cells(0).Value
        Dim inStock As Integer
        inStock = dgvStocks.CurrentRow.Cells(13).Value
        Dim inStockBal = inStock - 1
```

```vb
        Dim quantity As Integer
        quantity = 1
        Dim unit As String
        unit = dgvStocks.CurrentRow.Cells(6).Value
        Dim descr As String
        descr = dgvStocks.CurrentRow.Cells(14).Value
        Dim lot As String = dgvStocks.CurrentRow.Cells(8).Value
        Dim expDate As DateTime = dgvStocks.CurrentRow.Cells(9).Value
        Dim unitPrice As Double
        unitPrice = dgvStocks.CurrentRow.Cells(12).Value
        Dim totalPrice As Double
        totalPrice = dgvStocks.CurrentRow.Cells(12).Value

        If inStockBal < "0" Then
            MsgBox("Out of Stocks!", MsgBoxStyle.Information, "Sytem Checked")
            Return
        Else
            Dim save As String = "Insert into
tblcart_forpartition(ID,Quantity,Unit,Description,LotOrBatchNum,ExpirationDate,Uni
tPrice,TotalPrice)values('" & ID & "','" & quantity & "','" & unit & "','" &
convertQuotes(descr) & "','" & lot & "','" & Format(expDate, "yyyy/MM/dd") & "','"
& unitPrice & "','" & totalPrice & "') ON DUPLICATE KEY UPDATE
Quantity=Quantity+1, TotalPrice=UnitPrice*Quantity"
            Dim cmd1 As New MySqlCommand(save, con)
            If con.State = ConnectionState.Open Then
                con.Close()
            End If
            con.Open()
            cmd1.ExecuteNonQuery()
            con.Close()
        End If

        Dim updInStock As String = "UPDATE tblstock set Quantity = '" & inStockBal
& "' where ID = '" & ID & "'"
        Dim cmd As New MySqlCommand(updInStock, con)
        con.Open()
        cmd.ExecuteNonQuery()
        con.Close()
        listStock()
    End Sub

    Sub dgvContentToAddStock()
        M3_AddStock.lblIDNum.Text = dgvStocks.SelectedCells(0).Value
        M3_AddStock.tbGen.Text = dgvStocks.SelectedCells(1).Value
        M3_AddStock.tbBrand.Text = dgvStocks.SelectedCells(2).Value
        M3_AddStock.tbManu.Text = dgvStocks.SelectedCells(3).Value
        M3_AddStock.tbStrength.Text = dgvStocks.SelectedCells(4).Value
        M3_AddStock.cbDoseForm.Text = dgvStocks.SelectedCells(5).Value
        M3_AddStock.cbUnit.Text = dgvStocks.SelectedCells(6).Value
        M3_AddStock.tbNosVol.Text = dgvStocks.SelectedCells(7).Value
        M3_AddStock.tbLotNum.Text = dgvStocks.SelectedCells(8).Value
        M3_AddStock.dtpExpDate.Text = dgvStocks.SelectedCells(9).Value
        M3_AddStock.cbSupplier.Text = dgvStocks.SelectedCells(10).Value
        M3_AddStock.tbOPrice.Text = dgvStocks.SelectedCells(11).Value
        M3_AddStock.tbSPrice.Text = dgvStocks.SelectedCells(12).Value
        M3_AddStock.tbQty.Text = dgvStocks.SelectedCells(13).Value
        M3_AddStock.rtbDesc.Text = dgvStocks.SelectedCells(14).Value
        M3_AddStock.dtpDateAdded.Text = dgvStocks.SelectedCells(15).Value
```

```vb
    End Sub

    Sub closeToExpiry()
        Dim lt As String = "select ID as ID, GenericName as `Generic Name`,
BrandName as `Brand Name`, Manufacturer as  `Manufacturer/ Distributor`, Strength
as Strength, DosageForm as `Dosage Form`, Unit as Unit, NosOrVol as `Nos./Volume`,
LotOrBatchNum as `Lot/Batch No.`, ExpirationDate as `Expiration Date`, Supplier as
Supplier, OriginalPrice as `Acquisition Price`, SellingPrice as `Unit Price`,
Quantity as `In-Stock`, Description as Description, DateAdded as `Date Added` from
tblstock WHERE DATEDIFF(ExpirationDate,CURDATE()) > 0 AND
DATEDIFF(ExpirationDate,CURDATE()) < 90 order by GenericName"
        Dim da As New MySqlDataAdapter(lt, con)
        If con.State = ConnectionState.Open Then
            con.Close()
        End If
        con.Open()
        Dim ds As New DataSet
        da.Fill(ds, "tblstock")
        da.Dispose()
        dgvStocks.DataSource = ds.Tables(0)
        dgvStocks.Columns(0).Visible = False
        dgvStocks.Columns(3).Visible = False
        dgvStocks.Columns(6).Visible = False
        dgvStocks.Columns(11).Visible = False
        dgvStocks.Columns(14).Visible = False
        dgvStocks.Columns(15).Visible = False
        con.Close()
        chbManu.Checked = False
    End Sub

    Sub expiredStocks()
        Dim lt As String = "select ID as ID, GenericName as `Generic Name`,
BrandName as `Brand Name`, Manufacturer as  `Manufacturer/ Distributor`, Strength
as Strength, DosageForm as `Dosage Form`, Unit as Unit, NosOrVol as `Nos./Volume`,
LotOrBatchNum as `Lot/Batch No.`, ExpirationDate as `Expiration Date`, Supplier as
Supplier, OriginalPrice as `Acquisition Price`, SellingPrice as `Unit Price`,
Quantity as `In-Stock`, Description as Description, DateAdded as `Date Added` from
tblstock WHERE DATEDIFF(ExpirationDate,CURDATE()) <= 0 order by GenericName"
        Dim da As New MySqlDataAdapter(lt, con)
        If con.State = ConnectionState.Open Then
            con.Close()
        End If
        con.Open()
        Dim ds As New DataSet
        da.Fill(ds, "tblstock")
        da.Dispose()
        dgvStocks.DataSource = ds.Tables(0)
        dgvStocks.Columns(0).Visible = False
        dgvStocks.Columns(3).Visible = False
        dgvStocks.Columns(6).Visible = False
        dgvStocks.Columns(11).Visible = False
        dgvStocks.Columns(14).Visible = False
        dgvStocks.Columns(15).Visible = False
        con.Close()
        chbManu.Checked = False
    End Sub

    Sub chbSelectionState()
```

```vb
        If NewlyAddedToolStripMenuItem.Checked = True Then
            checkNewStocks()
        ElseIf CloseToExpiryToolStripMenuItem2.Checked = True Then
            closeToExpiry()
        ElseIf ExpiredStocksToolStripMenuItem.Checked = True Then
            expiredStocks()
        ElseIf CriticalStocksToolStripMenuItem2.Checked = True Then
            criticalStocks()
        ElseIf OutOfStockToolStripMenuItem2.Checked = True Then
            outOfStock()
        Else
            listStock()
        End If
    End Sub

    Sub searchStocks()
        Dim temp As Double = 0
        Dim lt As String = "select ID as ID, GenericName as `Generic Name`,
BrandName as `Brand Name`, Manufacturer as  `Manufacturer/ Distributor`," _
                         & "Strength as Strength, DosageForm as `Dosage Form`,
Unit as Unit, NosOrVol as `Nos./Volume`, LotOrBatchNum as `Lot/Batch No.`," _
                         & "ExpirationDate as `Expiration Date`, Supplier as
Supplier, OriginalPrice as `Acquisition Price`, SellingPrice as `Unit Price`," _
                         & "Quantity as `In-Stock`, Description as Description,
DateAdded as `Date Added` from tblstock where GenericName like '%" +
convertQuotes(tbSearchStocks.Text) + "%'" _
                         & "Or BrandName like '%" +
convertQuotes(tbSearchStocks.Text) + "%' Or Manufacturer like '%" +
convertQuotes(tbSearchStocks.Text) + "%' order by GenericName"
        Dim da As New MySqlDataAdapter(lt, con)
        con.Open()
        Dim ds As New DataSet
        da.Fill(ds, "tblstock")
        da.Dispose()
        dgvStocks.DataSource = ds.Tables(0)
        con.Close()
    End Sub

    Sub criticalStocks()
        Dim lt As String = "select ID as ID, GenericName as `Generic Name`,
BrandName as `Brand Name`, Manufacturer as  `Manufacturer/ Distributor`, Strength
as Strength, DosageForm as `Dosage Form`, Unit as Unit, NosOrVol as `Nos./Volume`,
LotOrBatchNum as `Lot/Batch No.`, ExpirationDate as `Expiration Date`, Supplier as
Supplier, OriginalPrice as `Acquisition Price`, SellingPrice as `Unit Price`,
Quantity as `In-Stock`, Description as Description, DateAdded as `Date Added`  from
tblstock WHERE Quantity < 50 AND Quantity > 0"
        Dim da As New MySqlDataAdapter(lt, con)
        If con.State = ConnectionState.Open Then
            con.Close()
        End If
        con.Open()
        Dim ds As New DataSet
        da.Fill(ds, "tblstock")
        da.Dispose()
        dgvStocks.DataSource = ds.Tables(0)
        dgvStocks.Columns(0).Visible = False
        dgvStocks.Columns(6).Visible = False
        dgvStocks.Columns(8).Visible = False
```

```vbnet
        dgvStocks.Columns(9).Visible = False
        dgvStocks.Columns(10).Visible = False
        dgvStocks.Columns(11).Visible = False
        dgvStocks.Columns(14).Visible = False
        dgvStocks.Columns(15).Visible = False
        con.Close()
        chbLot.Checked = False
        chbExpDate.Checked = False
    End Sub

    Sub outOfStock()
        Dim lt As String = "select ID as ID, GenericName as `Generic Name`,
BrandName as `Brand Name`, Manufacturer as  `Manufacturer/ Distributor`, Strength
as Strength, DosageForm as `Dosage Form`, Unit as Unit, NosOrVol as `Nos./Volume`,
LotOrBatchNum as `Lot/Batch No.`, ExpirationDate as `Expiration Date`, Supplier as
Supplier, OriginalPrice as `Acquisition Price`, SellingPrice as `Unit Price`,
Quantity as `In-Stock`, Description as Description, DateAdded as `Date Added` from
tblstock WHERE Quantity = 0"
        Dim da As New MySqlDataAdapter(lt, con)
        If con.State = ConnectionState.Open Then
            con.Close()
        End If
        con.Open()
        Dim ds As New DataSet
        da.Fill(ds, "tblstock")
        da.Dispose()
        dgvStocks.DataSource = ds.Tables(0)
        dgvStocks.Columns(0).Visible = False
        dgvStocks.Columns(6).Visible = False
        dgvStocks.Columns(8).Visible = False
        dgvStocks.Columns(9).Visible = False
        dgvStocks.Columns(10).Visible = False
        dgvStocks.Columns(11).Visible = False
        dgvStocks.Columns(12).Visible = False
        dgvStocks.Columns(14).Visible = False
        dgvStocks.Columns(15).Visible = False
        con.Close()
        chbLot.Checked = False
        chbExpDate.Checked = False
        chbUPr.Checked = False
    End Sub

    Sub saveToReservation()
        Dim ID As Integer
        ID = dgvStocks.CurrentRow.Cells(0).Value
        Dim inStock As Integer
        inStock = dgvStocks.CurrentRow.Cells(13).Value
        Dim inStockBal = inStock - 1
        Dim quantity As Integer
        quantity = 1
        Dim unit As String
        unit = dgvStocks.CurrentRow.Cells(6).Value
        Dim descr As String
        descr = dgvStocks.CurrentRow.Cells(14).Value
        Dim lot As String = dgvStocks.CurrentRow.Cells(8).Value
        Dim expDate As DateTime = dgvStocks.CurrentRow.Cells(9).Value
        Dim unitPrice As Double
        unitPrice = dgvStocks.CurrentRow.Cells(12).Value
```

```vbnet
        Dim totalPrice As Double
        totalPrice = dgvStocks.CurrentRow.Cells(12).Value

        If inStockBal < "0" Then
            MsgBox("Out of Stocks!", MsgBoxStyle.Information, "Sytem Checked")
            Return
        Else
            Dim save As String = "Insert into
tblcart_reservation(ID,Quantity,Unit,Description,LotOrBatchNum,ExpirationDate,Unit
Price,TotalPrice)" _
                                & "values('" & ID & "','" & quantity & "','" &
unit & "','" & convertQuotes(descr) & "','" & lot & "'" _
                                & ",'" & Format(expDate, "yyyy/MM/dd") & "','" &
unitPrice & "','" & totalPrice & "') ON DUPLICATE KEY UPDATE " _
                                & "Quantity=Quantity+1,
TotalPrice=UnitPrice*Quantity"
            Dim cmd1 As New MySqlCommand(save, con)
            If con.State = ConnectionState.Open Then
                con.Close()
            End If
            con.Open()
            cmd1.ExecuteNonQuery()
            con.Close()
        End If

        Dim updInStock As String = "UPDATE tblstock set Quantity = '" & inStockBal
& "' where ID = '" & ID & "'"
        Dim cmd As New MySqlCommand(updInStock, con)
        con.Open()
        cmd.ExecuteNonQuery()
        con.Close()
        chbSelectionState()
    End Sub

    Public Function convertQuotes(ByVal str As String) As String
        convertQuotes = str.Replace("'", "''")
    End Function

    Private Sub AddNewToolStripMenuItem_Click(sender As Object, e As EventArgs)
Handles AddNewToolStripMenuItem.Click
        Dim theQuery2 As String = "SELECT UserType FROM tblloggedin"
        Dim cmd2 As MySqlCommand = New MySqlCommand(theQuery2, con)
        If con.State = ConnectionState.Open Then
            con.Close()
        End If
        con.Open()
        Dim userType = cmd2.ExecuteScalar()
        con.Close()
        If userType = "Admin" Or userType = "User" Then
            M3_AddStock.Show()
            M3_AddStock.fillComboBox()
            M3_AddStock.btnUpd.Visible = False
            M3_AddStock.btnDel.Visible = False
        Else
            MsgBox("Please log-in to use this feature!", MsgBoxStyle.Information,
"No access!")
            Return
        End If
```

```vb
    End Sub

    Private Sub StocksToolStripMenuItem_DropDownClosed(sender As Object, e As
EventArgs) Handles StocksToolStripMenuItem.DropDownClosed
        StocksToolStripMenuItem.ForeColor = Color.White
    End Sub

    Private Sub StocksToolStripMenuItem_DropDownOpening(sender As Object, e As
EventArgs) Handles StocksToolStripMenuItem.DropDownOpening
        StocksToolStripMenuItem.ForeColor = Color.Black
    End Sub

    Private Sub SupplierToolStripMenuItem_Click(sender As Object, e As EventArgs)
Handles SupplierToolStripMenuItem.Click
        M3_AddSupplier.Show()
    End Sub

    Private Sub M3_Home_Load(sender As Object, e As EventArgs) Handles MyBase.Load
        listStock()
        checkAllCheckBox()
    End Sub

    Private Sub chbGenName_CheckedChanged(sender As Object, e As EventArgs)
Handles chbGenName.CheckedChanged
        If chbGenName.Checked = True Then
            dgvStocks.Columns(1).Visible = True
            chbSelectionState()
        ElseIf chbGenName.Checked = False Then
            dgvStocks.Columns(1).Visible = False
            chbSelectionState()
        End If
    End Sub

    Private Sub chbBrandN_CheckedChanged(sender As Object, e As EventArgs) Handles
chbBrandN.CheckedChanged
        If chbBrandN.Checked = True Then
            dgvStocks.Columns(2).Visible = True
            chbSelectionState()
        ElseIf chbBrandN.Checked = False Then
            dgvStocks.Columns(2).Visible = False
            chbSelectionState()
        End If
    End Sub

    Private Sub chbManu_CheckedChanged(sender As Object, e As EventArgs) Handles
chbManu.CheckedChanged
        If chbManu.Checked = True Then
            dgvStocks.Columns(3).Visible = True
            chbSelectionState()
        ElseIf chbManu.Checked = False Then
            dgvStocks.Columns(3).Visible = False
            chbSelectionState()
        End If
    End Sub

    Private Sub chbStrength_CheckedChanged(sender As Object, e As EventArgs)
Handles chbStrength.CheckedChanged
        If chbStrength.Checked = True Then
```

```vbnet
                dgvStocks.Columns(4).Visible = True
                chbSelectionState()
        ElseIf chbStrength.Checked = False Then
                dgvStocks.Columns(4).Visible = False
                chbSelectionState()
        End If
    End Sub

    Private Sub chbDosF_CheckedChanged(sender As Object, e As EventArgs) Handles
chbDosF.CheckedChanged
        If chbDosF.Checked = True Then
                dgvStocks.Columns(5).Visible = True
                chbSelectionState()
        ElseIf chbDosF.Checked = False Then
                dgvStocks.Columns(5).Visible = False
                chbSelectionState()
        End If
    End Sub

    Private Sub chbNosOrVol_CheckedChanged(sender As Object, e As EventArgs)
Handles chbNosOrVol.CheckedChanged
        If chbNosOrVol.Checked = True Then
                dgvStocks.Columns(7).Visible = True
                chbSelectionState()
        ElseIf chbNosOrVol.Checked = False Then
                dgvStocks.Columns(7).Visible = False
                chbSelectionState()
        End If
    End Sub

    Private Sub chbLot_CheckedChanged(sender As Object, e As EventArgs) Handles
chbLot.CheckedChanged
        If chbLot.Checked = True Then
                dgvStocks.Columns(8).Visible = True
                chbSelectionState()
        ElseIf chbLot.Checked = False Then
                dgvStocks.Columns(8).Visible = False
                chbSelectionState()
        End If
    End Sub

    Private Sub chbExpDate_CheckedChanged(sender As Object, e As EventArgs)
Handles chbExpDate.CheckedChanged
        If chbExpDate.Checked = True Then
                dgvStocks.Columns(9).Visible = True
                chbSelectionState()
        ElseIf chbExpDate.Checked = False Then
                dgvStocks.Columns(9).Visible = False
                chbSelectionState()
        End If
    End Sub

    Private Sub chbUPr_CheckedChanged(sender As Object, e As EventArgs) Handles
chbUPr.CheckedChanged
        If chbUPr.Checked = True Then
                dgvStocks.Columns(12).Visible = True
                chbSelectionState()
        ElseIf chbUPr.Checked = False Then
```

```vb
                dgvStocks.Columns(12).Visible = False
                chbSelectionState()
            End If
        End Sub

        Private Sub chbQty_CheckedChanged(sender As Object, e As EventArgs) Handles
chbQty.CheckedChanged
            If chbQty.Checked = True Then
                dgvStocks.Columns(13).Visible = True
                chbSelectionState()
            ElseIf chbQty.Checked = False Then
                dgvStocks.Columns(13).Visible = False
                chbSelectionState()
            End If
        End Sub

        Private Sub btnRefresh_Click(sender As Object, e As EventArgs) Handles
btnRefresh.Click
            If chbGenName.Checked = True Or chbBrandN.Checked = True Or
chbManu.Checked = True Or chbStrength.Checked = True Or chbDosF.Checked = True Or
chbDosF.Checked = True _
                Or chbNosOrVol.Checked = True Or chbLot.Checked = True Or
chbExpDate.Checked = True Or chbUPr.Checked = True Or chbQty.Checked = True Then
                listStock()
                chbGenName.Checked = False
                chbBrandN.Checked = False
                chbManu.Checked = False
                chbStrength.Checked = False
                chbDosF.Checked = False
                chbDosF.Checked = False
                chbNosOrVol.Checked = False
                chbLot.Checked = False
                chbExpDate.Checked = False
                chbUPr.Checked = False
                chbQty.Checked = False
            Else
                listStock()
                checkAllCheckBox()
            End If
        End Sub

        Private Sub dgvStocks_CellDoubleClick(sender As Object, e As
DataGridViewCellEventArgs) Handles dgvStocks.CellDoubleClick
            If dgvStocks.RowCount = 0 Then
                MsgBox("No Data!", MsgBoxStyle.Information, "Sytem Checked")
                Return
            ElseIf M3_Dispose.Visible = True Then
                saveToCart()
                M3_Dispose.TopMost = True
                M3_Disposal.totalQtyAmt()
                M3_Dispose.listCart()
                M3_Dispose.TopMost = False
            Else
                Dim theQuery2 As String = "SELECT UserType FROM tblloggedin"
                Dim cmd2 As MySqlCommand = New MySqlCommand(theQuery2, con)
                If con.State = ConnectionState.Open Then
                    con.Close()
                End If
```

```vb
            con.Open()
            Dim userType = cmd2.ExecuteScalar()
            con.Close()
            If userType = "Admin" Or userType = "User" Then
                dgvContentToAddStock()
                M3_AddStock.Show()
                M3_AddStock.lblAdding.Text = "Update/Delete Item..."
                M3_AddStock.fillComboBox()
                M3_AddStock.btnSave.Visible = False
            Else
                MsgBox("Please log-in to make changes to item.",
MsgBoxStyle.Information, "No access!")
                Return
            End If
        End If
    End Sub

    Private Sub CartToolStripMenuItem_Click(sender As Object, e As EventArgs)
Handles CartToolStripMenuItem.Click
        If CartToolStripMenuItem.BackColor = Color.LightBlue Then
            M3_Dispose.Close()
        Else
            Dim theQuery2 As String = "SELECT UserType FROM tblloggedin"
            Dim cmd2 As MySqlCommand = New MySqlCommand(theQuery2, con)
            If con.State = ConnectionState.Open Then
                con.Close()
            End If
            con.Open()
            Dim userType = cmd2.ExecuteScalar()
            con.Close()
            If userType = "Admin" Or userType = "User" Then
                M3_Dispose.Show()
            Else
                MsgBox("Please log-in to use this feature!",
MsgBoxStyle.Information, "No access!")
                Return
            End If
        End If
        If M3_Dispose.Visible = True Then
            CartToolStripMenuItem.BackColor = Color.LightBlue
        End If
    End Sub

    Dim ClickTime As DateTime
    Private Sub dgvStocks_MouseDown(sender As Object, e As MouseEventArgs) Handles
dgvStocks.MouseDown
        ClickTime = Now
    End Sub

    Private Sub dgvStocks_MouseUp(sender As Object, e As MouseEventArgs) Handles
dgvStocks.MouseUp
        If (Now - ClickTime).TotalMilliseconds > 1000 Then
            If lblUserType2.Text = "Admin" Or lblUserType2.Text = "User" Then
                dgvContentToAddStock()
                M3_AddStock.Show()
                M3_AddStock.lblAdding.Text = "Update/Delete Item..."
                M3_AddStock.btnSave.Visible = False
                M3_AddStock.fillComboBox()
```

```vb
            Else
                Return
            End If
        End If
    End Sub

    Private Sub lbl_M3FARMA_Click(sender As Object, e As EventArgs) Handles
lbl_M3FARMA.Click
        listStock()
        NewlyAddedToolStripMenuItem.Checked = False
        CloseToExpiryToolStripMenuItem2.Checked = False
        ExpiredStocksToolStripMenuItem.Checked = False
        CriticalStocksToolStripMenuItem2.Checked = False
        OutOfStockToolStripMenuItem2.Checked = False
    End Sub

    Private Sub tbSearchStocks_TextChanged(sender As Object, e As EventArgs)
Handles tbSearchStocks.TextChanged
        searchStocks()
        NewlyAddedToolStripMenuItem.Checked = False
        CloseToExpiryToolStripMenuItem2.Checked = False
        ExpiredStocksToolStripMenuItem.Checked = False
        CriticalStocksToolStripMenuItem2.Checked = False
        OutOfStockToolStripMenuItem2.Checked = False
    End Sub

    Private Sub btnSearchStocks_Click(sender As Object, e As EventArgs) Handles
btnSearchStocks.Click
        searchStocks()
        NewlyAddedToolStripMenuItem.Checked = False
        CloseToExpiryToolStripMenuItem2.Checked = False
        ExpiredStocksToolStripMenuItem.Checked = False
        CriticalStocksToolStripMenuItem2.Checked = False
        OutOfStockToolStripMenuItem2.Checked = False
    End Sub

    Private Sub StaffsToolStripMenuItem_Click(sender As Object, e As EventArgs)
Handles StaffsToolStripMenuItem.Click
        Dim theQuery2 As String = "SELECT UserType FROM tblloggedin"
        Dim cmd2 As MySqlCommand = New MySqlCommand(theQuery2, con)
        If con.State = ConnectionState.Open Then
            con.Close()
        End If
        con.Open()
        Dim userType = cmd2.ExecuteScalar()
        con.Close()
        If userType = "Admin" Then
            M3_Staff.Show()
        Else
            MsgBox("Please log-in as Admin to use this feature!",
MsgBoxStyle.Information, "No access!")
            Return
        End If
    End Sub

    Private Sub btnLogIn_Click(sender As Object, e As EventArgs) Handles
btnLogIn.Click
        M3_LogIn.Show()
```

```vbnet
    End Sub

    Private Sub llManageAcc_LinkClicked(sender As Object, e As
LinkLabelLinkClickedEventArgs) Handles llManageAcc.LinkClicked
        M3_ManageAccount.Show()
        M3_ManageAccount.accToFormForUpd()
        M3_ManageAccount.btnSave.Visible = False
        M3_ManageAccount.btnUpdate.Visible = True
        M3_ManageAccount.btnRemove.Visible = True
        M3_ManageAccount.cbUserType.Enabled = False
        M3_ManageAccount.cbUserName.Enabled = False
        M3_ManageAccount.cbUserType.Enabled = False
    End Sub

    Private Sub btnNotif_Click(sender As Object, e As EventArgs) Handles
btnNotif.Click
        M3_Notifications.TopMost = True
        M3_Notifications.Show()
        M3_Notifications.TopMost = False
    End Sub

    Private Sub M3_Home_FormClosed(sender As Object, e As FormClosedEventArgs)
Handles MyBase.FormClosed
        Dim del As String = "Delete from tblloggedin"
        Dim cmd6 As New MySqlCommand(del, con)
        If con.State = ConnectionState.Open Then
            con.Close()
        End If
        con.Open()
        cmd6.ExecuteNonQuery()
        con.Close()
    End Sub

    Private Sub M3_Home_Shown(sender As Object, e As EventArgs) Handles
MyBase.Shown
        M3_Notifications.Show()
    End Sub

    Private Sub Timer1_Tick(sender As Object, e As EventArgs) Handles Timer1.Tick
        lblDateTime.Text = "Date: " + Date.Now.ToString("MMMM dd,yyyy") + "  Time:
" + Date.Now.ToString("hh:mm:ss")
    End Sub

    Private Sub ReportsToolStripMenuItem_Click(sender As Object, e As EventArgs)
Handles ReportsToolStripMenuItem.Click
        Dim theQuery2 As String = "SELECT UserType FROM tblloggedin"
        Dim cmd2 As MySqlCommand = New MySqlCommand(theQuery2, con)
        If con.State = ConnectionState.Open Then
            con.Close()
        End If
        con.Open()
        Dim userType = cmd2.ExecuteScalar()
        con.Close()
        If userType = "Admin" Then
            M3_Records.Show()
        Else
            MsgBox("Please log-in as Admin to see records!",
MsgBoxStyle.Information, "No access!")
```

```vbnet
                Return
            End If
        End Sub

        Private Sub llLogOut_LinkClicked(sender As Object, e As
    LinkLabelLinkClickedEventArgs) Handles llLogOut.LinkClicked
            If MessageBox.Show("Log-out Account?", "System Checked",
    MessageBoxButtons.YesNo, MessageBoxIcon.Question) = Windows.Forms.DialogResult.Yes
    Then
                Dim del As String = "Delete from tblloggedin"
                Dim cmd6 As New MySqlCommand(del, con)
                If con.State = ConnectionState.Open Then
                    con.Close()
                End If
                con.Open()
                cmd6.ExecuteNonQuery()
                con.Close()

                lblUserType2.Visible = False
                lblUserName2.Visible = False
                llManageAcc.Visible = False
                llLogOut.Visible = False
            End If
        End Sub
        Private Sub CriticalStocksToolStripMenuItem1_Click(sender As Object, e As
    EventArgs) Handles CriticalStocksToolStripMenuItem1.Click
            M3_Print_CriticalStock.conCrystalReport()
            M3_Print_CriticalStock.Show()
        End Sub

        Private Sub OutOfStockToolStripMenuItem1_Click(sender As Object, e As
    EventArgs) Handles OutOfStockToolStripMenuItem1.Click
            M3_Print_OutOfStock.conCrystalReport()
            M3_Print_OutOfStock.Show()
        End Sub

        Private Sub ExpiredStockToolStripMenuItem_Click(sender As Object, e As
    EventArgs) Handles ExpiredStockToolStripMenuItem.Click
            M3_Print_ExpiredStock.conCrystalReport()
            M3_Print_ExpiredStock.Show()
        End Sub

        Private Sub CloseToExpiryToolStripMenuItem1_Click(sender As Object, e As
    EventArgs) Handles CloseToExpiryToolStripMenuItem1.Click
            M3_Print_CloseToExpiry.conCrystalReport()
            M3_Print_CloseToExpiry.Show()
        End Sub

        Private Sub NewStocksToolStripMenuItem1_Click(sender As Object, e As
    EventArgs) Handles NewStocksToolStripMenuItem1.Click
            M3_Print_NewStock.conCrystalReport()
            M3_Print_NewStock.Show()
        End Sub

        Private Sub NewlyAddedToolStripMenuItem_Click(sender As Object, e As
    EventArgs) Handles NewlyAddedToolStripMenuItem.Click
            If NewlyAddedToolStripMenuItem.Checked = True Then
                ExpiredStocksToolStripMenuItem.Checked = False
```

```vbnet
            CloseToExpiryToolStripMenuItem2.Checked = False
            CriticalStocksToolStripMenuItem2.Checked = False
            OutOfStockToolStripMenuItem2.Checked = False
            checkNewStocks()
            checkNewStocks()
        End If
    End Sub

    Private Sub CriticalStocksToolStripMenuItem2_Click(sender As Object, e As EventArgs) Handles CriticalStocksToolStripMenuItem2.Click
        If CriticalStocksToolStripMenuItem2.Checked = True Then
            NewlyAddedToolStripMenuItem.Checked = False
            CloseToExpiryToolStripMenuItem2.Checked = False
            ExpiredStocksToolStripMenuItem.Checked = False
            OutOfStockToolStripMenuItem2.Checked = False
            criticalStocks()
            criticalStocks()
        End If
    End Sub

    Private Sub OutOfStockToolStripMenuItem2_Click(sender As Object, e As EventArgs) Handles OutOfStockToolStripMenuItem2.Click
        If OutOfStockToolStripMenuItem2.Checked = True Then
            NewlyAddedToolStripMenuItem.Checked = False
            CloseToExpiryToolStripMenuItem2.Checked = False
            ExpiredStocksToolStripMenuItem.Checked = False
            CriticalStocksToolStripMenuItem2.Checked = False
            outOfStock()
            outOfStock()
        End If
    End Sub

    Private Sub CloseToExpiryToolStripMenuItem2_Click(sender As Object, e As EventArgs) Handles CloseToExpiryToolStripMenuItem2.Click
        If CloseToExpiryToolStripMenuItem2.Checked = True Then
            NewlyAddedToolStripMenuItem.Checked = False
            ExpiredStocksToolStripMenuItem.Checked = False
            CriticalStocksToolStripMenuItem2.Checked = False
            OutOfStockToolStripMenuItem2.Checked = False
            closeToExpiry()
            closeToExpiry()
        End If
    End Sub

    Private Sub ExpiredStocksToolStripMenuItem_Click(sender As Object, e As EventArgs) Handles ExpiredStocksToolStripMenuItem.Click
        If ExpiredStocksToolStripMenuItem.Checked = True Then
            NewlyAddedToolStripMenuItem.Checked = False
            CloseToExpiryToolStripMenuItem2.Checked = False
            CriticalStocksToolStripMenuItem2.Checked = False
            OutOfStockToolStripMenuItem2.Checked = False
            expiredStocks()
            expiredStocks()
        End If
    End Sub

    Private Sub TopReleasedToolStripMenuItem_Click(sender As Object, e As EventArgs) Handles TopReleasedToolStripMenuItem.Click
```

```vbnet
        M3_Report_TopReleased.procReceipt()
        M3_Report_TopReleased.Show()
    End Sub

    Private Sub CustomerToolStripMenuItem_Click(sender As Object, e As EventArgs)
Handles CustomerToolStripMenuItem.Click
        M3_Customer.Show()
    End Sub

    Private Sub ReleaseRecordsToolStripMenuItem_Click(sender As Object, e As
EventArgs) Handles ReleaseRecordsToolStripMenuItem.Click
        M3_Records.Show()
    End Sub
End Class
```

Adding Item Form

```vbnet
Imports MySql.Data.MySqlClient
Public Class M3_AddStock
    Sub saveStock()
        Dim dt1 As DateTime = Convert.ToDateTime(dtpExpDate.Text)
        Dim dt2 As DateTime = Date.Now.ToString("dd-MMM-yyyy")
        Dim ts As TimeSpan = dt1.Subtract(dt2)
        Dim theQuery As String = "SELECT * FROM tblstock WHERE
LotOrBatchNum=@LotOrBatchNum"
        Dim cmd1 As MySqlCommand = New MySqlCommand(theQuery, con)
        con.Open()
        cmd1.Parameters.AddWithValue("@LotOrBatchNum", tbLotNum.Text)
        Using reader As MySqlDataReader = cmd1.ExecuteReader()
            If reader.HasRows And tbLotNum.Text <> "-" Then
                MsgBox("Existed!", MsgBoxStyle.Exclamation, "Error")
                Return
            ElseIf UBound(Split(tbOPrice.Text, ".", -1, vbTextCompare)) > 1 Then
                MsgBox("Too many decimals!", MsgBoxStyle.Information, "Invalid
Amount")
                Return
            ElseIf UBound(Split(tbSPrice.Text, ".", -1, vbTextCompare)) > 1 Then
                MsgBox("Too many decimals!", MsgBoxStyle.Information, "Invalid
Amount")
                Return
            ElseIf Convert.ToInt32(ts.Days) <= 0 Then
                MsgBox("Please enter expiration date not less than today's date!",
MsgBoxStyle.Information, "Invalid Expiration Date!")
                Return
            Else
                If MessageBox.Show("Add Stock?", "System Checked",
MessageBoxButtons.YesNo, MessageBoxIcon.Question) = Windows.Forms.DialogResult.Yes
Then
                    Dim save As String = "Insert into
tblstock(GenericName,BrandName,Manufacturer,Strength,DosageForm,Unit,NosOrVol,LotO
rBatchNum,ExpirationDate,Supplier,OriginalPrice,SellingPrice,Quantity,Description,
DateAdded)values('" & convertQuotes(tbGen.Text) & "','" &
convertQuotes(tbBrand.Text) & "','" & convertQuotes(tbManu.Text) & "','" &
convertQuotes(tbStrength.Text) & "','" & convertQuotes(cbDoseForm.Text) & "','" &
cbUnit.Text & "','" & convertQuotes(tbNosVol.Text) & "','" & tbLotNum.Text & "','"
& Format(dtpExpDate.Value, "yyyy/MM/dd") & "','" & convertQuotes(cbSupplier.Text)
& "','" & tbOPrice.Text & "','" & tbSPrice.Text & "','" & tbQty.Text & "','" &
```

```vbnet
convertQuotes(rtbDesc.Text) & "','" & Format(dtpDateAdded.Value, "yyyy/MM/dd") &
"')"
                    Dim cmd As New MySqlCommand(save, con)
                    con.Close()
                    con.Open()
                    cmd.ExecuteNonQuery()
                    MsgBox("Added Successfully!", MsgBoxStyle.Information, "System
Checked")
                    con.Close()
                Else
                    Return
                End If
            End If
        End Using
        con.Close()
        Me.Close()
    End Sub

    Sub fillComboBox()
        Dim lt As String = "select BusinessName from tblsupplier Order by
BusinessName"
        Dim da As New MySqlDataAdapter(lt, con)
        If con.State = ConnectionState.Open Then
            con.Close()
        End If
        con.Open()
        Dim dt As New DataTable
        da.Fill(dt)
        cbSupplier.DataSource = dt
        cbSupplier.DisplayMember = "BusinessName"
        cbSupplier.ValueMember = "BusinessName"
        cbSupplier.Text = ""
        con.Close()
    End Sub
    Sub updateStock()
        Dim up As String = "Update tblstock set GenericName ='" &
convertQuotes(tbGen.Text) & "',BrandName ='" & convertQuotes(tbBrand.Text) &
"',Manufacturer='" & convertQuotes(tbManu.Text) & "',Strength='" &
convertQuotes(tbStrength.Text) & "',DosageForm='" & convertQuotes(cbDoseForm.Text)
& "',Unit='" & cbUnit.Text & "',NosOrVol='" & convertQuotes(tbNosVol.Text) &
"',LotOrBatchNum='" & tbLotNum.Text & "',ExpirationDate='" &
Format(dtpExpDate.Value, "yyyy/MM/dd") & "', Supplier='" &
convertQuotes(cbSupplier.Text) & "',OriginalPrice='" & tbOPrice.Text &
"',SellingPrice='" & tbSPrice.Text & "',Quantity='" & tbQty.Text &
"',Description='" & convertQuotes(rtbDesc.Text) & "',DateAdded='" &
Format(dtpDateAdded.Value, "yyyy/MM/dd") & "' where ID = '" & lblIDNum.Text & "'"
        Dim cmd As New MySqlCommand(up, con)
        If con.State = ConnectionState.Open Then
            con.Close()
        End If
        con.Open()
        cmd.ExecuteNonQuery()
        MsgBox("Changes Added!", MsgBoxStyle.Information, "Success")
        con.Close()
        M3_Home.listStock()
    End Sub

    Sub clearEntries()
```

```vb
        tbGen.Text = Nothing
        tbBrand.Text = Nothing
        tbManu.Text = Nothing
        tbStrength.Text = Nothing
        cbDoseForm.Text = Nothing
        cbUnit.Text = Nothing
        tbNosVol.Text = Nothing
        dtpExpDate.Text = Nothing
        cbSupplier.Text = Nothing
        tbOPrice.Text = Nothing
        tbSPrice.Text = Nothing
        tbQty.Text = Nothing
        rtbDesc.Text = Nothing
        lblIDNum.Text = "#"
    End Sub


    Sub checkDecimal()
        If UBound(Split(tbOPrice.Text, ".", -1, vbTextCompare)) > 1 Then
            MsgBox("Too many decimals!", MsgBoxStyle.Information, "Invalid
Amount")
            Return
        End If
        If UBound(Split(tbSPrice.Text, ".", -1, vbTextCompare)) > 1 Then
            MsgBox("Too many decimals!", MsgBoxStyle.Information, "Invalid
Amount")
            Return
        End If
    End Sub

    Public Function convertQuotes(ByVal str As String) As String
        convertQuotes = str.Replace("'", "''")
    End Function

    Private Sub tbPacks_KeyPress(sender As Object, e As KeyPressEventArgs) Handles
tbQty.KeyPress
        If Not Char.IsDigit(e.KeyChar) And Not Char.IsControl(e.KeyChar) And Not
e.KeyChar = "." Then
            e.Handled = True
        End If
    End Sub

    Private Sub tbTabs_KeyPress(sender As Object, e As KeyPressEventArgs)
        If Not Char.IsDigit(e.KeyChar) And Not Char.IsControl(e.KeyChar) And Not
e.KeyChar = "." Then
            e.Handled = True
        End If
    End Sub

    Private Sub btnAddPacks_Click(sender As Object, e As EventArgs) Handles
btnAddPacks.Click
        If tbQty.Text = Nothing Then
            tbQty.Text = 0
        End If
        If tbQty.Text = 9999 Then
            tbQty.Text = 1
        Else
            Dim packsValue = tbQty.Text
            Dim packsAdded As Integer
```

```vb
            packsAdded = packsValue + 1
            tbQty.Text = packsAdded
        End If
    End Sub

    Private Sub btnMinusPacks_Click(sender As Object, e As EventArgs) Handles
btnMinusPacks.Click
        If tbQty.Text = Nothing Then
            tbQty.Text = 0
        End If
        If tbQty.Text = 0 Then
            tbQty.Text = 9999
        Else
            Dim packsValue = tbQty.Text
            Dim packsMinus As Integer
            packsMinus = packsValue - 1
            tbQty.Text = packsMinus
        End If
    End Sub

    Private Sub tbOPrice_KeyPress(sender As Object, e As KeyPressEventArgs)
Handles tbOPrice.KeyPress
        If Not Char.IsDigit(e.KeyChar) And Not Char.IsControl(e.KeyChar) And Not
e.KeyChar = "." Then
            e.Handled = True
        End If
    End Sub

    Private Sub tbRPrice_KeyPress(sender As Object, e As KeyPressEventArgs)
Handles tbSPrice.KeyPress
        If Not Char.IsDigit(e.KeyChar) And Not Char.IsControl(e.KeyChar) And Not
e.KeyChar = "." Then
            e.Handled = True
        End If
    End Sub

    Private Sub _Leave(sender As Object, e As EventArgs) Handles tbGen.Leave,
tbBrand.Leave, tbManu.Leave,
        tbStrength.Leave, cbDoseForm.Leave, tbNosVol.Leave
        tbGen.Text =
Globalization.CultureInfo.CurrentCulture.TextInfo.ToTitleCase(tbGen.Text)
        tbBrand.Text =
Globalization.CultureInfo.CurrentCulture.TextInfo.ToTitleCase(tbBrand.Text)
        tbManu.Text =
Globalization.CultureInfo.CurrentCulture.TextInfo.ToTitleCase(tbManu.Text)
        Dim gen = tbGen.Text
        Dim brand = tbBrand.Text
        Dim manu = tbManu.Text
        Dim strength = tbStrength.Text
        Dim doseForm = cbDoseForm.Text
        Dim nosOrvol = tbNosVol.Text
        rtbDesc.Text = gen + " " + "(" + brand + ")" + " " + manu + " " + strength
+ " " + doseForm + " " + nosOrvol
    End Sub

    Private Sub btnSave_Click(sender As Object, e As EventArgs) Handles
btnSave.Click
```

```vb
        If (tbGen.Text = Nothing Or tbBrand.Text = Nothing Or cbUnit.Text =
Nothing Or tbLotNum.Text = Nothing Or tbSPrice.Text = Nothing _
            Or tbQty.Text = Nothing) Then
            MsgBox("Fields marked with * are required!", MsgBoxStyle.Information,
"Empty Fields!")
            Return
        Else
            saveStock()
        End If
        M3_Home.listStock()
    End Sub

    Private Sub cbSupplier_MouseHover(sender As Object, e As EventArgs) Handles
cbSupplier.MouseHover
        ttAddStock.SetToolTip(cbSupplier, cbSupplier.Text)
    End Sub

    Private Sub btnUpd_Click(sender As Object, e As EventArgs) Handles
btnUpd.Click
        If (lblIDNum.Text = "#") Then
            MsgBox("Invalid Operation!", MsgBoxStyle.Information, "System
Checked")
            Return
        ElseIf (tbGen.Text = Nothing Or tbBrand.Text = Nothing Or cbUnit.Text =
Nothing Or tbLotNum.Text = Nothing Or tbSPrice.Text = Nothing Or tbQty.Text =
Nothing) Then
            MsgBox("Fields marked with * are required!", MsgBoxStyle.Information,
"Empty Fields!")
            Return
        ElseIf UBound(Split(tbOPrice.Text, ".", -1, vbTextCompare)) > 1 Then
            MsgBox("Too many decimals!", MsgBoxStyle.Information, "Invalid
Amount")
            Return
        ElseIf UBound(Split(tbSPrice.Text, ".", -1, vbTextCompare)) > 1 Then
            MsgBox("Too many decimals!", MsgBoxStyle.Information, "Invalid
Amount")
            Return
        Else
            If MessageBox.Show("Add Changes?", "System Checked",
MessageBoxButtons.YesNo, MessageBoxIcon.Question) = Windows.Forms.DialogResult.Yes
Then
                updateStock()
            End If
        End If
    End Sub

    Private Sub btnDel_Click(sender As Object, e As EventArgs) Handles
btnDel.Click
        If (lblIDNum.Text = "#") Then
            MsgBox("Invalid Operation!", MsgBoxStyle.Information, "System
Checked")
            Return
        End If
        If MessageBox.Show("Delete Record?", "System Checked",
MessageBoxButtons.YesNo, MessageBoxIcon.Question) = Windows.Forms.DialogResult.Yes
Then
            Dim del As String = "Delete from tblstock where ID = '" &
lblIDNum.Text & "'"
```

```vb
            Dim cmd As New MySqlCommand(del, con)
            con.Open()
            cmd.ExecuteNonQuery()
            con.Close()
            M3_Home.listStock()
            Me.Close()
        End If
        clearEntries()
    End Sub

    Private Sub tbOPrice_Leave(sender As Object, e As EventArgs) Handles
tbOPrice.Leave
        checkDecimal()
    End Sub
    Private Sub tbSPrice_Leave(sender As Object, e As EventArgs) Handles
tbSPrice.Leave
        checkDecimal()
    End Sub
End Class
```

Supplier Form

```vb
Imports MySql.Data.MySqlClient
Public Class M3_Customer
    Sub saveCustomer()
        Dim theQuery As String = "SELECT * FROM tblcustomer WHERE Name=@Name And
Address=@Address "
        Dim cmd1 As MySqlCommand = New MySqlCommand(theQuery, con)
        con.Open()
        cmd1.Parameters.AddWithValue("@Name", tbName.Text)
        cmd1.Parameters.AddWithValue("@Address", tbAdd.Text)
        Using reader As MySqlDataReader = cmd1.ExecuteReader()
            If reader.HasRows Then
                MsgBox("Existed!", MsgBoxStyle.Exclamation, "Error")
                Return
            ElseIf tbName.Text = Nothing Or tbAdd.Text = Nothing Then
                MsgBox("Please input Name and Address!", MsgBoxStyle.Exclamation,
"Error")
                Return
            Else
                If MessageBox.Show("Add Customer?", "System Checked",
MessageBoxButtons.YesNo, MessageBoxIcon.Question) = Windows.Forms.DialogResult.Yes
Then
                    Dim save As String = "Insert into
tblcustomer(Name,Address,Contact)values('" & convertQuotes(tbName.Text) & "','" &
convertQuotes(tbAdd.Text) & "','" & tbCon.Text & "')"
                    Dim cmd As New MySqlCommand(save, con)
                    If con.State = ConnectionState.Open Then
                        con.Close()
                    End If
                    con.Open()
                    cmd.ExecuteNonQuery()
                    MsgBox("Saved!", MsgBoxStyle.Information, "System Checked")
                    con.Close()
                End If
            End If
        End Using
        con.Close()
```

```vbnet
            lblCustID.Text = "#"
        End Sub

        Sub updateCust()
            Dim up As String = "Update tblcustomer set Name ='" &
convertQuotes(tbName.Text) & "',Address ='" & convertQuotes(tbAdd.Text) &
"',Contact='" & tbCon.Text & "' where ID = '" & lblCustID.Text & "'"
            Dim cmd As New MySqlCommand(up, con)
            If con.State = ConnectionState.Open Then
                con.Close()
            End If
            con.Open()
            cmd.ExecuteNonQuery()
            MsgBox("Changes Added!", MsgBoxStyle.Information, "Success")
            con.Close()
            listCustomer()
        End Sub

        Sub listCustomer()
            Dim lt As String = "select ID as ID, Name as Name, Address as Address,
Contact as Contact from tblcustomer order by Name"
            Dim da As New MySqlDataAdapter(lt, con)
            If con.State = ConnectionState.Open Then
                con.Close()
            End If
            con.Open()
            Dim ds As New DataSet
            da.Fill(ds, "tblcustomer")
            da.Dispose()
            dgvCustomer.DataSource = ds.Tables(0)
            dgvCustomer.Columns(0).Visible = False
            con.Close()
        End Sub

        Sub clearEntries()
            tbName.Text = Nothing
            tbAdd.Text = Nothing
            tbCon.Text = Nothing
            lblCustID.Text = "#"
        End Sub

        Public Function convertQuotes(ByVal str As String) As String
            convertQuotes = str.Replace("'", "''")
        End Function

        Private Sub btnSave_Click(sender As Object, e As EventArgs) Handles
btnSave.Click
            saveCustomer()
            listCustomer()
            clearEntries()
        End Sub

        Private Sub M3_Customer_Load(sender As Object, e As EventArgs) Handles
MyBase.Load
            listCustomer()
        End Sub
```

```vb
    Private Sub dgvCustomer_CellDoubleClick(sender As Object, e As
DataGridViewCellEventArgs) Handles dgvCustomer.CellDoubleClick
        If dgvCustomer.RowCount = 0 Then
            MsgBox("No Data!", MsgBoxStyle.Information, "Sytem Checked")
            Return
        Else
            lblCustID.Text = dgvCustomer.SelectedCells(0).Value
            tbName.Text = dgvCustomer.SelectedCells(1).Value
            tbAdd.Text = dgvCustomer.SelectedCells(2).Value
            tbCon.Text = dgvCustomer.SelectedCells(3).Value
            Me.Show()
        End If
    End Sub

    Private Sub btnDelete_Click(sender As Object, e As EventArgs) Handles
btnDelete.Click
        If (lblCustID.Text = "#") Then
            MsgBox("Invalid Operation!", MsgBoxStyle.Information, "System
Checked")
            Return
        End If
        If MessageBox.Show("Delete Record?", "System Checked",
MessageBoxButtons.YesNo, MessageBoxIcon.Question) = Windows.Forms.DialogResult.Yes
Then
            Dim del As String = "Delete from tblcustomer where ID = '" &
lblCustID.Text & "'"
            Dim cmd As New MySqlCommand(del, con)
            con.Open()
            cmd.ExecuteNonQuery()
            con.Close()
            listCustomer()
            Me.Show()
            lblCustID.Text = "#"
        End If
        clearEntries()
    End Sub

    Private Sub btnUpdate_Click(sender As Object, e As EventArgs) Handles
btnUpdate.Click
        If (lblCustID.Text = "#") Then
            MsgBox("Invalid Operation!", MsgBoxStyle.Information, "System
Checked")
            Return
        End If
        If (tbName.Text = Nothing) Then
            MsgBox("Name your customer atleast!", MsgBoxStyle.Information, "System
Checked")
            Return
        End If
        Dim theQuery As String = "SELECT Name FROM tblcustomer WHERE ID  = '" &
lblCustID.Text & "'"
        Dim cmd1 As MySqlCommand = New MySqlCommand(theQuery, con)
        If con.State = ConnectionState.Open Then
            con.Close()
        End If
        con.Open()
        Dim custName As String = cmd1.ExecuteScalar()
        con.Close()
```

```vb
            If custName = tbName.Text Then
                If MessageBox.Show("Add Changes?", "System Checked",
MessageBoxButtons.YesNo, MessageBoxIcon.Question) = Windows.Forms.DialogResult.Yes
Then
                    updateCust()
                End If
            End If
            If custName <> tbName.Text Then
                Dim theQuery2 As String = "SELECT * FROM tblcustomer WHERE Name=@Name"
                Dim cmd2 As MySqlCommand = New MySqlCommand(theQuery2, con)
                con.Open()
                cmd2.Parameters.AddWithValue("@Name", tbName.Text)
                Using reader As MySqlDataReader = cmd2.ExecuteReader()
                    If reader.HasRows Then
                        MsgBox("Existed!", MsgBoxStyle.Exclamation, "Error")
                    Else
                        If MessageBox.Show("Add Changes?", "System Checked",
MessageBoxButtons.YesNo, MessageBoxIcon.Question) = Windows.Forms.DialogResult.Yes
Then
                            updateCust()
                        End If
                    End If
                End Using
            End If
    End Sub

    Private Sub tbName_MouseHover(sender As Object, e As EventArgs) Handles
tbName.MouseHover
        ToolTip_M3.SetToolTip(tbName, tbName.Text)
    End Sub

    Private Sub tbAdd_MouseHover(sender As Object, e As EventArgs) Handles
tbAdd.MouseHover
        ToolTip_M3.SetToolTip(tbAdd, tbAdd.Text)
    End Sub

    Private Sub tbCon_MouseHover(sender As Object, e As EventArgs) Handles
tbCon.MouseHover
        ToolTip_M3.SetToolTip(tbCon, tbCon.Text)
    End Sub

    Private Sub lblClear_Click(sender As Object, e As EventArgs) Handles
lblClear.Click
        clearEntries()
    End Sub

    Private Sub tbName_Leave(sender As Object, e As EventArgs) Handles
tbName.Leave
        tbName.Text =
Globalization.CultureInfo.CurrentCulture.TextInfo.ToTitleCase(tbName.Text)
    End Sub

    Private Sub tbAdd_Leave(sender As Object, e As EventArgs) Handles tbAdd.Leave
        tbAdd.Text =
Globalization.CultureInfo.CurrentCulture.TextInfo.ToTitleCase(tbAdd.Text)
    End Sub
End Class
```

Staff Form

```vb
Imports MySql.Data.MySqlClient
Imports System.IO
Imports AForge.Video.DirectShow
Imports AForge.Video
Public Class M3_Staff
    Dim t As Boolean = False
    Dim camera As VideoCaptureDevice
    Sub saveStaff()
        Dim theQuery As String = "SELECT * FROM tblstaff WHERE Name=@Name"
        Dim cmd1 As MySqlCommand = New MySqlCommand(theQuery, con)
        con.Open()
        cmd1.Parameters.AddWithValue("@Name", tbName.Text)
        Using reader As MySqlDataReader = cmd1.ExecuteReader()
            If reader.HasRows Then
                MsgBox("Staff Existed!", MsgBoxStyle.Exclamation, "Error")
                Return
            ElseIf tbName.Text = Nothing Then
                MsgBox("Please input Name atleast!", MsgBoxStyle.Exclamation,
"Error")
                Return
            Else
                If MessageBox.Show("Add Staff?", "System Checked",
MessageBoxButtons.YesNo, MessageBoxIcon.Question) = Windows.Forms.DialogResult.Yes
Then
                    Dim save As String = "Insert into
tblstaff(Name,Address,Contact,Image)values('" & convertQuotes(tbName.Text) & "','"
& convertQuotes(tbAdd.Text) & "','" & tbCon.Text & "',@image_data)"
                    Dim FileSize As UInt64
                    Dim mstream As New System.IO.MemoryStream()
                    pbStaff.Image.Save(mstream,
System.Drawing.Imaging.ImageFormat.Jpeg)
                    Dim arrImage() As Byte = mstream.GetBuffer()
                    FileSize = mstream.Length
                    mstream.Close()
                    Dim cmd As New MySqlCommand(save, con)
                    If con.State = ConnectionState.Open Then
                        con.Close()
                    End If
                    con.Open()
                    cmd.Parameters.AddWithValue("@image_data", arrImage)
                    cmd.ExecuteNonQuery()
                    MsgBox("Saved!", MsgBoxStyle.Information, "System Checked")
                    con.Close()
                End If
            End If
        End Using
        con.Close()
        lblStaffID.Text = "#"
    End Sub

    Sub updateStaff()
        Dim up As String = "Update tblstaff set Name ='" &
convertQuotes(tbName.Text) & "',Address ='" & convertQuotes(tbAdd.Text) &
"',Contact='" & tbCon.Text & "', image=@image_data where ID = '" & lblStaffID.Text
& "'"
        Dim FileSize As UInt64
```

```vbnet
        Dim mstream As New System.IO.MemoryStream()
        pbStaff.Image.Save(mstream, System.Drawing.Imaging.ImageFormat.Jpeg)
        Dim arrImage() As Byte = mstream.GetBuffer()
        FileSize = mstream.Length
        mstream.Close()
        Dim cmd As New MySqlCommand(up, con)
        If con.State = ConnectionState.Open Then
            con.Close()
        End If
        con.Open()
        cmd.Parameters.AddWithValue("@image_data", arrImage)
        cmd.ExecuteNonQuery()
        MsgBox("Changes Added!", MsgBoxStyle.Information, "Success")
        con.Close()
        listStaff()
    End Sub

    Sub listStaff()
        Dim lt As String = "select ID as ID, Name as Name, Address as Address,
Contact as Contact, Image as Image from tblstaff order by Name"
        Dim da As New MySqlDataAdapter(lt, con)
        If con.State = ConnectionState.Open Then
            con.Close()
        End If
        con.Open()
        Dim ds As New DataSet
        da.Fill(ds, "tblstaff")
        da.Dispose()
        dgvStaff.DataSource = ds.Tables(0)
        dgvStaff.Columns(0).Visible = False
        con.Close()
        DirectCast(dgvStaff.Columns("Image"), DataGridViewImageColumn).ImageLayout
= DataGridViewImageCellLayout.Zoom
    End Sub

    Sub clearEntries()
        tbName.Text = Nothing
        tbAdd.Text = Nothing
        tbCon.Text = Nothing
        lblStaffID.Text = "#"
        pbStaff.Image = pbStaff.InitialImage
    End Sub

    Sub captureImage()
        Dim cameras As VideoCaptureDeviceForm = New VideoCaptureDeviceForm
        If cameras.ShowDialog() = Windows.Forms.DialogResult.OK Then
            camera = cameras.VideoDevice
            AddHandler camera.NewFrame, New NewFrameEventHandler(AddressOf
Captured)
            camera.Start()
            t = True
        End If
    End Sub

    Sub closeCam()
        If t = True Then
            camera.Stop()
        End If
```

```vb
    End Sub

    Public Function convertQuotes(ByVal str As String) As String
        convertQuotes = str.Replace("'", "''")
    End Function

    Private Sub btnCustSave_Click(sender As Object, e As EventArgs) Handles
btnSave.Click
        saveStaff()
        listStaff()
        clearEntries()
    End Sub

    Private Sub M3_AddSupplier_Load(sender As Object, e As EventArgs) Handles
MyBase.Load
        listStaff()
    End Sub

    Private Sub dgvStaff_CellMouseDoubleClick(sender As Object, e As
DataGridViewCellMouseEventArgs) Handles dgvStaff.CellMouseDoubleClick
        If dgvStaff.RowCount = 0 Then
            MsgBox("No Data!", MsgBoxStyle.Information, "Sytem Checked")
            Return
        Else
            Dim bytes As [Byte]() = dgvStaff.SelectedCells(4).Value
            Dim ms As New MemoryStream(bytes)
            pbStaff.Image = Image.FromStream(ms)
            lblStaffID.Text = dgvStaff.SelectedCells(0).Value
            tbName.Text = dgvStaff.SelectedCells(1).Value
            tbAdd.Text = dgvStaff.SelectedCells(2).Value
            tbCon.Text = dgvStaff.SelectedCells(3).Value
            Me.Show()
        End If
    End Sub

    Private Sub btnDelete_Click(sender As Object, e As EventArgs) Handles
btnDelete.Click
        If (lblStaffID.Text = "#") Then
            MsgBox("Invalid Operation!", MsgBoxStyle.Information, "System
Checked")
            Return
        End If
        If MessageBox.Show("Delete Record?", "System Checked",
MessageBoxButtons.YesNo, MessageBoxIcon.Question) = Windows.Forms.DialogResult.Yes
Then
            Dim del As String = "Delete from tblstaff where ID = '" &
lblStaffID.Text & "'"
            Dim cmd As New MySqlCommand(del, con)
            con.Open()
            cmd.ExecuteNonQuery()
            con.Close()
            listStaff()
            Me.Show()
            lblStaffID.Text = "#"
        End If
        clearEntries()
    End Sub
```

```vb
    Private Sub btnSuppUpdate_Click(sender As Object, e As EventArgs) Handles
btnUpdate.Click
        If (lblStaffID.Text = "#") Then
            MsgBox("Invalid Operation!", MsgBoxStyle.Information, "System
Checked")
            Return
        End If
        If (tbName.Text = Nothing) Then
            MsgBox("Input name atleast!", MsgBoxStyle.Information, "System
Checked")
            Return
        End If
        Dim theQuery As String = "SELECT Name FROM tblstaff WHERE ID  = '" &
lblStaffID.Text & "'"
        Dim cmd1 As MySqlCommand = New MySqlCommand(theQuery, con)
        If con.State = ConnectionState.Open Then
            con.Close()
        End If
        con.Open()
        Dim staffName As String = cmd1.ExecuteScalar()
        con.Close()
        If staffName = tbName.Text Then
            If MessageBox.Show("Add Changes?", "System Checked",
MessageBoxButtons.YesNo, MessageBoxIcon.Question) = Windows.Forms.DialogResult.Yes
Then
                updateStaff()
            End If
        End If
        If staffName <> tbName.Text Then
            Dim theQuery2 As String = "SELECT * FROM tblstaff WHERE Name=@Name"
            Dim cmd2 As MySqlCommand = New MySqlCommand(theQuery2, con)
            con.Open()
            cmd2.Parameters.AddWithValue("@Name", tbName.Text)
            Using reader As MySqlDataReader = cmd2.ExecuteReader()
                If reader.HasRows Then
                    MsgBox("Existed!", MsgBoxStyle.Exclamation, "Error")
                Else
                    If MessageBox.Show("Add Changes?", "System Checked",
MessageBoxButtons.YesNo, MessageBoxIcon.Question) = Windows.Forms.DialogResult.Yes
Then
                        updateStaff()
                    End If
                End If
            End Using
        End If
    End Sub

    Private Sub tbName_MouseHover(sender As Object, e As EventArgs) Handles
tbName.MouseHover
        ToolTip_M3.SetToolTip(tbName, tbName.Text)
    End Sub

    Private Sub tbAdd_MouseHover(sender As Object, e As EventArgs) Handles
tbAdd.MouseHover
        ToolTip_M3.SetToolTip(tbAdd, tbAdd.Text)
    End Sub
```

```vbnet
    Private Sub tbCon_MouseHover(sender As Object, e As EventArgs) Handles
tbCon.MouseHover
        ToolTip_M3.SetToolTip(tbCon, tbCon.Text)
    End Sub

    Private Sub lblClear_Click(sender As Object, e As EventArgs) Handles
lblClear.Click
        closeCam()
        clearEntries()
    End Sub

    Private Sub tbName_Leave(sender As Object, e As EventArgs) Handles
tbName.Leave
        tbName.Text =
Globalization.CultureInfo.CurrentCulture.TextInfo.ToTitleCase(tbName.Text)
    End Sub

    Private Sub tbAdd_Leave(sender As Object, e As EventArgs) Handles tbAdd.Leave
        tbAdd.Text =
Globalization.CultureInfo.CurrentCulture.TextInfo.ToTitleCase(tbAdd.Text)
    End Sub

    Private Sub pbStaff_Click(sender As Object, e As EventArgs) Handles
pbStaff.Click
        Try
            Dim fopen As New OpenFileDialog
            fopen.FileName = ""
            fopen.Filter = "Image Files
(*.jpg)|*.jpg|(*.jpeg)|*.JPEG|(*.gif)|*.gif|(*.png)|*.png|All Files(*.*)|*.*"
            fopen.ShowDialog()
            pbStaff.Image = System.Drawing.Bitmap.FromFile(fopen.FileName)
            pbStaff.Text = fopen.FileName
        Catch ex As Exception
        End Try
    End Sub

    Private Sub Captured(ByVal sender As Object, ByVal eventargs As
NewFrameEventArgs)
        Dim bmp As Bitmap
        bmp = DirectCast(eventargs.Frame.Clone(), Bitmap)
        pbStaff.Image = DirectCast(eventargs.Frame.Clone(), Bitmap)
    End Sub

    Private Sub btnStartCam_Click(sender As Object, e As EventArgs) Handles
btnStartCam.Click
        captureImage()
    End Sub

    Private Sub btnStopCam_Click(sender As Object, e As EventArgs) Handles
btnStopCam.Click
        closeCam()
    End Sub
End Class
```

Customer Form

```vbnet
Imports MySql.Data.MySqlClient
Public Class M3_AddSupplier
```

```vb
    Sub saveSupplier()
        Dim theQuery As String = "SELECT * FROM tblsupplier WHERE
BusinessName=@BusinessName"
        Dim cmd1 As MySqlCommand = New MySqlCommand(theQuery, con)
        con.Open()
        cmd1.Parameters.AddWithValue("@BusinessName", tbBussName.Text)
        Using reader As MySqlDataReader = cmd1.ExecuteReader()
            If reader.HasRows Then
                MsgBox("Existed!", MsgBoxStyle.Exclamation, "Error")
            Else
                If MessageBox.Show("Add Supplier?", "System Checked",
MessageBoxButtons.YesNo, MessageBoxIcon.Question) = Windows.Forms.DialogResult.Yes
Then
                    Dim save As String = "Insert into
tblsupplier(BusinessName,Owner,Contact)values('" & tbBussName.Text & "','" &
tbOwner.Text & "','" & tbContact.Text & "')"
                    Dim cmd As New MySqlCommand(save, con)
                    con.Close()
                    con.Open()
                    cmd.ExecuteNonQuery()
                    MsgBox("Saved!", MsgBoxStyle.Information, "System Checked")
                    con.Close()
                End If
            End If
        End Using
        con.Close()
        lblSuppID.Text = "#"
    End Sub

    Sub updateSupplier()
        Dim up As String = "Update tblsupplier set BusinessName ='" &
tbBussName.Text & "',Owner ='" & tbOwner.Text & "',Contact='" & tbContact.Text &
"' where SupplierID = '" & lblSuppID.Text & "'"
        Dim cmd As New MySqlCommand(up, con)
        If con.State = ConnectionState.Open Then
            con.Close()
        End If
        con.Open()
        cmd.ExecuteNonQuery()
        MsgBox("Changes Added!", MsgBoxStyle.Information, "Success")
        con.Close()
        listSupplier()
    End Sub

    Sub listSupplier()
        Dim lt As String = "select SupplierID as ID, BusinessName as `Business
Name`, Owner as Owner, Contact as Contact from tblsupplier order by BusinessName"
        Dim da As New MySqlDataAdapter(lt, con)
        con.Open()
        Dim ds As New DataSet
        da.Fill(ds, "tblsupplier")
        da.Dispose()
        dgvSupplier.DataSource = ds.Tables(0)
        dgvSupplier.Columns(0).Visible = False
        con.Close()
    End Sub

    Sub listBankAcc()
```

```vbnet
        Dim supIDMatch As String = "SELECT tblsupplier.SupplierID,
tblsupplier_bankacc.SupplierID, tblsupplier_bankacc.BankID,
tblsupplier_bankacc.BankName as `Bank Name`, tblsupplier_bankacc.AccountName as
`Account Name`, tblsupplier_bankacc.AccountNumber as `Account No.` FROM
tblsupplier, tblsupplier_bankacc WHERE tblsupplier.SupplierID =
tblsupplier_bankacc.SupplierID AND tblsupplier_bankacc.SupplierID = '" &
lblSuppID.Text & "'"
        Dim da As New MySqlDataAdapter(supIDMatch, con)
        con.Open()
        Dim ds As New DataSet
        da.Fill(ds, "tblsupplier_bankacc")
        da.Dispose()
        dgvBankAcc.DataSource = ds.Tables(0)
        dgvBankAcc.Columns(0).Visible = False
        dgvBankAcc.Columns(1).Visible = False
        dgvBankAcc.Columns(2).Visible = False
        con.Close()
    End Sub

    Sub clearEntries()
        tbBussName.Text = Nothing
        tbOwner.Text = Nothing
        tbContact.Text = Nothing
        lblSuppID.Text = "#"
    End Sub

    Sub clearEntriesBank()
        tbBankName.Text = Nothing
        tbBankAccName.Text = Nothing
        tbBankAccNum.Text = Nothing
        lblSuppID.Text = "#"
        lblBankIDNum.Text = "#"
        listBankAcc()
    End Sub

    Sub saveBank()
        Dim theQuery As String = "SELECT * FROM tblsupplier_bankacc"
        Dim cmd1 As MySqlCommand = New MySqlCommand(theQuery, con)
        con.Open()
        If MessageBox.Show("Add Bank Account?", "System Checked",
MessageBoxButtons.YesNo, MessageBoxIcon.Question) = Windows.Forms.DialogResult.Yes
Then
            Dim save As String = "Insert into tblsupplier_bankacc(SupplierID,
BankName, AccountName, AccountNumber)values('" & lblSuppID.Text & "','" &
tbBankName.Text & "','" & tbBankAccName.Text & "','" & tbBankAccNum.Text & "')"
            Dim cmd As New MySqlCommand(save, con)
            con.Close()
            con.Open()
            cmd.ExecuteNonQuery()
            MsgBox("Saved!", MsgBoxStyle.Information, "System Checked")
            con.Close()
        End If
        con.Close()
        lblSuppID.Text = "#"
    End Sub

    Sub updateBank()
```

```vbnet
        If MessageBox.Show("Add Changes?", "System Checked",
MessageBoxButtons.YesNo, MessageBoxIcon.Question) = Windows.Forms.DialogResult.Yes
Then
            Dim up As String = "Update tblsupplier_bankacc set BankName ='" &
tbBankName.Text & "',AccountName ='" & tbBankAccName.Text & "',AccountNumber='" &
tbBankAccNum.Text & "' where BankID = '" & lblBankIDNum.Text & "'"
            Dim cmd As New MySqlCommand(up, con)
            If con.State = ConnectionState.Open Then
                con.Close()
            End If
            con.Open()
            cmd.ExecuteNonQuery()
            MsgBox("Changes Added!", MsgBoxStyle.Information, "Success")
            con.Close()
            listBankAcc()
        End If
    End Sub

    Private Sub tbBankAccNum_KeyPress(sender As Object, e As KeyPressEventArgs)
Handles tbBankAccNum.KeyPress
        If Not Char.IsDigit(e.KeyChar) And Not Char.IsControl(e.KeyChar) And Not
e.KeyChar = "-" Then
            e.Handled = True
        End If
    End Sub

    Private Sub btnSuppSave_Click(sender As Object, e As EventArgs) Handles
btnSuppSave.Click
        If (tbBussName.Text = Nothing) Then
            MsgBox("Name the supplier atleast!", MsgBoxStyle.Information, "System
Checked")
            Return
        End If
        saveSupplier()
        listSupplier()
        clearEntries()
    End Sub

    Private Sub M3_AddSupplier_Load(sender As Object, e As EventArgs) Handles
MyBase.Load
        listSupplier()
        listBankAcc()
    End Sub

    Private Sub dgvSupplier_CellMouseDoubleClick(sender As Object, e As
DataGridViewCellMouseEventArgs) Handles dgvSupplier.CellMouseDoubleClick
        If dgvSupplier.RowCount = 0 Then
            MsgBox("No Data!", MsgBoxStyle.Information, "Sytem Checked")
            Return
        Else
            lblSuppID.Text = dgvSupplier.SelectedCells(0).Value
            tbBussName.Text = dgvSupplier.SelectedCells(1).Value
            tbOwner.Text = dgvSupplier.SelectedCells(2).Value
            tbContact.Text = dgvSupplier.SelectedCells(3).Value
            Me.Show()
            lblBankIDNum.Text = "#"
        End If
    End Sub
```

```vb
    Private Sub btnDelete_Click(sender As Object, e As EventArgs) Handles
btnDelete.Click
        If (lblSuppID.Text = "#") Then
            MsgBox("Invalid Operation!", MsgBoxStyle.Information, "System
Checked")
            Return
        End If
        If MessageBox.Show("Delete Record?", "System Checked",
MessageBoxButtons.YesNo, MessageBoxIcon.Question) = Windows.Forms.DialogResult.Yes
Then
            Dim del As String = "Delete from tblsupplier where SupplierID = '" &
lblSuppID.Text & "'; Delete from tblsupplier_bankacc where SupplierID = '" &
lblSuppID.Text & "'"
            Dim cmd As New MySqlCommand(del, con)
            con.Open()
            cmd.ExecuteNonQuery()
            con.Close()
            listSupplier()
            Me.Show()
            lblSuppID.Text = "#"
        End If
        clearEntries()
    End Sub

    Private Sub btnSuppUpdate_Click(sender As Object, e As EventArgs) Handles
btnSuppUpdate.Click
        If (lblSuppID.Text = "#") Then
            MsgBox("Invalid Operation!", MsgBoxStyle.Information, "System
Checked")
            Return
        End If
        If (tbBussName.Text = Nothing) Then
            MsgBox("Name the supplier atleast!", MsgBoxStyle.Information, "System
Checked")
            Return
        End If
        Dim theQuery As String = "SELECT BusinessName FROM tblsupplier WHERE
SupplierID  = '" & lblSuppID.Text & "'"
        Dim cmd1 As MySqlCommand = New MySqlCommand(theQuery, con)
        If con.State = ConnectionState.Open Then
            con.Close()
        End If
        con.Open()
        lblBName.Text = cmd1.ExecuteScalar()
        con.Close()
        If lblBName.Text = tbBussName.Text Then
            If MessageBox.Show("Add Changes?", "System Checked",
MessageBoxButtons.YesNo, MessageBoxIcon.Question) = Windows.Forms.DialogResult.Yes
Then
                updateSupplier()
            End If
        End If
        If lblBName.Text <> tbBussName.Text Then
            Dim theQuery2 As String = "SELECT * FROM tblsupplier WHERE
BusinessName=@BusinessName"
            Dim cmd2 As MySqlCommand = New MySqlCommand(theQuery2, con)
            con.Open()
```

```vb
                cmd2.Parameters.AddWithValue("@BusinessName", tbBussName.Text)
                Using reader As MySqlDataReader = cmd2.ExecuteReader()
                    If reader.HasRows Then
                        MsgBox("Existed!", MsgBoxStyle.Exclamation, "Error")
                    Else
                        If MessageBox.Show("Add Changes?", "System Checked",
MessageBoxButtons.YesNo, MessageBoxIcon.Question) = Windows.Forms.DialogResult.Yes
Then
                            updateSupplier()
                        End If
                    End If
                End Using
            End If
    End Sub

    Private Sub tbBussName_MouseHover(sender As Object, e As EventArgs) Handles
tbBussName.MouseHover
        ToolTip_M3.SetToolTip(tbBussName, tbBussName.Text)
    End Sub

    Private Sub tbOwner_MouseHover(sender As Object, e As EventArgs) Handles
tbOwner.MouseHover
        ToolTip_M3.SetToolTip(tbOwner, tbOwner.Text)
    End Sub

    Private Sub tbContact_MouseHover(sender As Object, e As EventArgs) Handles
tbContact.MouseHover
        ToolTip_M3.SetToolTip(tbContact, tbContact.Text)
    End Sub

    Private Sub lblClear_Click(sender As Object, e As EventArgs) Handles
lblClear.Click
        clearEntries()
        clearEntriesBank()
    End Sub

    Private Sub btnSaveBank_Click(sender As Object, e As EventArgs) Handles
btnSaveBank.Click
        If (tbBankName.Text = Nothing Or tbBankAccNum.Text = Nothing) Then
            MsgBox("Lacking Relevant Information!", MsgBoxStyle.Information,
"System Checked")
            Return
        End If
        If lblSuppID.Text <> "#" Then
            saveBank()
        Else
            MsgBox("Please chose from the list of suppliers!",
MsgBoxStyle.Information, "System Checked")
        End If
        listBankAcc()
        clearEntriesBank()
        clearEntries()
    End Sub

    Private Sub btnDelBank_Click(sender As Object, e As EventArgs) Handles
btnDelBank.Click
        If (lblBankIDNum.Text = "#") Then
```

```vbnet
            MsgBox("Invalid Operation!", MsgBoxStyle.Information, "System
Checked")
            Return
        End If
        If MessageBox.Show("Remove Account?", "System Checked",
MessageBoxButtons.YesNo, MessageBoxIcon.Question) = Windows.Forms.DialogResult.Yes
Then
            Dim del As String = "Delete from tblsupplier_bankacc where BankID = '"
& lblBankIDNum.Text & "'"
            Dim cmd As New MySqlCommand(del, con)
            con.Open()
            cmd.ExecuteNonQuery()
            con.Close()
            listBankAcc()
            Me.Show()
        End If
        clearEntriesBank()
    End Sub

    Private Sub dgvBankAcc_CellDoubleClick(sender As Object, e As
DataGridViewCellEventArgs) Handles dgvBankAcc.CellDoubleClick
        If dgvBankAcc.RowCount = 0 Then
            MsgBox("No Data!", MsgBoxStyle.Information, "Sytem Checked")
            Return
        Else
            lblSuppID.Text = dgvBankAcc.SelectedCells(0).Value
            lblBankIDNum.Text = dgvBankAcc.SelectedCells(2).Value
            tbBankName.Text = dgvBankAcc.SelectedCells(3).Value
            tbBankAccName.Text = dgvBankAcc.SelectedCells(4).Value
            tbBankAccNum.Text = dgvBankAcc.SelectedCells(5).Value
            Me.Show()
        End If
    End Sub

    Private Sub btnUpdateBank_Click(sender As Object, e As EventArgs) Handles
btnUpdateBank.Click
        If (lblSuppID.Text = "#" Or lblBankIDNum.Text = "#") Then
            MsgBox("Invalid Operation!", MsgBoxStyle.Information, "System
Checked")
            Return
        End If
        If (tbBankName.Text = Nothing Or tbBankAccNum.Text = Nothing) Then
            MsgBox("Lacking Relevant Information!", MsgBoxStyle.Information,
"System Checked")
            Return
        End If
        updateBank()
    End Sub

    Private Sub tbBussName_TextChanged(sender As Object, e As EventArgs) Handles
tbBussName.TextChanged
        listBankAcc()
        Refresh()
    End Sub

    Private Sub lblSuppID_TextChanged(sender As Object, e As EventArgs) Handles
lblSuppID.TextChanged
```

```vbnet
        Dim theQuery As String = "SELECT BusinessName FROM tblsupplier WHERE
SupplierID  = '" & lblSuppID.Text & "'"
        Dim cmd1 As MySqlCommand = New MySqlCommand(theQuery, con)
        If con.State = ConnectionState.Open Then
            con.Close()
        End If
        con.Open()
        lblSelSupp.Text = cmd1.ExecuteScalar()
        con.Close()
    End Sub

    Private Sub lblBankIDNum_TextChanged(sender As Object, e As EventArgs) Handles
lblBankIDNum.TextChanged
        Dim theQuery As String = "SELECT BankName FROM tblsupplier_bankacc WHERE
BankID  = '" & lblBankIDNum.Text & "'"
        Dim theQuery2 As String = "SELECT AccountNumber FROM tblsupplier_bankacc
WHERE BankID  = '" & lblBankIDNum.Text & "'"
        Dim cmd1 As MySqlCommand = New MySqlCommand(theQuery, con)
        Dim cmd2 As MySqlCommand = New MySqlCommand(theQuery2, con)
        If con.State = ConnectionState.Open Then
            con.Close()
        End If
        con.Open()
        Dim selItems As String = cmd1.ExecuteScalar() + "-" + cmd2.ExecuteScalar()
        lblSelBankAcc.Text = selItems
        con.Close()
    End Sub

    Private Sub tbBussName_Leave(sender As Object, e As EventArgs) Handles
tbBussName.Leave
        tbBussName.Text =
Globalization.CultureInfo.CurrentCulture.TextInfo.ToTitleCase(tbBussName.Text)
    End Sub

    Private Sub tbOwner_Leave(sender As Object, e As EventArgs) Handles
tbOwner.Leave
        tbOwner.Text =
Globalization.CultureInfo.CurrentCulture.TextInfo.ToTitleCase(tbOwner.Text)
    End Sub
End Class
```

Release Form

```vbnet
Imports MySql.Data.MySqlClient
Public Class M3_Dispose
    Sub listCart()
        Dim temp As Double = 0
        Dim lt As String = "select ID as ID, Quantity as Quantity, Unit as Unit,
Description as Description, LotOrBatchNum as `Lot/Batch No.`, ExpirationDate as
`Expiration Date`, UnitPrice as `Unit Price`, TotalPrice as `Total Price` from
tblcart order by Description"
        Dim da As New MySqlDataAdapter(lt, con)
        con.Open()
        Dim ds As New DataSet
        da.Fill(ds, "tblcart")
        da.Dispose()
        dgvCart.DataSource = ds.Tables(0)
        dgvCart.Columns(0).Visible = False
```

```vb
            con.Close()
    End Sub

    Sub removeFromCart()
        Dim ID As Integer
        ID = dgvCart.CurrentRow.Cells(0).Value
        Dim quantity As Integer
        quantity = 1
        Dim unit As String
        unit = dgvCart.CurrentRow.Cells(2).Value
        Dim descr As String
        descr = dgvCart.CurrentRow.Cells(3).Value
        Dim lot As String = dgvCart.CurrentRow.Cells(4).Value
        Dim expDate As DateTime = dgvCart.CurrentRow.Cells(5).Value
        Dim unitPrice As Double
        unitPrice = dgvCart.CurrentRow.Cells(6).Value
        Dim totalPrice As Double
        totalPrice = dgvCart.CurrentRow.Cells(7).Value
        Dim save As String = "Insert into
tblcart(ID,Quantity,Unit,Description,LotOrBatchNum,ExpirationDate,UnitPrice,TotalP
rice)values('" & ID & "','" & quantity & "','" & unit & "','" &
convertQuotes(descr) & "','" & lot & "','" & expDate & "','" & unitPrice & "','" &
totalPrice & "')  ON DUPLICATE KEY UPDATE Quantity=Quantity-1,
TotalPrice=UnitPrice*Quantity"
        Dim cmd As New MySqlCommand(save, con)
        con.Open()
        cmd.ExecuteNonQuery()
        con.Close()
    End Sub

    Sub addToStock()
        Dim theQuery As String = "SELECT Quantity FROM tblstock WHERE ID  = '" &
dgvCart.CurrentRow.Cells(0).Value & "'"
        Dim cmd1 As MySqlCommand = New MySqlCommand(theQuery, con)
        If con.State = ConnectionState.Open Then
            con.Close()
        End If
        con.Open()
        lblInStock2.Text = cmd1.ExecuteScalar()
        con.Close()
        Dim inStockCart As Integer = dgvCart.CurrentRow.Cells(1).Value
        Dim inStockBal As Integer = lblInStock2.Text + inStockCart
        Dim updInStock As String = "UPDATE tblstock set Quantity = '" & inStockBal
& "' where ID = '" & dgvCart.CurrentRow.Cells(0).Value & "'"
        Dim cmd2 As New MySqlCommand(updInStock, con)
        con.Open()
        cmd2.ExecuteNonQuery()
        con.Close()
    End Sub

    Sub addToStockDClick()
        Dim theQuery As String = "SELECT Quantity FROM tblstock WHERE ID  = '" &
dgvCart.CurrentRow.Cells(0).Value & "'"
        Dim cmd1 As MySqlCommand = New MySqlCommand(theQuery, con)
        If con.State = ConnectionState.Open Then
            con.Close()
        End If
        con.Open()
```

```vbnet
        lblInStock2.Text = cmd1.ExecuteScalar()
        con.Close()
        Dim inStockBal As Integer = lblInStock2.Text + 1
        Dim updInStock As String = "UPDATE tblstock set Quantity = '" & inStockBal
& "' where ID = '" & dgvCart.CurrentRow.Cells(0).Value & "'"
        Dim cmd2 As New MySqlCommand(updInStock, con)
        con.Open()
        cmd2.ExecuteNonQuery()
        con.Close()
    End Sub

    Sub validQty()
        Dim theQuery As String = "SELECT Quantity FROM tblstock WHERE ID  = '" &
dgvCart.CurrentRow.Cells(0).Value & "'"
        Dim cmd1 As MySqlCommand = New MySqlCommand(theQuery, con)
        If con.State = ConnectionState.Open Then
            con.Close()
        End If
        con.Open()
        lblInStock.Text = cmd1.ExecuteScalar()
        con.Close()
        Dim qtyEdit As Integer
        qtyEdit = dgvCart.CurrentRow.Cells(1).Value
        If qtyEdit > lblInStock.Text Then
            MsgBox("Out of Stocks!", MsgBoxStyle.Information, "Sytem Checked")
            dgvCart.CurrentRow.Cells(1).Value = lblInStock.Text
        End If
    End Sub

    Sub deductAfterQtyEdit()
        Dim theQuery As String = "SELECT Quantity FROM tblstock WHERE ID  = '" &
dgvCart.CurrentRow.Cells(0).Value & "'"
        Dim cmd1 As MySqlCommand = New MySqlCommand(theQuery, con)
        If con.State = ConnectionState.Open Then
            con.Close()
        End If
        con.Open()
        lblStockEEdit.Text = cmd1.ExecuteScalar()
        con.Close()

        Dim endEditQty As Integer = dgvCart.CurrentRow.Cells(1).Value
        Dim inStockEndEdit As Integer = lblStockEEdit.Text - endEditQty

        Dim updEndEdit As String = "UPDATE tblstock set Quantity = '" &
inStockEndEdit & "' where ID = '" & dgvCart.CurrentRow.Cells(0).Value & "'"
        Dim cmd2 As New MySqlCommand(updEndEdit, con)
        If con.State = ConnectionState.Open Then
            con.Close()
        End If
        con.Open()
        cmd2.ExecuteNonQuery()
        con.Close()
    End Sub

    Public Function convertQuotes(ByVal str As String) As String
        convertQuotes = str.Replace("'", "''")
    End Function
```

```vbnet
    Private Sub M3_Cart_FormClosed(sender As Object, e As FormClosedEventArgs)
Handles MyBase.FormClosed
        M3_Home.CartToolStripMenuItem.BackColor = Color.FromArgb(230, 0, 0)
        M3_Home.listStock()
        Dim del As String = "Delete from tblcart"
        Dim cmd As New MySqlCommand(del, con)
        con.Open()
        cmd.ExecuteNonQuery()
        con.Close()
    End Sub

    Private Sub dgvCart_DoubleClick(sender As Object, e As EventArgs) Handles
dgvCart.DoubleClick
        If dgvCart.RowCount = 0 Then
            Return
        Else
            Dim quantity2 As Integer
            quantity2 = dgvCart.CurrentRow.Cells(1).Value
            Dim id As Integer
            id = dgvCart.CurrentRow.Cells(0).Value
            removeFromCart()
            addToStockDClick()
            M3_ReleaseToStaff.totalQtyAmt()
            If quantity2 <= 1 Then
                Dim del As String = "Delete from tblcart where ID = '" & id & "'"
                Dim cmd As New MySqlCommand(del, con)
                con.Open()
                cmd.ExecuteNonQuery()
                con.Close()
            End If
            listCart()
            M3_Home.listStock()
        End If
    End Sub

    Dim ClickTime As DateTime
    Private Sub dgvCart_MouseDown(sender As Object, e As MouseEventArgs) Handles
dgvCart.MouseDown
        ClickTime = Now
    End Sub

    Private Sub dgvCart_MouseUp(sender As Object, e As MouseEventArgs) Handles
dgvCart.MouseUp
        If dgvCart.RowCount = 0 Then
            Return
        Else
            If (Now - ClickTime).TotalMilliseconds > 500 Then
                Dim id As Integer
                id = dgvCart.CurrentRow.Cells(0).Value
                Dim del As String = "Delete from tblcart where ID = '" & id & "'"
                Dim cmd As New MySqlCommand(del, con)
                con.Open()
                cmd.ExecuteNonQuery()
                con.Close()
                addToStock()
                M3_ReleaseToStaff.totalQtyAmt()
                M3_Home.listStock()
                listCart()
```

```vb
            End If
        End If
    End Sub

    Private Sub dgvCart_CellEndEdit(sender As Object, e As
DataGridViewCellEventArgs) Handles dgvCart.CellEndEdit
        validQty()
        Dim ID As Integer
        ID = dgvCart.CurrentRow.Cells(0).Value
        Dim qtyEdit As Integer
        qtyEdit = dgvCart.CurrentRow.Cells(1).Value
        Dim unit As String
        unit = dgvCart.CurrentRow.Cells(2).Value
        Dim descr As String
        descr = dgvCart.CurrentRow.Cells(3).Value
        Dim lot As String = dgvCart.CurrentRow.Cells(4).Value
        Dim expDate As DateTime = dgvCart.CurrentRow.Cells(5).Value
        Dim unitPrice As Double
        unitPrice = dgvCart.CurrentRow.Cells(6).Value
        Dim totalPrice As Double = qtyEdit * unitPrice
        dgvCart.CurrentRow.Cells(7).Value = totalPrice
        Dim save As String = "UPDATE tblcart set ID = '" & ID & "',Quantity = '" &
qtyEdit & "',Unit = '" & unit & "',Description = '" & convertQuotes(descr) &
"',LotOrBatchNum = '" & lot & "',ExpirationDate = '" & Format(expDate,
"yyyy/MM/dd") & "',UnitPrice= '" & unitPrice & "',TotalPrice = '" & totalPrice &
"' where ID = '" & ID & "'"
        Dim cmd As New MySqlCommand(save, con)
        If con.State = ConnectionState.Open Then
            con.Close()
        End If
        con.Open()
        cmd.ExecuteNonQuery()
        con.Close()
        deductAfterQtyEdit()
        M3_ReleaseToStaff.totalQtyAmt()
        M3_Home.listStock()
    End Sub

    Private Sub M3_Cart_FormClosing(sender As Object, e As FormClosingEventArgs)
Handles MyBase.FormClosing
        If dgvCart.RowCount <> 0 Then
            MsgBox("Please clear all items before closing!",
MsgBoxStyle.Information, "Sytem Checked")
            e.Cancel = True
        End If
    End Sub

    Private Sub dgvCart_CellBeginEdit(sender As Object, e As
DataGridViewCellCancelEventArgs) Handles dgvCart.CellBeginEdit
        Dim theQuery As String = "SELECT Quantity FROM tblstock WHERE ID  = '" &
dgvCart.CurrentRow.Cells(0).Value & "'"
        Dim cmd1 As MySqlCommand = New MySqlCommand(theQuery, con)
        If con.State = ConnectionState.Open Then
            con.Close()
        End If
        con.Open()
        lblStockBegEdi.Text = cmd1.ExecuteScalar()
        con.Close()
```

```vbnet
        Dim curBEditQty As Integer = dgvCart.CurrentRow.Cells(1).Value
        Dim inStockfromEdit As Integer = lblStockBegEdi.Text + curBEditQty
        Dim updInStock As String = "UPDATE tblstock set Quantity = '" &
inStockfromEdit & "' where ID = '" & dgvCart.CurrentRow.Cells(0).Value & "'"
        Dim cmd2 As New MySqlCommand(updInStock, con)
        con.Open()
        cmd2.ExecuteNonQuery()
        con.Close()
        M3_Home.listStock()
    End Sub
    Private Sub btnDispose_Click(sender As Object, e As EventArgs) Handles
btnDispose.Click
        If dgvCart.RowCount = 0 Then
            MsgBox("Empty!", MsgBoxStyle.Information, "Sytem Checked")
            Return
        Else
            M3_ReleaseToStaff.Show()
        End If
    End Sub
    Private Sub M3_Cart_Load(sender As Object, e As EventArgs) Handles MyBase.Load
        M3_Home.NewlyAddedToolStripMenuItem.Checked = False
    End Sub
End Class
```

Release To Form

```vbnet
Imports MySql.Data.MySqlClient
Public Class M3_ReleaseToStaff
    Sub stockDisposal()
        If MessageBox.Show("Confirm Release?", "System Checked",
MessageBoxButtons.YesNo, MessageBoxIcon.Question) = Windows.Forms.DialogResult.Yes
Then
            Dim save As String = "Insert into tbldisposal(DateDisposed,
TotalItems, StaffInCharge, Customer, Address)values " _
                                & "('" & lblDateReleased.Text & "','" &
lblTItems.Text & "','" & convertQuotes(lblName.Text) & "','" &
convertQuotes(lblSelCust.Text) & "','" & convertQuotes(lblCusAdd.Text) & "')"
            Dim cmd As New MySqlCommand(save, con)
            con.Open()
            cmd.ExecuteNonQuery()
            MsgBox("Disposed Successfully!", MsgBoxStyle.Information, "System
Checked")
            con.Close()
            M3_Report_Disposal.procReceipt()
            M3_Report_Disposal.Show()
        End If
    End Sub

    Sub totalQtyAmt()
        Dim temp As Double = 0
        Dim colSum As String = "select sum(Quantity) as `Total Items`,
sum(TotalPrice) as `Total Amount` from tblcart"
        Dim da As New MySqlDataAdapter(colSum, con)
        con.Open()
        Dim ds As New DataSet
        da.Fill(ds, "tblcart")
        da.Dispose()
```

```vbnet
        M3_Dispose.dgvTotalItems.DataSource = ds.Tables(0)
        M3_Dispose.dgvTotalItems.Columns(1).Visible = False
        con.Close()
    End Sub

    Sub insToTblDisposed()
        Dim save As String = "INSERT INTO tbldisposed SELECT ID, Description, LotOrBatchNum, Quantity FROM tblcart " _
                             & "ON DUPLICATE KEY UPDATE tbldisposed.Quantity = tblcart.Quantity + tbldisposed.quantity"
        Dim cmd1 As New MySqlCommand(save, con)
        If con.State = ConnectionState.Open Then
            con.Close()
        End If
        con.Open()
        cmd1.ExecuteNonQuery()
        con.Close()
    End Sub

    Sub disposedRecord()
        Dim theQuery As String = "SELECT ID FROM tblcart"
        Dim cmd1 As MySqlCommand = New MySqlCommand(theQuery, con)
        If con.State = ConnectionState.Open Then
            con.Close()
        End If
        con.Open()
        Dim idStock = cmd1.ExecuteScalar()
        con.Close()
        Dim theQuery2 As String = "SELECT UserName FROM tblloggedin"
        Dim cmd2 As MySqlCommand = New MySqlCommand(theQuery2, con)
        If con.State = ConnectionState.Open Then
            con.Close()
        End If
        con.Open()
        Dim representative = cmd2.ExecuteScalar()
        con.Close()
        Dim theQuery3 As String = "SELECT Description FROM tblcart"
        Dim cmd3 As MySqlCommand = New MySqlCommand(theQuery3, con)
        If con.State = ConnectionState.Open Then
            con.Close()
        End If
        con.Open()
        Dim descr = cmd3.ExecuteScalar()
        con.Close()
        Dim theQuery4 As String = "SELECT LotOrBatchNum FROM tblcart"
        Dim cmd4 As MySqlCommand = New MySqlCommand(theQuery4, con)
        If con.State = ConnectionState.Open Then
            con.Close()
        End If
        con.Open()
        Dim lotNum = cmd4.ExecuteScalar()
        con.Close()
        Dim theQuery5 As String = "SELECT Quantity FROM tblcart"
        Dim cmd5 As MySqlCommand = New MySqlCommand(theQuery5, con)
        If con.State = ConnectionState.Open Then
            con.Close()
        End If
        con.Open()
```

```vb
        Dim qty = cmd5.ExecuteScalar()
        con.Close()

        Dim save As String = "INSERT INTO
tbldisposed_record(ID_Stock,Customer,Address,AuthorizedRepresentative,DateReleased
, Description, LotOrBatchNum,Quantity)" _
                                & "SELECT ID, '" & lblSelCust.Text & "','" &
lblCusAdd.Text & "','" & representative & "','" & lblDateReleased.Text & "',
Description, LotOrBatchNum, Quantity FROM tblcart "
        Dim cmd6 As New MySqlCommand(save, con)
        If con.State = ConnectionState.Open Then
            con.Close()
        End If
        con.Open()
        cmd6.ExecuteNonQuery()
        con.Close()
    End Sub

    Sub listCustomer()
        Dim lt As String = "select ID as ID, Name as Name, Address as Address from
tblcustomer order by Name"
        Dim da As New MySqlDataAdapter(lt, con)
        con.Open()
        Dim ds As New DataSet
        da.Fill(ds, "tblcustomer")
        da.Dispose()
        dgvCustomerList.DataSource = ds.Tables(0)
        dgvCustomerList.Columns(0).Visible = False
        con.Close()
    End Sub

    Sub searchCustomer()
        Dim temp As Double = 0
        Dim lt As String = "select ID, Name as Name, Address as Address from
tblcustomer where Name like '%" + tbSearchCustomer.Text + "%' Or Address like '%"
+ tbSearchCustomer.Text + "%' order by Name"
        Dim da As New MySqlDataAdapter(lt, con)
        con.Open()
        Dim ds As New DataSet
        da.Fill(ds, "tblcustomer")
        da.Dispose()
        dgvCustomerList.DataSource = ds.Tables(0)
        con.Close()
    End Sub

    Public Function convertQuotes(ByVal str As String) As String
        convertQuotes = str.Replace("'", "''")
    End Function

    Private Sub M3_Disposal_Load(sender As Object, e As EventArgs) Handles
MyBase.Load
        listCustomer()
        lblDateReleased.Text = Date.Now.ToString("yyyy/MM/dd") + " " +
Date.Now.ToString("hh:mm")
        M3_Dispose.Enabled = False
        M3_Home.Enabled = False

        Dim theQuery As String = "SELECT UserName FROM tblloggedin"
```

```vbnet
        Dim cmd1 As MySqlCommand = New MySqlCommand(theQuery, con)
        If con.State = ConnectionState.Open Then
            con.Close()
        End If
        con.Open()
        Dim name = cmd1.ExecuteScalar()
        con.Close()
        lblName.Text = name

        lblTItems.Text = M3_Dispose.dgvTotalItems.CurrentRow.Cells(0).Value

        Dim del As String = "Delete from tbldisposal"
        Dim cmd As New MySqlCommand(del, con)
        con.Open()
        cmd.ExecuteNonQuery()
        con.Close()
    End Sub

    Private Sub btnSubmit_Click(sender As Object, e As EventArgs) Handles
btnSubmit.Click
        If (lblSelCust.Text = "#") Then
            MsgBox("Double-click on the list to select. If your customer is not on
the list, add new in the Home, Customer Menu.", MsgBoxStyle.Information, "Select a
customer from the list!")
            Return
        Else
            disposedRecord()
            insToTblDisposed()
            stockDisposal()
        End If
    End Sub

    Private Sub M3_Disposal_FormClosed(sender As Object, e As FormClosedEventArgs)
Handles MyBase.FormClosed
        M3_Dispose.Enabled = True
        M3_Home.Enabled = True
    End Sub

    Private Sub tbSearchCustomer_TextChanged(sender As Object, e As EventArgs)
Handles tbSearchCustomer.TextChanged
        searchCustomer()
    End Sub

    Private Sub tbSearchCustomer_Leave(sender As Object, e As EventArgs) Handles
tbSearchCustomer.Leave
        searchCustomer()
    End Sub
    Private Sub btnSearch_Click(sender As Object, e As EventArgs) Handles
btnSearch.Click
        searchCustomer()
    End Sub

    Private Sub dgvCustomerList_DoubleClick(sender As Object, e As EventArgs)
Handles dgvCustomerList.DoubleClick
        lblSelCust.Text = dgvCustomerList.CurrentRow.Cells(1).Value
        lblCusAdd.Text = dgvCustomerList.CurrentRow.Cells(2).Value
    End Sub
End Class
```

Release To Report

```vbnet
Imports MySql.Data.MySqlClient
Public Class M3_Report_Disposal
    Sub procReceipt()
        Dim daCompanies As New MySqlDataAdapter()
        Dim dsCompanies As New DataSet()
        dsCompanies.Clear()
        Dim ds As New DataSet
        Try
            With con
                If .State = ConnectionState.Open Then .Close()
                con.Open()
            End With
        Catch ex As MySqlException
            MsgBox(ex.ToString)
        End Try
        Try
            Dim lt As String = "select * from tblcart"
            Dim da As New MySqlDataAdapter(lt, con)
            Dim lt2 As String = "select * from tbldisposal"
            Dim da2 As New MySqlDataAdapter(lt2, con)
            da.Fill(ds, "tblcart")
            da2.Fill(ds, "tbldisposal")
            da.Dispose()
            da2.Dispose()
        Catch ex As MySqlException
            MsgBox(ex.ToString)
        Finally
            con.Close()
        End Try
        Dim objRpt As New M3_CrystalReport_ReleaseToStaff
        objRpt.SetDataSource(ds)
        crvReceipt.ReportSource = objRpt
        crvReceipt.ReportSource = objRpt
        crvReceipt.RefreshReport()
        crvReceipt.RefreshReport()
    End Sub

    Private Sub M3_Report_Load(sender As Object, e As EventArgs) Handles
MyBase.Load
        Dim del As String = "Delete from tblcart"
        Dim cmd As New MySqlCommand(del, con)
        con.Open()
        cmd.ExecuteNonQuery()
        con.Close()
        M3_Dispose.listCart()
    End Sub

    Private Sub M3_Report_FormClosed(sender As Object, e As FormClosedEventArgs)
Handles MyBase.FormClosed
        M3_Disposal.Close()
    End Sub
End Class
```

Top Released Report

```vbnet
Imports MySql.Data.MySqlClient
```

```vbnet
Public Class M3_Report_TopReleased
    Sub procReceipt()
        Dim daCompanies As New MySqlDataAdapter()
        Dim dsCompanies As New DataSet()
        dsCompanies.Clear()
        Dim ds As New DataSet
        Try
            With con
                If .State = ConnectionState.Open Then .Close()
                con.Open()
            End With
        Catch ex As MySqlException
            MsgBox(ex.ToString)
        End Try
        Try
            Dim lt As String = "select * from tbldisposed"
            Dim da As New MySqlDataAdapter(lt, con)
            da.Fill(ds, "tbldisposed")
            da.Dispose()
        Catch ex As MySqlException
            MsgBox(ex.ToString)
        Finally
            con.Close()
        End Try
        Dim objRpt As New M3_CrystalReport_TopReleased
        objRpt.SetDataSource(ds)
        crvTopReleased.ReportSource = objRpt
        crvTopReleased.RefreshReport()
    End Sub
End Class
```

Release Records

```vbnet
Imports MySql.Data.MySqlClient
Public Class M3_Records
    Sub listDisposedRec()
        Dim lt As String = "select
ID,ID_Stock,Customer,Address,AuthorizedRepresentative as `Authorized
Representative`, DateReleased as `Date Released`, " _
                         & "Description as Description, LotOrBatchNum as
`Lot/Batch No.`, Quantity as Quantity from tbldisposed_record" _
                         & " order by AuthorizedRepresentative"
        Dim da As New MySqlDataAdapter(lt, con)
        If con.State = ConnectionState.Open Then
            con.Close()
        End If
        con.Open()
        Dim ds As New DataSet
        da.Fill(ds, "tbldisposed_record")
        da.Dispose()
        dgvDisposedRec.DataSource = ds.Tables(0)
        dgvDisposedRec.Columns(0).Visible = False
        dgvDisposedRec.Columns(1).Visible = False
        con.Close()
    End Sub

    Sub fillComboBox()
        Dim lt As String = "select UserName from tblaccounts Order by UserName"
```

```vb
        Dim da As New MySqlDataAdapter(lt, con)
        If con.State = ConnectionState.Open Then
            con.Close()
        End If
        con.Open()
        Dim dt As New DataTable
        da.Fill(dt)
        cbRep.DataSource = dt
        cbRep.DisplayMember = "UserName"
        cbRep.ValueMember = "UserName"
        cbRep.Text = ""
        con.Close()
    End Sub

    Sub selectRepOrMonthOrYear()
        Dim lt As String = "select
ID,ID_Stock,Customer,Address,AuthorizedRepresentative as `Authorized
Representative`, DateReleased as `Date Released`, " _
                        & "Description as Description, LotOrBatchNum as
`Lot/Batch No.`, Quantity as Quantity from tbldisposed_record" _
                        & " WHERE AuthorizedRepresentative = '" & cbRep.Text &
"' Or Month(DateReleased) = '" & cbMonth.Text & "' " _
                        & "OR Year(DateReleased) = '" & tbYear.Text & "' order
by AuthorizedRepresentative"
        Dim da As New MySqlDataAdapter(lt, con)
        If con.State = ConnectionState.Open Then
            con.Close()
        End If
        con.Open()
        Dim ds As New DataSet
        da.Fill(ds, "tblstock")
        da.Dispose()
        dgvDisposedRec.DataSource = ds.Tables(0)
        dgvDisposedRec.Columns(0).Visible = False
        dgvDisposedRec.Columns(1).Visible = False
        con.Close()
    End Sub

    Sub searchDesc()
        Dim temp As Double = 0
        Dim lt As String = "select ID,ID_Stock,Customer,Address,
AuthorizedRepresentative as `Authorized Representative`, DateReleased as `Date
Released`, " _
                        & "Description as Description, LotOrBatchNum as
`Lot/Batch No.`, Quantity as Quantity from tbldisposed_record " _
                        & "where Description like '%" +
convertQuotes(tbSearch.Text) + "%' order by Description"
        Dim da As New MySqlDataAdapter(lt, con)
        If con.State = ConnectionState.Open Then
            con.Close()
        End If
        con.Open()
        Dim ds As New DataSet
        da.Fill(ds, "tbldisposed_record")
        da.Dispose()
        dgvDisposedRec.DataSource = ds.Tables(0)
        con.Close()
    End Sub
```

```vbnet
    Sub searchRep()
        Dim temp As Double = 0
        Dim lt As String = "select ID,ID_Stock,Customer,Address,
AuthorizedRepresentative as `Authorized Representative`, DateReleased as `Date
Released`, " _
                            & "Description as Description, LotOrBatchNum as
`Lot/Batch No.`, Quantity as Quantity from tbldisposed_record " _
                            & "where AuthorizedRepresentative like '%" +
convertQuotes(cbRep.Text) + "%' order by AuthorizedRepresentative"
        Dim da As New MySqlDataAdapter(lt, con)
        If con.State = ConnectionState.Open Then
            con.Close()
        End If
        con.Open()
        Dim ds As New DataSet
        da.Fill(ds, "tbldisposed_record")
        da.Dispose()
        dgvDisposedRec.DataSource = ds.Tables(0)
        con.Close()
    End Sub

    Public Function convertQuotes(ByVal str As String) As String
        convertQuotes = str.Replace("'", "''")
    End Function

    Private Sub M3_Reports_Load(sender As Object, e As EventArgs) Handles
MyBase.Load
        listDisposedRec()
        fillComboBox()
    End Sub

    Private Sub btnOk_Click(sender As Object, e As EventArgs) Handles btnOk.Click
        If cbRep.Text IsNot Nothing And cbMonth.Text = Nothing And tbYear.Text =
Nothing Then
            Dim lt As String = "select ID,ID_Stock,Customer,Address,
AuthorizedRepresentative as `Authorized Representative`, DateReleased as `Date
Released`, " _
                            & "Description as Description, LotOrBatchNum as
`Lot/Batch No.`, Quantity as Quantity from tbldisposed_record" _
                            & " WHERE AuthorizedRepresentative = '" & cbRep.Text &
"' order by AuthorizedRepresentative"
            Dim da As New MySqlDataAdapter(lt, con)
            If con.State = ConnectionState.Open Then
                con.Close()
            End If
            con.Open()
            Dim ds As New DataSet
            da.Fill(ds, "tbldisposed_record")
            da.Dispose()
            dgvDisposedRec.DataSource = ds.Tables(0)
            dgvDisposedRec.Columns(0).Visible = False
            dgvDisposedRec.Columns(1).Visible = False
            con.Close()
        ElseIf cbRep.Text = Nothing And cbMonth.Text IsNot Nothing And tbYear.Text
= Nothing Then
            Dim lt As String = "select ID,ID_Stock,Customer,Address,
AuthorizedRepresentative as `Authorized Representative`, DateReleased as `Date
Released`, " _
```

```vbnet
                        & "Description as Description, LotOrBatchNum as
`Lot/Batch No.`, Quantity as Quantity from tbldisposed_record" _
                        & " WHERE Month(DateReleased) = '" & cbMonth.Text & "'
order by AuthorizedRepresentative"
            Dim da As New MySqlDataAdapter(lt, con)
            If con.State = ConnectionState.Open Then
                con.Close()
            End If
            con.Open()
            Dim ds As New DataSet
            da.Fill(ds, "tbldisposed_record")
            da.Dispose()
            dgvDisposedRec.DataSource = ds.Tables(0)
            dgvDisposedRec.Columns(0).Visible = False
            dgvDisposedRec.Columns(1).Visible = False
            con.Close()
        ElseIf cbRep.Text = Nothing And cbMonth.Text = Nothing And tbYear.Text
IsNot Nothing Then
            Dim lt As String = "select ID,ID_Stock,Customer,Address,
AuthorizedRepresentative as `Authorized Representative`, DateReleased as `Date
Released`, " _
                        & "Description as Description, LotOrBatchNum as
`Lot/Batch No.`, Quantity as Quantity from tbldisposed_record" _
                        & " WHERE Year(DateReleased) = '" & tbYear.Text & "'
order by AuthorizedRepresentative"
            Dim da As New MySqlDataAdapter(lt, con)
            If con.State = ConnectionState.Open Then
                con.Close()
            End If
            con.Open()
            Dim ds As New DataSet
            da.Fill(ds, "tbldisposed_record")
            da.Dispose()
            dgvDisposedRec.DataSource = ds.Tables(0)
            dgvDisposedRec.Columns(0).Visible = False
            dgvDisposedRec.Columns(1).Visible = False
            con.Close()
        ElseIf cbRep.Text IsNot Nothing And cbMonth.Text IsNot Nothing And
tbYear.Text = Nothing Then
            Dim lt As String = "select ID,ID_Stock,Customer,Address,
AuthorizedRepresentative as `Authorized Representative`, DateReleased as `Date
Released`, " _
                        & "Description as Description, LotOrBatchNum as
`Lot/Batch No.`, Quantity as Quantity from tbldisposed_record" _
                        & " WHERE AuthorizedRepresentative = '" & cbRep.Text &
"' AND Month(DateReleased) = '" & cbMonth.Text & "' " _
                        & "order by AuthorizedRepresentative"
            Dim da As New MySqlDataAdapter(lt, con)
            If con.State = ConnectionState.Open Then
                con.Close()
            End If
            con.Open()
            Dim ds As New DataSet
            da.Fill(ds, "tbldisposed_record")
            da.Dispose()
            dgvDisposedRec.DataSource = ds.Tables(0)
            dgvDisposedRec.Columns(0).Visible = False
            dgvDisposedRec.Columns(1).Visible = False
```

```vb
                con.Close()
            ElseIf cbRep.Text IsNot Nothing And cbMonth.Text = Nothing And tbYear.Text
IsNot Nothing Then
                Dim lt As String = "select ID,ID_Stock,Customer,Address,
AuthorizedRepresentative as `Authorized Representative`, DateReleased as `Date
Released`, " _
                            & "Description as Description, LotOrBatchNum as
`Lot/Batch No.`, Quantity as Quantity from tbldisposed_record" _
                            & " WHERE AuthorizedRepresentative = '" & cbRep.Text & "'
AND Year(DateReleased) = '" & tbYear.Text & "' " _
                            & "order by AuthorizedRepresentative"
                Dim da As New MySqlDataAdapter(lt, con)
                If con.State = ConnectionState.Open Then
                    con.Close()
                End If
                con.Open()
                Dim ds As New DataSet
                da.Fill(ds, "tbldisposed_record")
                da.Dispose()
                dgvDisposedRec.DataSource = ds.Tables(0)
                dgvDisposedRec.Columns(0).Visible = False
                dgvDisposedRec.Columns(1).Visible = False
                con.Close()
            ElseIf cbRep.Text = Nothing And cbMonth.Text IsNot Nothing And tbYear.Text
IsNot Nothing Then
                Dim lt As String = "select ID,ID_Stock,Customer,Address,
AuthorizedRepresentative as `Authorized Representative`, DateReleased as `Date
Released`, " _
                            & "Description as Description, LotOrBatchNum as `Lot/Batch
No.`, Quantity as Quantity from tbldisposed_record" _
                            & " WHERE Month(DateReleased) = '" & cbMonth.Text & "' AND
Year(DateReleased) = '" & tbYear.Text & "' " _
                            & "order by AuthorizedRepresentative"
                Dim da As New MySqlDataAdapter(lt, con)
                If con.State = ConnectionState.Open Then
                    con.Close()
                End If
                con.Open()
                Dim ds As New DataSet
                da.Fill(ds, "tbldisposed_record")
                da.Dispose()
                dgvDisposedRec.DataSource = ds.Tables(0)
                dgvDisposedRec.Columns(0).Visible = False
                dgvDisposedRec.Columns(1).Visible = False
                con.Close()
            ElseIf cbRep.Text IsNot Nothing And cbMonth.Text IsNot Nothing And
tbYear.Text IsNot Nothing Then
                Dim lt As String = "select ID,ID_Stock,Customer,Address,
AuthorizedRepresentative as `Authorized Representative`, DateReleased as `Date
Released`, " _
                            & "Description as Description, LotOrBatchNum as
`Lot/Batch No.`, Quantity as Quantity from tbldisposed_record" _
                            & " WHERE AuthorizedRepresentative = '" & cbRep.Text &
"' AND Month(DateReleased) = '" & cbMonth.Text & "' " _
                            & "AND Year(DateReleased) = '" & tbYear.Text & "' order
by AuthorizedRepresentative"
                Dim da As New MySqlDataAdapter(lt, con)
                If con.State = ConnectionState.Open Then
```

```vb
                con.Close()
            End If
            con.Open()
            Dim ds As New DataSet
            da.Fill(ds, "tbldisposed_record")
            da.Dispose()
            dgvDisposedRec.DataSource = ds.Tables(0)
            dgvDisposedRec.Columns(0).Visible = False
            dgvDisposedRec.Columns(1).Visible = False
            con.Close()
        End If
    End Sub

    Private Sub lblRecords_Click(sender As Object, e As EventArgs) Handles
lblRecords.Click
        listDisposedRec()
    End Sub

    Private Sub tbSearch_TextChanged(sender As Object, e As EventArgs) Handles
tbSearch.TextChanged
        searchDesc()
    End Sub

    Private Sub btnSearchStocks_Click(sender As Object, e As EventArgs) Handles
btnSearch.Click
        searchDesc()
    End Sub

    Private Sub tbSearch_KeyDown(sender As Object, e As KeyEventArgs) Handles
tbSearch.KeyDown
        If (e.KeyCode = Keys.Enter) Then
            btnSearch.PerformClick()
        End If
    End Sub

    Private Sub cbMonth_KeyPress(sender As Object, e As KeyPressEventArgs) Handles
cbMonth.KeyPress
        If Not Char.IsDigit(e.KeyChar) And Not Char.IsControl(e.KeyChar) Then
            e.Handled = True
        End If
    End Sub

    Private Sub tbYear_KeyPress(sender As Object, e As KeyPressEventArgs) Handles
tbYear.KeyPress
        If Not Char.IsDigit(e.KeyChar) And Not Char.IsControl(e.KeyChar) Then
            e.Handled = True
        End If
    End Sub

    Private Sub cbRep_KeyDown(sender As Object, e As KeyEventArgs) Handles
cbRep.KeyDown
        If (e.KeyCode = Keys.Enter) Then
            btnOk.PerformClick()
        End If
    End Sub

    Private Sub tbYear_KeyDown(sender As Object, e As KeyEventArgs) Handles
tbYear.KeyDown
```

```vbnet
        If (e.KeyCode = Keys.Enter) Then
            btnOk.PerformClick()
        End If
    End Sub

    Private Sub cbMonth_KeyDown(sender As Object, e As KeyEventArgs) Handles
cbMonth.KeyDown
        If (e.KeyCode = Keys.Enter) Then
            btnOk.PerformClick()
        End If
    End Sub

    Private Sub cbRep_TextChanged(sender As Object, e As EventArgs) Handles
cbRep.TextChanged
        searchRep()
    End Sub
    Private Sub dgvDisposedRec_DoubleClick(sender As Object, e As EventArgs)
Handles dgvDisposedRec.DoubleClick
        btnTrash.Visible = True
    End Sub

    Private Sub btnTrash_Click(sender As Object, e As EventArgs) Handles
btnTrash.Click
        Dim id As Integer
        id = dgvDisposedRec.CurrentRow.Cells(0).Value
        Dim del As String = "Delete from tbldisposed_record where ID = '" & id &
"'"
        Dim cmd As New MySqlCommand(del, con)
        con.Open()
        cmd.ExecuteNonQuery()
        con.Close()
        listDisposedRec()
        btnTrash.Visible = False
    End Sub
End Class
```

Print Expired Stock Report

```vbnet
Imports MySql.Data.MySqlClient
Public Class M3_Print_ExpiredStock
    Sub conCrystalReport()
        Dim daCompanies As New MySqlDataAdapter()
        Dim dsCompanies As New DataSet()
        dsCompanies.Clear()
        Dim ds As New DataSet
        Try
            With con
                If .State = ConnectionState.Open Then .Close()
                con.Open()
            End With
        Catch ex As MySqlException
            MsgBox(ex.ToString)
        End Try
        Try
            Dim lt As String = "select * from tblstock"
            Dim da As New MySqlDataAdapter(lt, con)
            da.Fill(ds, "tblstock")
            da.Dispose()
```

```vbnet
        Catch ex As MySqlException
            MsgBox(ex.ToString)
        Finally
            con.Close()
        End Try
        Dim objRpt As New M3_CrystalReport_ExpiredStock
        objRpt.SetDataSource(ds)
        crvExpiredStock.ReportSource = objRpt
        crvExpiredStock.RefreshReport()
    End Sub
End Class
```

Client-Server Connect Form

```vbnet
Imports MySql.Data.MySqlClient
Public Class M3_ServerConnection
    Dim stat As Boolean = False
    Private Sub btnLogIn_Click(sender As Object, e As EventArgs) Handles
btnLogIn.Click
        Try
            con = New MySqlConnection
            con.ConnectionString = "Server=" & server.Text & ";Database=" &
database.Text & ";User ID=" & userid.Text & ";Password=" & password.Text & ""
            con.Open()
            MessageBox.Show("Successfully Connected to Server!", "Confirmation",
MessageBoxButtons.OK, MessageBoxIcon.Information)
            con.Close()
            stat = True
            My.Settings.host = server.Text
            My.Settings.dbname = database.Text
            My.Settings.usid = userid.Text
            My.Settings.pwd = password.Text
            Call setconnection()
            Me.Hide()
            M3_LogIn.Show()
            Exit Sub
        Catch ex As Exception
            ex.ToString()
        End Try
        stat = False
        MessageBox.Show("Connection failed!", "Error", MessageBoxButtons.OK,
MessageBoxIcon.Error)
    End Sub
Private Sub show_CheckStateChanged(sender As Object, e As EventArgs) Handles
show.CheckStateChanged
        If show.Checked = True Then
            password.UseSystemPasswordChar = True
        Else
            password.UseSystemPasswordChar = False
        End If
    End Sub
Private Sub password_KeyDown(sender As Object, e As KeyEventArgs) Handles
password.KeyDown
        If (e.KeyCode = Keys.Enter) Then
            btnLogIn.PerformClick()
        End If
    End Sub
End Class
```

DEVELOPER'S BIODATA

Name   : **Junrey O. Ligtas**

Place of Birth   : **Tagbilaran City**

Date of Birth   : **January 29, 2000**

Age   : **22**

Home Address   : **Campagao, Bilar, Bohol**

Email Address   : **junrey.safe@gmail.com**

Religion   : **Roman Catholic**

Citizenship   : **Filipino**

Father's Name   : **Jacinto G. Ligtas**

Mother's Name   : **Francisca O. Ligtas**
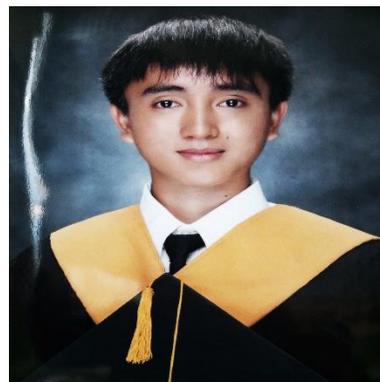
EDUCATIONAL BACKGROUND

Elementary   : **Campagao Elementary School**

                      Campagao,Bilar, Bohol

                      2011-2012

Secondary   : **St. Mary's College of Tagum**

                      Tagum City, Philippines

                      2017-2018

Tertiary   : **Bohol Island State University**

                      Zamora, Bilar, Bohol

                      2021-2022

Degree Earned   : **Bachelor of Science in Computer Science**

Work Experience   : **On-the-Job Training**   : **Sales Representative**

                      BISU-BILAR CAMPUS   **M3 Farma Marketing**

                      Zamora, Bilar, Bohol   **Riverside, Bilar, Bohol**

                      July - October 2021   **Jan – May 2022**

DEVELOPER'S BIODATA

Name   : **Kimberly C. Curay**

Place of Birth : **Bulilis, Ubay, Bohol**

Date of Birth   : **July 27,1999**

Age   : **22**

Home Address   : **Bulilis, Ubay, Bohol**

Email Address   :
curaykimberly1@gmail.com

Religion   : **Roman Catholic**

Citizenship   : **Filipino**

Father's Name   : **Teodoro E. Curay**

Mother's Name   : **Rowena C. Curay**

EDUCATIONAL BACKGROUND

Elementary   : **Bulilis Elementary School**

                  Bulilis, Ubay, Bohol

                  2011-2012

Secondary   : **Bulilis National High School**

                  Bulilis, Ubay, Bohol

                  2017-2018

Tertiary   : **Bohol Island State University**

                  Zamora, Bilar, Bohol

                  2021-2022

Degree Earned   : **Bachelor of Science in Computer Science**

Work Experience : **On-the Job Training**

                  M3 Farma Marketing

                  Riverside, Bilar, Bohol

                  Aug – Sep 2021

DEVELOPER'S BIODATA

Name	: **Larry C. Quilario**
Place of Birth	: **Bugang Sur, Bilar, Bohol**
Date of Birth	: **February 2, 1999**
Age	: **23**
Home Address	: **Bugang Sur, Bilar, Bohhol**
Email Address	: **larryquilario@gmail.com**
Religion	: **New Apostolic**
Citizenship	: **Filipino**
Father's Name	: **Tito C. Quilario**
Mother's Name	: **Conchita C. Quilario**

EDUCATIONAL BACKGROUND

Elementary	: **Batuan Central Elementary School**
	Batuan, Bohol
	2011-2012
Secondary	: **Batuan National High School**
	2017-2018
Tertiary	: **Bohol Island State University**
	Zamora, Bilar, Bohol
	2021-2022
Degree Earned	: **Bachelor of Science in Computer Science**
Work Experience	: **On-the-Job Training**
	LGU Bilar
	Aug - Sep 2021