ASHESI UNIVERSITY COLLEGE

An Inventory Management and Accounting System (IMS)

BY

ROSELINE MILLS

Dissertation submitted to the Department of Computer Science, Ashesi University College.

In partial fulfilment of the requirements for the Bachelor of Science degree in Management Information Systems.

APRIL 2010

Declaration

I hereby declare that the dissertation is the result of my own original work and that no part of it has been presented for another degree in this university or elsewhere.

Candidate's Name:……………………………………………………………………………………

Candidate's Signature:……………………………………………………………………………

Date:………………………………………………………………………………………………

I hereby declare that the preparation and the presentation of the dissertation were supervised in accordance with the guidelines on supervision of dissertation laid down by Ashesi University College.

Supervisor's Name:…………………………………………………………………………………..

Supervisor's Signature:…………………………………………………………………………….

Date:……………………………………………………………………………………………..

Acknowledgements

I am extremely grateful to the Almighty God for his grace and mercies in seeing me through this project.

Secondly, my appreciation goes to my entire family for the immense support they have given me throughout my four year degree programme.

Again my appreciation goes to my supervisor, Mr. Kwadwo Gyamfi Osafo-Maafo, for taking time off his busy schedule to help me complete this project successfully.

Finally, my heartfelt appreciation to Mr. Ekow Koomson, for all the sleepless nights he had to help me debug my codes and through whom I have learnt a great deal.

Abstract

Inventory management and control with its associated financial books preparation/management has been a major source of concern for a good number of businesses over the past years. For most businesses, the inability to effectively monitor and control has been a source of preoccupation with the businesses inevitably incurring huge loses, breaking even, let alone making profits. As a key element in overhead management, inventory management is a major factor in the realization of net profit of these businesses. The rise in the scale of such small scale businesses has made it imperative for a system or mechanism that helps merchants to effectively keep an eye on and control stock hence a way of ensuring profitability.

The project, an inventory management system allows the average business person to monitor and control inventory as well as act as a point of sale system. The system is developed using Microsoft.NET utilizing Seagate Crystal Report and with a MySQL database for storage. It is also networked to enable different users running the application connect to and use a shared database. Overall, the system is designed to increase the user's confidence.

Table of Contents

1.1 Introduction and objective 1

1.2 Motivation 2

Chapter 2 4

2.1 Previous Work 4

2.2 Approach 5

2.2.1 Development Tools 6

2.2.1 Methods 7

Chapter 3 9

3.1 Requirement Analysis and design 9

3.1.1 System Objectives 10

3.1.2 System Context 11

3.1.3 Functional Requirements 11

3.1.4 Non-functional Requirements 14

3.1.5 Future Requirements 16

Chapter 4 17

4.1 System design and implementation 17

Modules 17

4.1.1 Login 18

4.1.2 Main Page 19

4.1.2.1 File 20

4.1.2.2 Maintenance 25

4.1.2.3 Invoicing 30

4.1.3 Employee Password Change 35

4.2 Database Implementation 35

4.3 Testing and evaluation 37

CHAPTER 5 41

5.1 Limitations of the system 41

5.2 Benefits of the system and conclusion 41

Bibliography..43

Appendix...44

Appendix 1: Use Case Diagram of Administrator...44

Appendix 2: Conceptual Database Design...45

Appendix 3: UML diagram of the Inventory database...46

Appendix 4: Logical Design...47

Appendix 5: System architecture design...49

Table of figures

Figure 4.2: Employee Registration form 21

Figure 4.3: Administrator password change........ 24

Figure 4.4: Edit Customer Details form........ 26

Figure 4.5: Remove Product details........ 30

Figure 4.6: Sales form 31

Chapter 1

1.1 Introduction and objective

The problem of being on top of issues regarding stock management in SME's can be a daunting experience for operators or owners of such businesses. Being able to effectively handle pilfering, goods and cash shortages and possibly unproductive employees poses a great challenge. These among other things can eventually contribute to the failure of the company. Indeed ignorance about inventory can be costly. For such SME's, proper books of account are not kept and source documents are in most cases not filed.

Presently through the use of

- simple bin/tally cards
- regular consistent stock taking
- transaction log books

these businesses manage to keep some form of control over their operations. A major challenge for the businesses is the affordability of automated systems that can effectively manage and control stock. Availability and ease of access to technology (software and hardware) is another challenge. Another major challenge is the inability of these operators to clearly define roles of management and or employees; roles usually overlap. An attendant may be the same person who takes stock which is often the case in these situations.

Inventory management has significantly developed in the past years but its application is yet to be experienced in the provision of services by SME's. A good number of these businesses treat inventory as products on the shelf that need to be sold. To many small-business owners, inventory is simply the items they keep on the shelves to generate sales. Despite the investment required to maintain appropriate levels of inventory, they put little effort into monitoring it. [1] This paper focuses on the development, implementation and testing of a software system that would allow users to systematically manage their inventory in order to ensure that, problems that arise because of the mismanagement of inventory in a business are avoided.

1.2 Motivation

While picking up some documents from the boss's house during an internship last year, his wife, a lovely lady who happened to operate a small to medium sized shop that dealt in assorted goods was encountered. In the course of observing attendants as services were provided to customers, it became increasingly clear that the turnaround time per customer, the efforts an attendant had to go through to remember all that had been sold since transactions had to be logged into a transaction book of a sort as well as the seeming difficulty with which products and prices on a shelf were frantically located were amazingly unnecessary and tedious. Efforts were made to get Mrs. Asem to purchase an off- the- shelf inventory/ point of sale system that could aid operations in the business. The prohibitive costs involved and seeming complications of such systems hindered the adoption. As a management information system's student, the idea of developing a system that would not only aid in the management of stock but could also serve as a point of sale system for small to medium scale businesses to fit the average Ghanaian shop was formed. The proposed inventory solution will have more success for the following reasons:

- Reduces turnaround time in processing orders
- Eliminates duplication of duties
- Reduce data input time
- Eliminate costly over stocking or stock outs
- Ease off the time and effort for physical inventory
- Increase employees productivity
- To prevent pilfering
- For accurate cash assessment
- Greatly Reduce the amount of paper work involved

Chapter 2

2.1 Previous Work

Some researchers who have tried to address inventory management problems include Harris, 1913; Beamon and Kotleba, 2006; Buffa and Reynolds, 1977; Landeros and Lyth, 1989; Langley, 1976, 1980; Larson, 1989; as well as Mattsson, 2007. Harris approach to solve the challenges was to create an inventory model that allows a firm to place orders of a particular size when its inventory position reaches a re-order point. The key insights for the reorder level approach was to enable businesses eliminate costly overstocking or stock out. This approach has been successful and is widely accepted and largely used today.

Researchers such as Beamon and Kotleba, 2006; Buffa and Reynolds, 1977; Landeros and Lyth, 1989; Langley, 1976, 1980; Larson, 1989; as well as Mattsson, 2007 further extended the approach in several aspects. [2] Such researchers considered additional factors such as transportation factors, buyer/seller relationships, quality considerations, short lead times and emergency conditions, among others. More still, others such as Namit and Chen, 1999; Tyworth and Ganeshan, 2000; Tyworth et al., 1996 evaluated the approach under particular demand and lead-times. [2]

A second inventory control approach widely used is the periodic review control system. The model as explained by Hadley and Whitin (1963), controls inventory through ordering on pre-set review intervals. Upon reaching a review interval, an order is placed such that inventory position is brought to some up-to-level. [2]

All these approaches and theories have one source of motivation; to understand and to resolve issues that arise as a result of inappropriate management of stock. The project progresses from the theoretical to a practical where some of the approaches discussed are integrated to address the inventory management challenge. The reorder level approach is one of such approaches used, to help address the question of either over or under stocking which is a major concern for inventory management.

2.2 Approach

The approaches discussed above appear to take into considerations manual attempts to manage inventory. The approach to addressing the challenge therefore was to design and to develop an automated system that took into considering reorder levels, customer management as well as employee efficiency and productivity. In the attempt to address stocking goods in excess, a feature of the system allows a user to specify a reorder level and alerts a user when that level or point is attained. It also addresses the issue of goods expiration on shelves which generally is a major source of worry for the SME's. When an expiry date is close, the system consistently alerts the user either to sell such products at discount prices or otherwise. This approach is also designed to generally reduce turnaround time. Turnaround time refers to the time it takes to move from customer to the next.

2.2.1 Development Tools

Tools used in the course of the development include

- Visual Basic.Net

Visual Basic as compared to other programming languages has rich and easy to use Graphical User Interface (GUI). This makes tasks easier and improves the look of an application. Because of this, users feel more relaxed and hence productivity increases. VB.NET also offers support for object oriented programming which simplifies coding.

- MySQL

MySQL was chosen for this project mainly because of its consistent reliability and ease of use [3]. MySQL is also an open source which saves the cost involved in buying. Additionally, MySQL also runs on more than 20 platforms including Linux, Windows, Mac OS, Solaris, HP-UX, and IBM AIX which gives the developer some flexibility [3].

- Windows 7 operating system

Windows 7 operating system was used because of the familiarity the developer has with the platform.

- Personal computer/ laptop

The basic hardware tool used for the project development was a personal computer and a laptop and requires one to run.

2.2.1 Methods

To tackle the project, the approach took into consideration the requirement elicitation process and requirement analysis and design. The requirement elicitation process dealt with the different ways to extract information. The first was to interview and interact with some of managers and attendants of the target; small scale enterprises and second, to observe them in their work, in delivering these services to their customers. This step was necessary because it is usually the case that what people say they do is sometimes different from what they actually do. In trying to come up with the requirements, survey research was employed.

Per my observation, small to medium businesses had to put in place certain mechanisms to take proper care of their businesses. There arose the need to test for their readiness and responsiveness to take charge of the flow of products (i.e. how products come in and how it goes out), monetary transactions and offer their staff the process to co-ordinate the range of activities vital for efficient inventory control.

To better appreciate their operations, considerable amount of time was spent with various small to medium business outlets administering the usage of simple bin/tally cards, cash/product transaction sheets and employee activity sheet to capture vital business operation data/information.

To get an accurate business operation data/information, the entire data capturing process was made very simple and straight forward by motivating the shop attendants with promises of less work burden should the results of the experiment come out good to entice their business owners to invest in automated inventory management systems.

Achieving a quick stock turnover in the shortest time possible, handling and accounting for cash proceeds and shop attendants adaptability and positive responsiveness to efficient inventory management were the set objective measures. After all, the quicker you turn over your stock the quicker cash is available to the business and hence the likelihood to make profit. The methods employed under survey research include observations and interviews.

2.2.2.1 Interviews

Interviews were sometimes in the form of questions and answers and at other times in the form of interactions. This was to prevent the interviewer being limited to the initial set of questions, but rather explore more by picking up on comments or statements made by the interviewee. Interviewees included some owners, managers as well as some customers.

2.2.2.2 Observations

Observations were sometimes done with little or no awareness of the staff. This was to enable observations to be conducted without the fear of staff or customers acting other than they would ordinarily have done with fore knowledge. Where necessary, senior management had knowledge but refrained from interruptions. In some other cases, there were some interactions with customers and staff.

Chapter 3

3.1 Requirement Analysis and design

This aspect of the paper deals with the development of a Software Requirements Specification document that specifies what the inventory management system should and should not do. For clarity the software requirements are split into five main sections namely [4];

1. System Objectives

   The system objectives lists all the goals and objectives of the system based on the viewpoint of the company. These are higher-level goals which are help in a top-down development of the IMS.

2. System Context

   This section shows the environment and confines of the IMS and the various units with which it interacts. It helps us see how the system fits into the existing scheme of things while defining what the system will do by itself and what it expects other units to do.

3. Functional Requirements

   This section specifically states the functions of the system – what it should and what it should not do. The section is split into subsections modelled after the real world activities such as selling to costumers etc. For example, a customer cannot buy on account if he does not already have an account with the business or a product cannot be bought if there is none available.

4. Non-functional Requirements

   Non functional requirements are quality requirements that specify the performance levels required of the system for the various kinds of activities.

Conditions on the access times, response times etc of the system are set in the non functional requirements. Usually where there are conflicting interests, it is necessary to make tradeoffs among various non-functional requirements.

5. Future Requirements

   These are the specifications which are not provided for now in the current version but which could be incorporated into future versions. The system could be designed in future to enhance the existing capabilities or add entirely new ones.

3.1.1 **System Objectives**

3.1.1.1 The Inventory Management System is a software application to assist a small scale business with the management and control of inventory as well as in transactions related to buying and selling of products which includes payments, receipts and account registration of customers.

3.1.1.2 from the viewpoint of the business –

3.1.1.2.1 Avoid tasks done manually by the shop attendants

3.1.1.2.2 Maintain customer information in case of shortage of goods, know consumer preferences etc.

3.1.1.2.3 Reduce the probability of shortage of goods.

3.1.1.2.4 Reduce the waiting time of a customer by knowing whether a product is available or not.

3.1.1.2.5 Show all possible substitutes for products.

3.1.1.2.6 Check the validity of input data and give a feedback to the user in case of errors or inconsistency.

3.1.2 **System Context**

The system will provide a Graphical User Interface (GUI) as part of the attendant/Administrator’s desktop environment.

DB-inventory is the database to hold inventory, customer, product, supplier, purchases and sales details. DB-inventory is updated by the manager or supervisor or one who has been assigned the privilege with the latest status of the inventory whenever there is any change in inventory. For example, if a new product is added to the products sold, the DB- inventory is updated. DB-inventory also contains the prices of products for various items. It also contains information regarding the number of products after each purchase. An employee table contains all the personal information of the registered members of the IMS. This can be updated by the attendant by logging in to the system. Information from this database is used during transactions like account payments etc. There is just one interface, for the administrator, manager or shop attendant.

3.1.3 Functional Requirements

3.1.3.1 Authentication

The system shall require authentication before any form of transaction can be carried out. It will ask the user for the following information– a user name and a password.

3.1.3.2 User registration

An administrator or a person with the required privileges shall be able to register an employee or users of the system. Employee information such as Employee Name, address, phone number, employee ID and date of employment shall be captured during the registration process.

3.1.3.3 Customer accounts

3.1.3.3.1 The customer would have a choice whether to be a registered member or not.

A registered member can either hold a debit or a credit account. A debit account allows a customer to pay some amount of money to the entity (subject to change) and is not required to make payments per every purchase.

3.1.3.3.2 To register a customer, customer details such as name, telephone number and address are captured. His initial deposit if available is also captured.

3.1.3.3.3 A credit limit shall be set for a registered customer. A credit account holder is allowed to purchase products on credit, based on an assigned credit limit. When the credit limit is attained, the system prompts the user and the customer may be prevented from purchasing on credit (based on the company's policy).

3.1.3.3.4 A submitted form shall be validated to ensure that all necessary details are captured correctly. For example, alphabets shall not be permitted in a field that should hold numeric values.

3.1.3.4 Checking availability of products

A user shall be able to check the availability of product as and when needed.

3.1.3.5 Total Amount payable

The system shall calculate the total amount payable based on the number of goods bought and the price of each product.

3.1.3.6 Payment

The system shall allow users to select payment options, either cash, debit or credit sales. For a registered customer, payment shall be charged to his account whiles for a non- registered customer, payment shall be required on demand before a customer is allowed to check out. Where the amount paid exceeds the amount payable, the balance is calculated and displayed to user.

3.1.3.7 Cancellation

The system shall give the user an option to cancel a process or an update at any point in time.

3.1.3.8 Customer profile update

The system shall enable the user to update a customer's profile at any time. Changes can be made in fields including but not limited to address, phone number and payment amount.

3.1.3.7 View Purchase, Sales and Product details

The system shall allow a user to view all information about a purchase, sale or product.

3.1.3.8 Report Generation

The system shall be able to generate the following reports

1. Balance Sheet
2. Profit and loss statement
3. Product report
4. Sales report
5. Purchases report

3.1.3.9 Validations

All forms shall be validated to ensure that all necessary fields filled in. It shall also check to ensure that the fields were filled in the correct way. If the submitted form meets all the correct requirements, the database is updated with the details.

3.1.4 Non-functional Requirements

3.1.4.1 Performance

3.1.4.1.1 Response time of the Inventory Management System shall be less than 2 second most of the time. Response time refers to the waiting time while the system accesses, queries and retrieves the information from the database.

3.1.4.1.2 IMS shall be able to handle at least 1000 transactions/inquiries per second.

3.1.4.1.3 IMS shall show no visible deterioration in response time as the number of purchases increase

3.1.4.2 Reliability

3.1.4.2.1 IMS shall be available 24 hours a day, 7 days a week

3.1.4.2.2 IMS shall always provide real time information about product

3.1.4.2.3 IMS shall be robust enough to have a high degree of fault tolerance. For example, if the user enters a negative quantity of products or a value too large, the system should not crash and shall identify the invalid input and produce a suitable error message.

3.1.4.3 Usability

3.1.4.3.1 IMS shall provide an easy-to-use graphical interface similar to other inventory management system so that users would not have to learn a new style of interaction.

3.1.4.3.2 Any notification or error messages generated by IMS shall be clear, polite and free of jargon.

3.1.4.4 Integrity

3.1.4.4.1 Only system administrator has the right to change system parameters, such as pricing policy etc. The system shall be secure and must use encryption to protect the databases.

3.1.4.4.2 Users need to be authenticated before having access to any personal data.

3.1.5 Future Requirements

3.1.5.1 Users shall be able to access product information via a website

3.1.5.2 Customers shall be able to view available products online

3.1.5.3 Customers shall be able to purchase available products online

Chapter 4

4.1 System design and implementation

Through the requirements gathered, the key pieces needed to be implemented were identified. These were divided into modules.

Modules

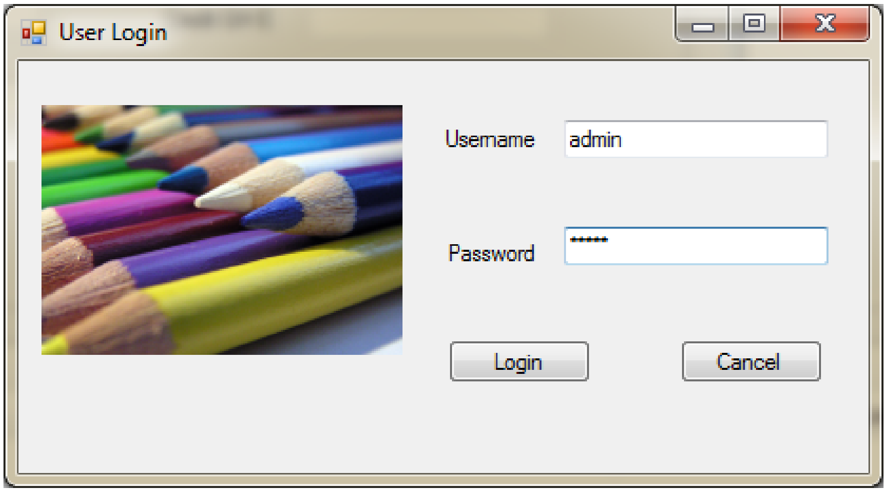
Modules that had to be executed for the user interface design were as follows:

- Login page
- A main page
- Add User
- Edit user
- Remove User
- Administrator password change
- Employee password change
- Manage Privileges
- Add customer Account
- Edit customer account
- Remove customer account
- Add supplier account
- Edit supplier account
- Remove supplier account
- Add product
- Edit product details

- Remove products
- Add purchase
- Edit purchase details
- Remove purchase
- Add sale
- Edit sale details
- Remove sale
- Sales report
- Purchases report
- Product report
- Balance Sheet
- Profit and loss statement

4.1.1 Login

The login form authenticates a user. At the login page, a user logs in by entering his details; a username and a password. When he is authenticated, the main page opens up and based on the privileges he has, he is able to access the system accordingly. The login system is an essential part of the system since it keeps unauthorised users out. It further helps to resolve the issue of who did what in the daily operations of the business.



Username admin

Password ******

Figure 4.1: User Login form

The login authentication process:

Figure 4.1 shows the user login form. When a user logs in, the system checks the database that houses user information whether that user exists or not. User is notified through a pop up if the username or password does not exist. For a first time user with valid username and password, the user is asked to change his password. A notification that the process was successful or otherwise is displayed and the main page is presented for the use. The user is then able to manipulate the system according to the privileges he has.

4.1.2 Main Page

The main page has four different modules. These are the file, Maintenance, invoicing and reports modules. A user may or not be able to access these modules based on his access rights or privileges set by the administrator.

4.1.2.1 File

File is the first module in the main page. File has a sub module; “users” where an administrator can add, edit or remove users, change user passwords and assign privileges.

4.1.2.1.1 Users

“Users” is a direct sub module for file. When user is clicked, the options to add edit or remove users, change password as well as assign privileges are opened.

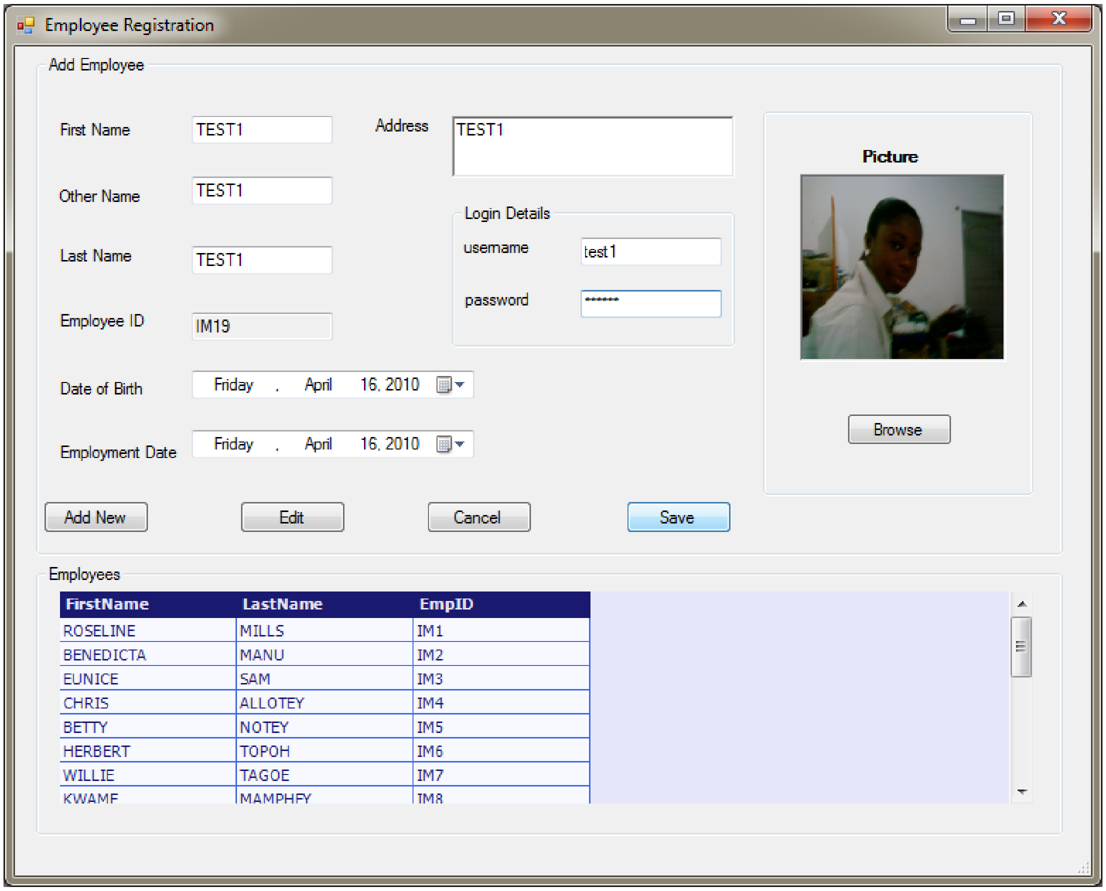
Add user

Add user is the sub module under the “user” module.

Add user presents an employee registration form (figure 4.2) for user details to be recorded. The employee registration form has text fields that take as inputs user details. To add user, details such as user’s first name, last name, other name, employee ID, address, date of birth and date of employment details are recorded. The save button updates the database with the new information. Users are uniquely identified by the user ID which is automatically generated for each user during the add user process.

On the employee registration form, there are features that allow a user to add new user and to edit user details on the form. These features reduce the number of user clicks employed to perform an “edit” or “add new” task. Where a user has to register a second user immediately after registering a first, he only has to click on “add new” button on the same form. To edit user details also whiles on the user registration form, a user can instantly edit details on the same form without having to go back to the “user” module, before he is able to, thereby making the system more efficient.

An administrator assigns a username and a password to a user. Usernames as well as passwords are unique. The “cancel” button is another feature on the employee registration form. Cancel takes a user back to the main page where he can select other modules for different functions. The “Save” button updates the employee database with employee details as and when a new user is added.



First Name: TEST1

Other Name: TEST1

Last Name: TEST1

Employee ID: IM19

Date of Birth: Friday , April 16, 2010

Employment Date: Friday , April 16, 2010

Address: TEST1

username: test1

password: ******

FirstName	LastName	EmpID
ROSELINE	MILLS	IM1
BENEDICTA	MANU	IM2
EUNICE	SAM	IM3
CHRIS	ALLOTEY	IM4
BETTY	NOTEY	IM5
HERBERT	TOPOH	IM6
WILLIE	TAGOE	IM7
KWAME	MAMPHEY	IM8

Figure 4.1: Employee Registration form

Edit User

The next sub module under the “users” module is the “edit user”.

The edit user module presents an “edit user” form to the user. However, with this module, instead of a text field that holds the user ID generated as with the “add user”, a drop down is provided to the user. The user is prompted to select a user ID for the user whose details are to be edited. When the id is selected, the other fields are automatically populated with his details from the database. The details can then be edited and saved. The employee database is updated following each edit. the same form as the user registration form. In the instead of a text field that would hold the user ID generated, a drop down button is made visible to the user. To activate this drop down however, a user first has to click on cancel to reset the mode, click on edit. The drop down button is enabled and a user is prompted to select a user ID whose details are to be edited. When the employee is selected, the other fields are automatically populated with the employee details from the database. The details can then be edited and the save button clicked. The employee table in the database is updated following each edit. The “Save” button operates in a dual mode namely; add new and edit. The “add new” mode updates the database with the new information when a new user is added and the edit updates the database with the particular detail that was changed.

Remove User

“Remove user” deletes a user record from the database. Before this is done however, a copy of the database is saved in another section of the database. When remove employee is clicked, a dialog box pops up to confirm the removal. A remove user page is presented and a user selects the employee ID (ID that identifies the employee to be removed). When the remove button is clicked, the employee is details are deleted from the database.

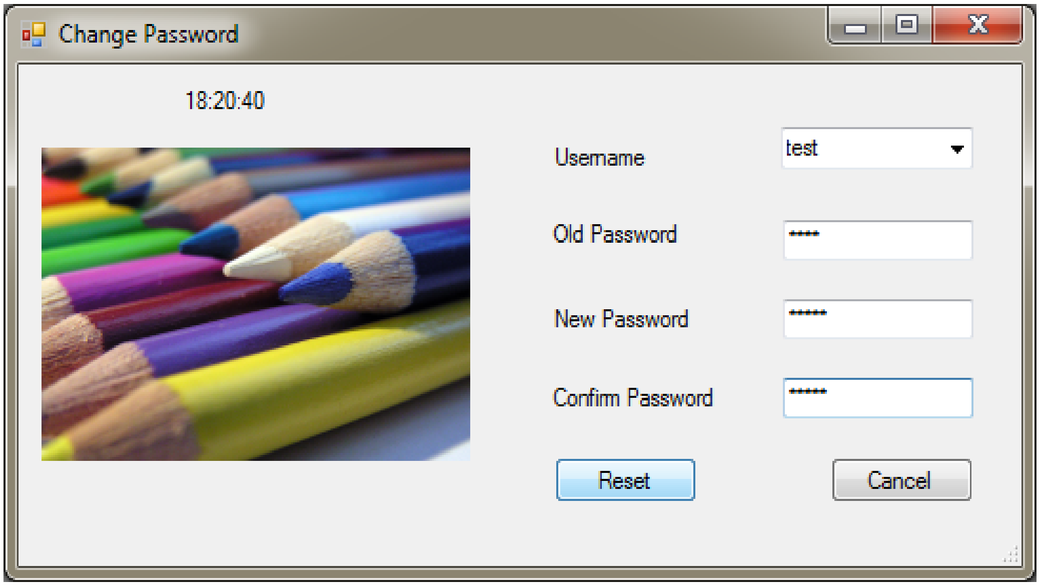
4.1.2.1.2 Assign Privileges

“Assign privileges” allows an administrator to set or edit user privileges. To set privileges, employee first name, last name and employee ID are recorded; the various modules that should be accessible to users are checked and saved. Where a privilege has already been specified for a user, the user is prompted and reminded. The user is then given the steps to edit the privileges or to create new privileges for a different user. The main purpose of this functionality is to address the accountability issues and to protect the integrity of the system. The database is updated with the user privileges when the save button is clicked.

The cancel button allows a user to cancel any he privileges attempted to set. The text fields are cleared and he returns to the main page.

4.1.2.1.3 Administrator password change

Under this module, an administrator is able to reset passwords for users who may have forgotten their passwords. The “administrator password change” form (figure 4.3) is presented when the “change password” module is clicked. A drop down button allows an administrator to select a user ID for the user whose password is to be changed. When the ID is selected, old password text field is populated with the old password for that user from the database. The user/ administrator provide a new password for the user. The administrator confirms the password in the “confirm password” text field and clicks on reset when he is done. When the user whose ID was changed logs in, he is forced to change his password with the “employee password change” form. The password that populates the old password field is encrypted; therefore the administrator is not able to see the user’s password. This security feature was implemented as a measure against password theft.



18:20:40

Username: test

Old Password: ****

New Password: *****

Confirm Password: *****

Figure 4.2: Administrator password change

4.1.2.2 Maintenance

Maintenance is the second module on the main page. The maintenance module has a collection of sub modules which are the customer, supplier and product. These items are filed under sub modules because such items need to be constantly maintained and updated.

4.1.2.2.1 Customer

When the “customer” sub module is selected, a user is presented with the option to add, edit or remove customer details.

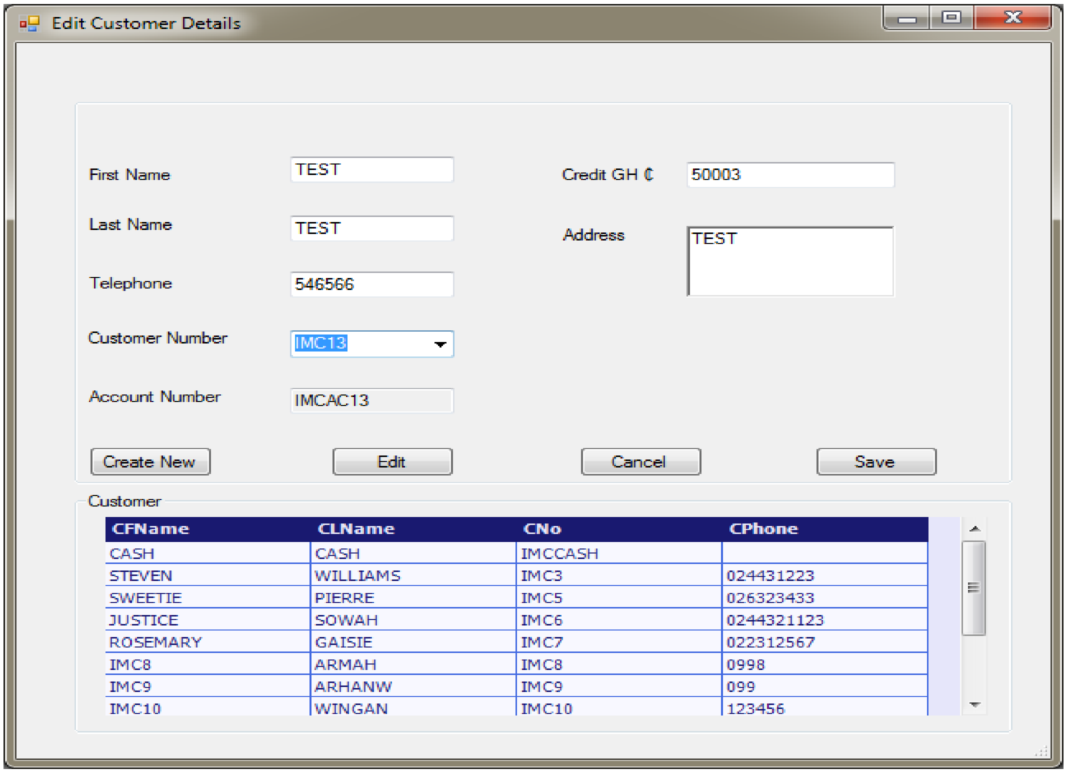
Add Customers

The add customer module presents a customer registration form to the user. Customer details such as first name, last name, telephone, and credit amount are recorded. A customer ID as well as an account number is automatically generated for each customer during the customer registration process. Customers are uniquely identified by their ID’s. The customer’s initial credit amount is entered during registration and subsequent deposits are recorded on a “customer deposits” page. The deposits update the customer’s account in the database.

On the “customer registration” are the “Add new” and “edit” functions. These functions reduce the back and forth a user might have to do to edit customer details or create new customer account. The save button updates the customer database with new customer when he is added whiles the cancel button allows a user to cancel any update he attempted to make. The text fields are cleared when “cancel” is invoked and the administrator can return to the main page.

Edit Customer

As with the “edit user” sub module, when the “edit customer” is selected, the edit customer details form (figure 4.4) is presented to the user. The form comes with a drop down button from which he selects the customer ID whose details are to be edited. The remaining customer detail fields are automatically populated with his details from the database. The user is then able to edit the required details. The save button then updates the database with the new information.



Edit Customer Details

First Name	TEST	Credit GH ₵	50003
Last Name	TEST	Address	TEST
Telephone	546566		
Customer Number	IMC13		
Account Number	IMCAC13		

Customer

CFName	CLName	CNo	CPhone
CASH	CASH	IMCCASH	
STEVEN	WILLIAMS	IMC3	024431223
SWEETIE	PIERRE	IMC5	026323433
JUSTICE	SOWAH	IMC6	0244321123
ROSEMARY	GAISIE	IMC7	022312567
IMC8	ARMAH	IMC8	0998
IMC9	ARHANW	IMC9	099
IMC10	WINGAN	IMC10	123456

Figure 3.4: Edit Customer Details form

Remove Customer

“Remove customer” deletes a customer record from the database. When remove customer is clicked, a remove customer page is presented and the user is able to select the customer id for which deletion is required. The administrator/user is asked to confirm the removal when he selects “remove” through a dialog box. Before any record is deleted from the database, a copy of the record is created and saved in another section of the database for possible future reference. The customer details are deleted from the database if confirmed, but returns to the same page if not. When cancel is selected, the page clears and the user is able to go back to other pages.

4.1.2.2.2 Supplier

Supplier is another sub module under the maintenance module. Sub modules under supplier include add, edit and remove supplier details.

Add Supplier

When “Add supplier” is selected, a supplier registration form is presented to the user. Supplier details such as Supplier name, supplier ID, telephone, address and account numbers are recorded. The supplier registration form has text fields that take as inputs supplier details. The save button is clicked after a supplier is added to update information in database. Suppliers are uniquely identified by their ID’s which are automatically generated for each user during the registration process.

Modules such as the “Add New” and “edit” on the supplier form are designed to enhance speed and efficiency. These features effectively the number of user clicks required to create a new account or to edit an account. For instance, whiles on the supplier registration form, a user can edit details without having to go back to the user model, before he is able to. The Cancel button allows a user to cancel any addition he attempted to make to the supplier database. The text fields are cleared when cancel is selected and he returns to the main page.

Edit Supplier

Edit supplier is a page where supplier details are edited. The "edit supplier" form is loaded when edit is selected and the user is given the chance to edit. As in the edit feature under "add supplier" module, the details are filled when a supplier ID is selected from the drop down button and a user can edit the details. The supplier table in the database is updated when edit is clicked again.

Remove Supplier

The "Remove supplier" module deletes a supplier record from the database. When remove supplier is selected, a "remove supplier" page is presented and a user selects the supplier ID for which supplier detail is to be deleted. When the remove button is selected, the system checks if the user initiated the remove process in the first place by seeking confirmation from user. Use is warned of the implication of clicking the remove button and proceeds to ask the user for confirmation. When yes is clicked, the supplier details are deleted from the database supplier and text fields are cleared when the user does not.

4.1.2.2.2 Products

The third sub module under the maintenance module is the “products”. “Products” has sub modules to register new products, edit product details and to remove products.

Add Products

When Add product is selected, a product registration form is presented to the user. Product details such as Product name, Product ID, barcode, category, unit and price are recorded and saved in database. The product registration form has text fields that take as inputs these details. The user fills in the details and saves the record. Products are uniquely identified by their ID’s which are automatically generated by the system for each product during the product registration process

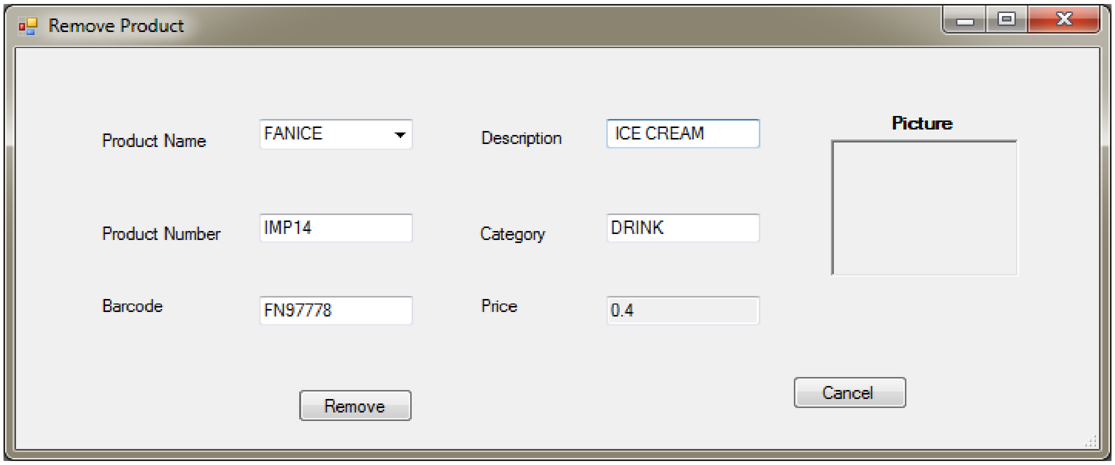
There are also the “add new” and “edit” sub modules that allow a user to add new product and edit details on the add product form. An advantage of such features is that, the number of user clicks is reduced. To edit product details also whiles on the product registration form makes the system simple and easy to use. , a user can immediately edit details on the same form without having to go back to the user module, click on edit before he is able to. Cancel allows a user to cancel any addition he attempted to make to the product table. The text fields are cleared and he returns to the main page.

Edit Product

The “edit product” sub module allows product details to be updated. A user is presented with an “edit product page when edit is selected”. To edit a product, the product number is selected from a drop down button. The remaining product details text fields are automatically populated upon the selection of the ID which allows a user to edit product details. The save button updates the database with the new information when the user finishes with the update.

Remove Product

Remove product deletes the record of a product from the database. When “remove product” is selected, a remove product page (figure 4.5) is presented and a user is able to select the product ID from a drop down button provided. When the remove button is clicked, a dialog box asks the user for confirmation. The product is deleted from the database when a user confirms the removal and the details that were populated are cleared when he clicks on cancel.



Product Name: FANICE

Description: ICE CREAM

Product Number: IMP14

Category: DRINK

Barcode: FN97778

Price: 0.4

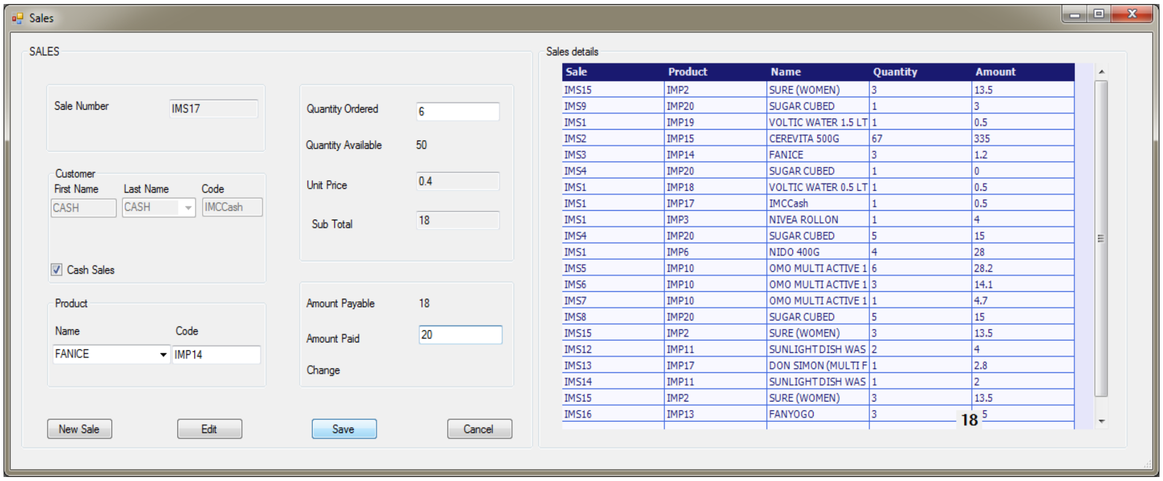
Figure 4.4: Remove Product details

4.1.2.3 Invoicing

Invoicing is the third module on the main page. Invoicing takes care of purchases and sales invoicing. Sales Invoicing is concerned with the recording of sale transaction with customers whiles the purchases is concerned with purchases made from suppliers.

4.1.2.3.1 Sales

Under the sales module, a user is able to add a sale, edit sale details and delete sales from the database



Sale Number: IMS17

Customer

First Name	Last Name	Code
CASH	CASH	IMCCash

☑ Cash Sales

Product

Name	Code
FANICE	IMP14

Quantity Ordered: 6

Quantity Available: 50

Unit Price: 0.4

Sub Total: 18

Amount Payable: 18

Amount Paid: 20

Change

Sales details

Sale	Product	Name	Quantity	Amount
IMS15	IMP2	SURE (WOMEN)	3	13.5
IMS9	IMP20	SUGAR CUBED	1	3
IMS1	IMP19	VOLTIC WATER 1.5 LT	1	0.5
IMS2	IMP15	CEREVITA 500G	67	335
IMS3	IMP14	FANICE	3	1.2
IMS4	IMP20	SUGAR CUBED	1	0
IMS1	IMP18	VOLTIC WATER 0.5 LT	1	0.5
IMS1	IMP17	IMCCash	1	0.5
IMS1	IMP3	NIVEA ROLLON	1	4
IMS4	IMP20	SUGAR CUBED	5	15
IMS1	IMP6	NIDO 400G	4	28
IMS5	IMP10	OMO MULTI ACTIVE 1	6	28.2
IMS6	IMP10	OMO MULTI ACTIVE 1	3	14.1
IMS7	IMP10	OMO MULTI ACTIVE 1	1	4.7
IMS8	IMP20	SUGAR CUBED	5	15
IMS15	IMP2	SURE (WOMEN)	3	13.5
IMS12	IMP11	SUNLIGHT DISH WAS	2	4
IMS13	IMP17	DON SIMON (MULTI F	1	2.8
IMS14	IMP11	SUNLIGHT DISH WAS	1	2
IMS15	IMP2	SURE (WOMEN)	3	13.5
IMS16	IMP13	FANYOGO	3	18 5

Figure 4.5: Sales form

Add Sale

Add sale is a sub module of the sales module. An authorised user (a user with the necessary privilege) is able to undergo a sales transaction. Sales transaction demonstrated in figure 4.6, records the sales number, product number, customer number (where the customer is a registered one), quantity ordered and the unit price. The amount is calculated and displayed as quantity ordered * unit price. The sale number is auto generated by the system. The price of a product is fetched when a product is selected. The customer balance (where he is registered) is also fetched and displayed on the form when a customer is selected. Where the customer has exceeded his credit limit, the user is prompted and the customer may not be served based on the policy of the company. There are also multiple dropdowns that allow the sale of several different products at the same time. When the save button is clicked, all transaction information are entered into the database and the database is updated with the new information.

Sub modules that allow a user to add new sale and edit on the sales form. Such features allow for little number of clicks. To edit sales details, whiles on the sales form makes the system simple and easy to use. A user can immediately edit sales details on the same form without having to go back to the sales model, click on edit before he is able to edit. Cancel allows a user to cancel any additional changes he attempted to make to the product table. The text fields are cleared and he returns to the main page.

Edit Sales

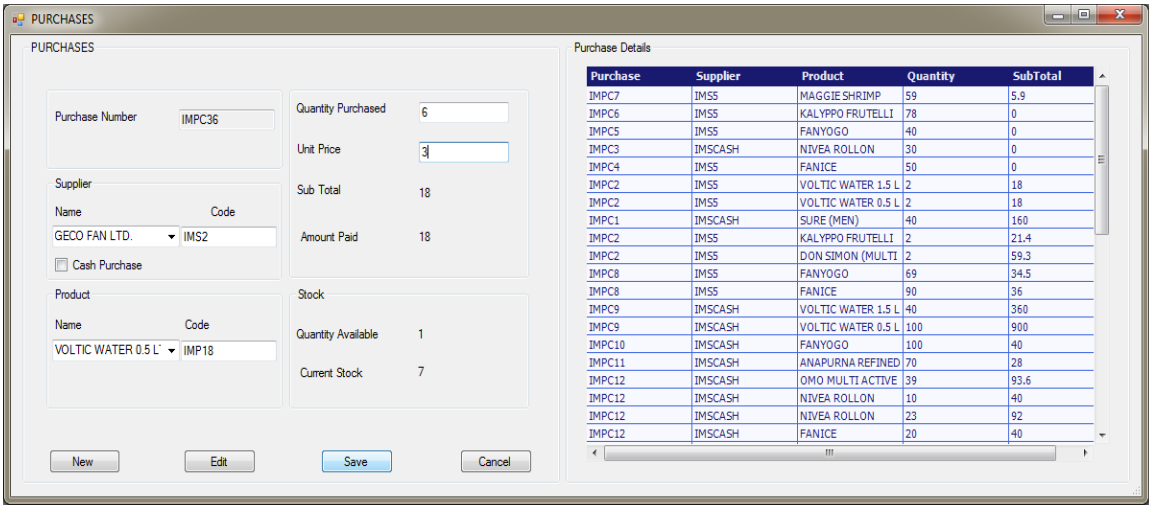
The second sub model under the Sales model is Edit Sales. Edit allows sales details to be updated. When edit is clicked, sales number is selected from a drop down button. The sales details text fields are then automatically populated. A user is therefore able to edit the desired details. When he is done, he clicks edit again which updates the database with the new information.

Remove Sales

Remove sales deletes a sales record from the database. The “remove sales” page allows a user to select sales ID from a drop down button. The rest of the text fields on the "remove sales" form are populated. When the remove button is clicked, a dialog box asks the user whether he really wants to remove. The sales record is deleted from the database when a user confirms the removal and the details that were populated are cleared when he clicks on cancel.

4.1.2.3.2 Purchases

The final module under the invoicing is the purchases sub module. Add Purchase, edit and remove purchase are sub modules listed under purchases.



Purchase Number	IMPC36
Quantity Purchased	6
Unit Price	3
Sub Total	18
Amount Paid	18

Supplier

Name	Code
GECO FAN LTD.	IMS2

□ Cash Purchase

Product

Name	Code
VOLTIC WATER 0.5 L	IMP18

Stock

Quantity Available	1
Current Stock	7

Purchase	Supplier	Product	Quantity	SubTotal
IMPC7	IMS5	MAGGIE SHRIMP	59	5.9
IMPC6	IMS5	KALYPPO FRUTELLI	78	0
IMPC5	IMS5	FANYOGO	40	0
IMPC3	IMSCASH	NIVEA ROLLON	30	0
IMPC4	IMS5	FANICE	50	0
IMPC2	IMS5	VOLTIC WATER 1.5 L	2	18
IMPC2	IMS5	VOLTIC WATER 0.5 L	2	18
IMPC1	IMSCASH	SURE (MEN)	40	160
IMPC2	IMS5	KALYPPO FRUTELLI	2	21.4
IMPC2	IMS5	DON SIMON (MULTI	2	59.3
IMPC8	IMS5	FANYOGO	69	34.5
IMPC8	IMS5	FANICE	90	36
IMPC9	IMSCASH	VOLTIC WATER 1.5 L	40	360
IMPC9	IMSCASH	VOLTIC WATER 0.5 L	100	900
IMPC10	IMSCASH	FANYOGO	100	40
IMPC11	IMSCASH	ANAPURNA REFINED	70	28
IMPC12	IMSCASH	OMO MULTI ACTIVE	39	93.6
IMPC12	IMSCASH	NIVEA ROLLON	10	40
IMPC12	IMSCASH	NIVEA ROLLON	23	92
IMPC12	IMSCASH	FANICE	20	40

Figure 4.7: Purchases form

Add Purchase

Add purchase is a sub module of the purchases module. An authorised user (a user with the necessary privilege) is able to undergo a purchase transaction. Purchase transaction records the purchase number, product number, supplier number, quantity ordered, price per unit and total amount paid as shown in figure 4.7. The purchase number is auto generated by the system. There are multiple dropdowns that allow different products to be captured for the same purchase. When the save button is clicked, all purchase transaction information are entered into the database and the database is updated with the new information.

The edit sub model on the add purchase form allows purchase details to be updated. When edit is clicked, purchase number is selected from a drop down button and the purchase details are automatically filled in the text fields. A user can therefore edit the details. When he is done, he clicks edit again which updates the database with the new information. The text fields are cleared and a user returns to the main page when the cancel button is clicked.

Edit Purchase

Edit purchase is a page where purchase details can be edited. Add purchase page comes up when this is clicked and the user clicks on edit. Like the edit button function on the purchases form, the edit purchase form allows a user to select from the drop down button for purchase Number. The details are filled when a purchase number is entered and a user can edit the details. The purchase table in the database is updated when edit is clicked again.

Remove Purchase

Remove purchase deletes a purchase record from the database. When remove purchase is clicked, a “remove purchase” page is presented and a user selects the purchase number (number for the purchase record going to be removed). When the remove button is clicked, a dialog box asks the user whether he really wants to remove. The purchase record is deleted from the database when a user confirms the removal and the details that were populated are cleared when he clicks on cancel.

4.1.3 Employee Password Change

The “employee password change” is a feature that is hidden and only presented when a user has to change his password. On the “employee password change” form, a user is required to enter his username and old password. He also has to confirm the new password in the confirm password text field. The system validates the user by checking if the user name and old password exist in the database. When the confirm password is typed in, it is matched against the new password typed. If there are changes, the user is notified and he will have to redo it. The password is saved when a user click on the save button. The employee password change is presented in situations where users forgot their passwords and had to change passwords. It is also presented to first time users who have never logged onto the system.

4.2 Database Implementation

The database built to manage the program was developed using MySQL. The MySQL database employed was part of a free and open source cross-platform web server package, consisting mainly of the Apache HTTP Server, MySQL database and interpreters for scripts written in the PHP and Perl programming languages (XAMPP) [5]. XAMPP creates a local server on a computer through which the database is accessed. The inventory database comprises of 23 tables which hold all information concerning transactions and processes. The IMS connects to the database to run queries and transactions.

The database system is implemented using data definition languages (DDL) and the storage definition language of the relational database management System (DBMS) and compiled [6]. These are used to create the database schemas and database files. The database is then populated with data through the forms created for the application. Database transactions are implemented by referring to the conceptual specifications of transactions and then codes are written and tested.

Out of the 23 tables, 10 tables were solely created to make copies of transactions that had to be updated. SQL queries were used to copy information from the original tables to these supplementary tables. An audit trail was also created in the database to monitor actions or activities that went on in the system. Whenever a form was called, an entry was logged into the database. Details such as the user id, date, time and activity was recorded; activity being the action that was performed. E.g. “user account was viewed”

4.3 Testing and evaluation

To satisfy the requirements gathered during the requirement elicitation stage, the principle of incrementality in Software Engineering was used during the development stage. This principle suggests that, modules are developed and tested before the next module is added. Therefore in the testing stage, the modules were rigorously tested and evaluated before other modules were added. The application was tested by writing queries to the database and the results documented. Where the application behaved in an unsuitable or inappropriate manner, steps were taken to correct it. Validations were also done to ensure the integrity of the database. The modules were evaluated, based on how they effective they met the functional requirements collected during the requirements stage. The application installer was deployed unto a client machine with component of the database for installation.

Some features that were tested include but not limited to

- Login authentication
- Numeric text field validation
- Password change validation
- Payment
- Reports

Login authentication

A user has to be registered before he is able to access the system. The login feature authenticates any user by comparing the username and password provided with the record in the database. To test the functionality of this feature, an invalid username was first provided in the field. The user got a feedback that notified him of the invalidity of the username. A valid username but an invalid password was tried next. The tester got the same message as before. After three tries, the user's account was disabled and alerted to contact the administrator. This security feature was implemented to prevent intrusion into the system. A user whose account has been disabled has to contact the administrator personally to have his account reset with a new password which he would be able to login with. However, he is forced to change his password after a reset. If the system permits an unauthorised user to access information he is not entitled to, it will mean a failure of the system terms of security.

Numeric text field validations

In modules that had fields which take numeric data as input, a numeric field test was conducted. Whenever an alphabet was typed in, a message which indicated that numbers or spaces were not allowed popped up. The text fields therefore accepted only numbers which in one way, ensured the integrity of the data input.

Password Change validations

Where a user’s account had been disabled or a user forgets password, the user is required to change passwords. After the old password (the reset password for a disabled account) and new password is entered, the tester typed in a confirmation password that did not match the one typed in the new password field. A message that indicated that the new and confirmed passwords did not match popped up. Again, where the tester attempted to copy the password typed in the ‘new password’ field into the ‘confirm password’ field, he was prevented. This was done to ensure that the user is fully aware of the password he chooses to use.

Input field validations

All input fields were tested for uniformity of data that was input into the database. Letters in small caps which were entered into the fields were converted into capital letters. Usernames however were converted to small letters for consistency. The test run indicated that this feature worked as expected.

Payment

In a sales transaction, a functional requirement was for the system to allow users to select payment options, either for cash, debit or credit sales. To test for the functionality of this feature, the tester tried to make a sale with the “sales” form. First a credit account was selected from the drop down list which showed the customer’s credit balance. The total amount payable was calculated after the amount was entered. The price appeared automatically when product was selected. Where a cash sale was selected, the system forced the user to enter an amount before the sale was processed. The balance is calculated as well where the amount paid exceeded amount payable.

Reports

The system was also tested for its ability to generate report as specified in the requirement document. The main report tested for, was the purchases report. The period for which the report was required was selected from a drop down button and all purchase details were viewed. The system was successful in that regard since it satisfied the requirement.

In checking for how users' task could be simplified using the system, the number of user clicks were measured at various stages of the development and care was taken to ensure that clicks were reduced to the barest minimum in order to facilitate the performance of a task.

CHAPTER 5

5.1 Limitations of the system

In as much as the system helps in the management processes and sales transactions, some limitations of it include the inability of the system to capture shop attendants idle time. The system again cannot show customer satisfaction levels which is an essential part of the success of the business. Again, the first in, first out policy for products cannot be fully enforced by the system. Applying the average costing method for products is almost impossible since costs of operations cannot be fully captured. The inability of the system to run in a web browser- like environment makes deployment on a network quite a difficult task since all installations would have to be carried out on every single PC. Thus the future requirements; to enable the system run via website.

5.2 Benefits of the system and conclusion

Its limitations notwithstanding, the system when implemented, would be able to effectively eliminate overstocks or stock out, ease off the time and effort required for physical inventory, greatly reduce the amount of paper work involved in the manual process as well as increase employee productivity.

The inventory management system is a desktop application that acts as a solution to most problems small scale businesses encounter in the course of managing inventory. Some of these problems include pilfering by employees, lack of any monitoring system or inability of management to keep track of stock. Most business are able to run without a system like this but it is becoming increasingly clear to management that, without any proper inventory management for their assets, the business would only suffer in the long run. As the modules offered by this system do not only support inventory management, but also support point of sale transactions, businesses stand to gain a great deal if this system is implemented.

Bibliography

[1] Jr. J. Tol Brooame. (1999, June) findarticles Web Site. [Online]. http://findarticles.com/p/articles/mi_m1154/is_6_87/ai_54695720/

[2] Brent D. Williams and Travis Tokar, "A review of inventory management research in logistics journals Themes and future directions," *The International Journal of Logistics*, vol. 19, pp. 212-232, 2008.

[3] MySQL. (2010) MySQL website. [Online]. http://www.mysql.com/why-mysql/

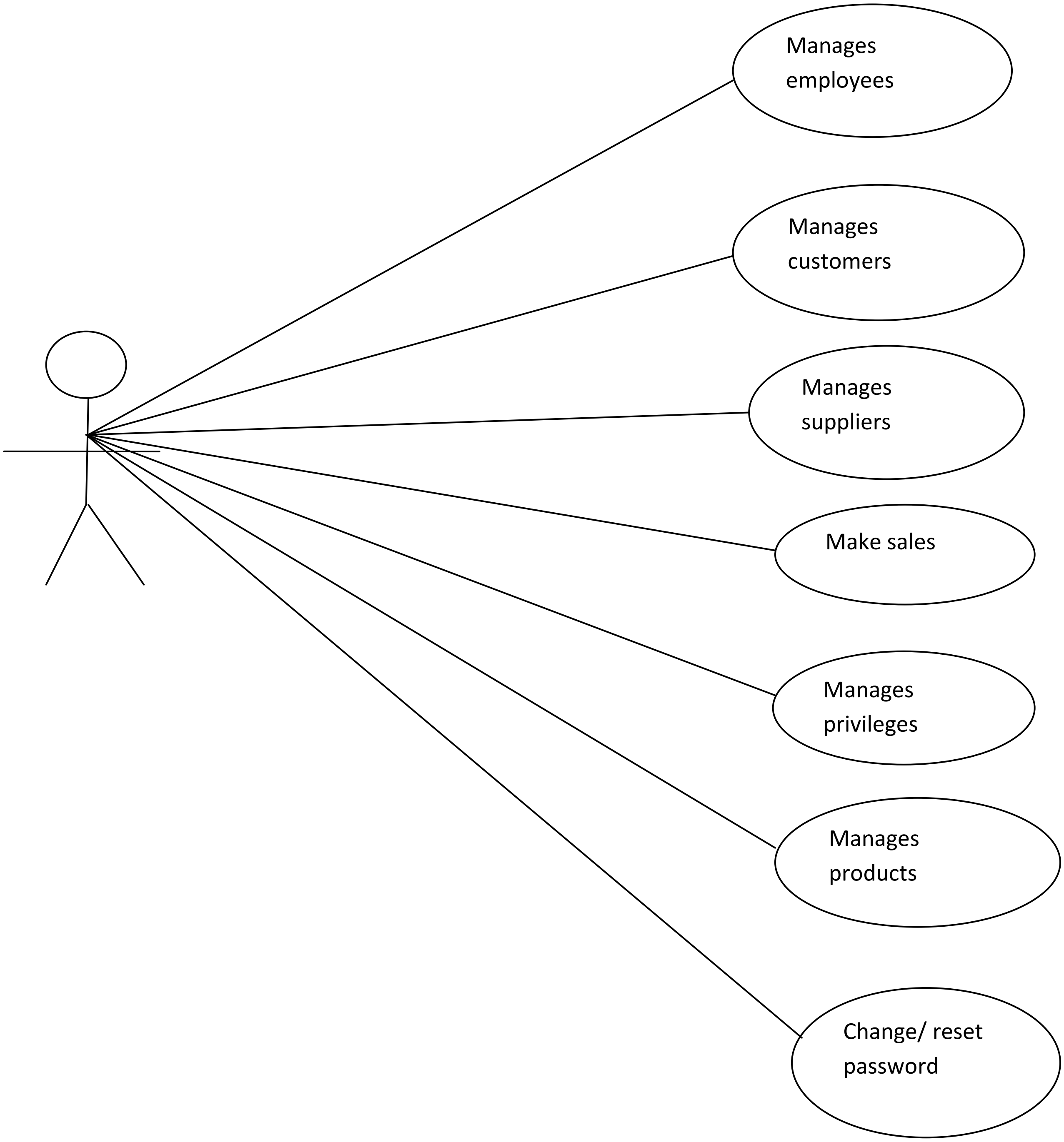
[4] Sonthi et al, Airline Reservation System; Software Requirements Specifications, (2000), Georgia Institute of Technology.

[5] Encyclo Online Encyclopedia. [Online]. http://www.encyclo.co.uk/define/XAMPP

[6] Elmasri & Navathe, *Fundamentals of Database Systems*, 5th ed.: Pearson, Addison Wesley, 2007.

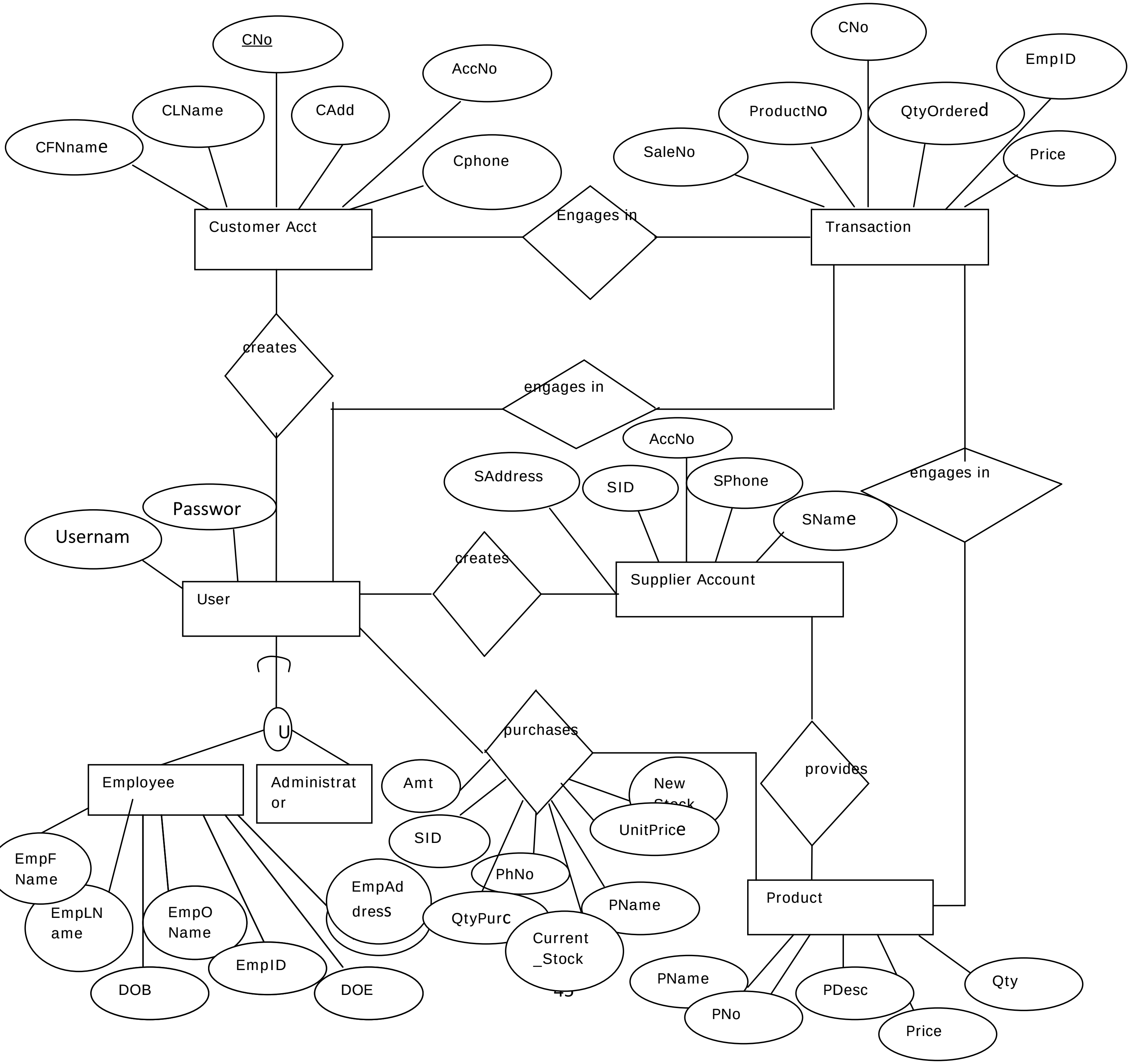
Appendix

Appendix 1: Use Case Diagram of Administrator

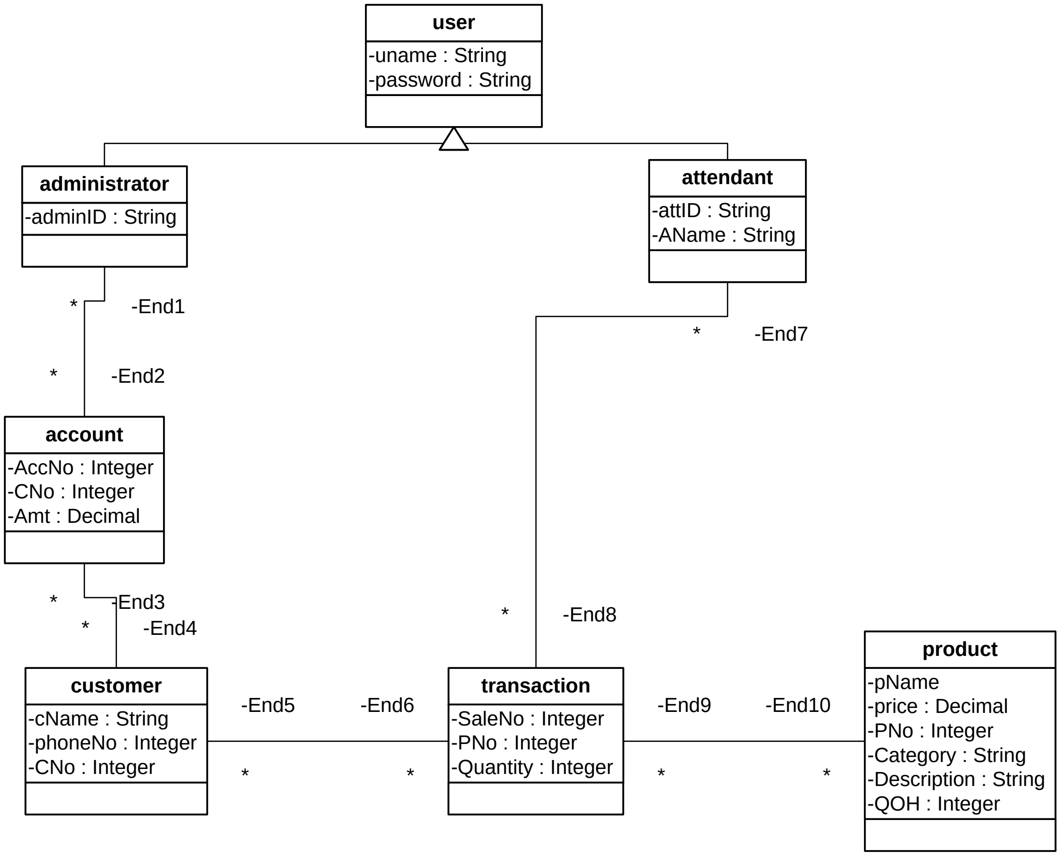


Appendix 2: Conceptual Database Design

The Enhanced Entity Relationship diagram is a high level conceptualization of what happens in the course of managing the business. This represents entities and their relations.



Appendix 3: UML diagram of the Inventory database



Appendix 4: Logical Design

Relational Schema:

A Schema (or intention) is the shape or framework of a relation. The schema defines the number of attributes, their names and domains. The schema therefore shows the structure of the database.

CustomerAccount (CNo, CAccNo, CreatedBy, DateCreated, time)
Primary Key CAccNo

Customer (CFName, CLName, CNo, CAccNo, CAdd, CPhone, CreatedBy, dateCreatedBy, time)
Primary Key CNo

Transaction (SaleNo, PNo, PName, QtyOrdered, CNo, Price, Amount, CreatedBy, dateCreatedBy, time)
Primary Key SaleNo
Foreign Key CNo, PNo, EmpID, Price

Password (EmpID, username, password, CreatedBy, dateCreatedBy, time)
Primary Key username, password

Employee (EmpFName, EmpOName, EmpLName, EmpID, EmpAdd, DOB, DOE, CreatedBy, dateCreatedBy, time)
Primary key EmpID
Alternate key EmpLName, DOB

Product (PName, PNo, barcode, PCat, Price, Unit, CreatedBy, dateCreatedBy, time)
Primary Key: PNo
Alternate key: PName

Purchase (SID, PurchaseNo, PNo, PName, PDescr, QtyPurchased, UnitPrice, SubTotal, CreatedBy, dateCreatedBy, time)
Primary key: PurchaseNo
Alternate key: SID, PNo
Foreign Key: PNo, PName, PDescr, UnitPrice

Supplier (SName, SID, SPhone, SAcc, SAddress, CreatedBy, dateCreatedBy, time)
Primary Key: SID
Alternate key: Sname

Assigned_Privileges (EmpID, manageUsers, manageCustomers, manageProducts, MakeSales, EditSales, RemoveSales, ManageSuppliers, ManagePurchases, GenSalesReport, GenPurchaseReport, createdBy, dateCreated, Time)
Primary key: EmpID
Foreign Key: EmpID

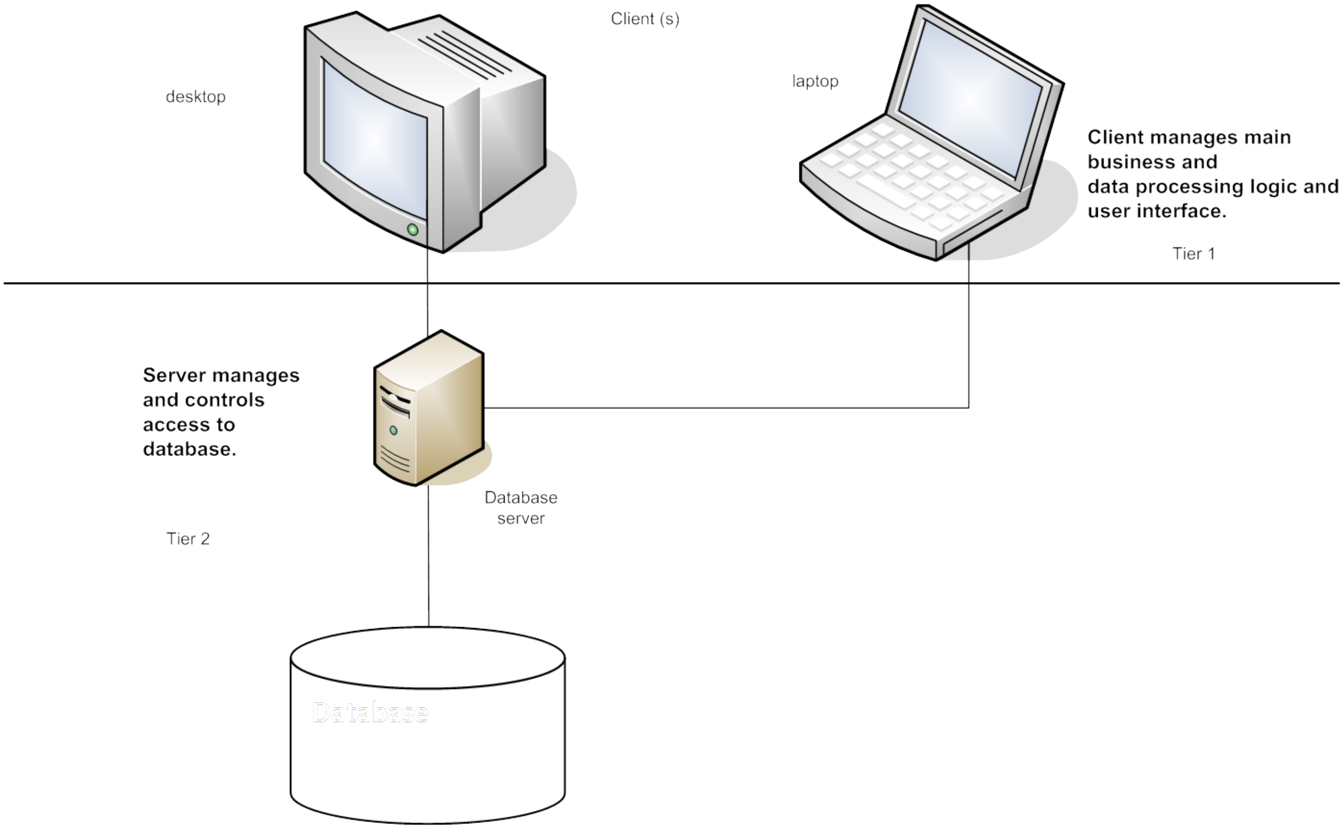
Stock (PNo, currentStock, createdBy, dateCreated, time)
Primary key: PNo, currentStock

Foreign Key: PNo

Login_History (status, username, Date, Time)

Appendix 5: System architecture design

The program is installed on a machine (client) which queries the database for transactions.


Client (s)
desktop
laptop
Client manages main business and data processing logic and user interface.
Tier 1
Server manages and controls access to database.
Database server
Tier 2
Database