**Department of Information Technology (BIT)**

Please fill in all the required details for your assignment to be accepted.

Homework Title: E-commerce Website Project Course with Code: BIT 244

Course Instructor: Bigyan Dhungana Program: BIT

Date of Allotment: 27th Feb. 2023 Date of Submission: 27th March 2023

Student’s LUC Reg. No.:LC0003000841 Semester: 4th (2nd Year)

**Declaration**

I declare that this is my individual work. I have not copied from any other student’s work or from any other source except where due acknowledgment is made explicitly in the text, nor has any part been written for me by another person.

Student’s Name : Hikmat Raj Ghodasaini

Email ID : ***hikmatraj.ghodasaini@pcmgmt.edu.np***

Contact No. : 9848485494

Evaluator’s Comments:

Marks obtained out of

Evaluator’s Signature

**Table of Content**

Contents

[Introduction 3](#_Toc130850287)

[1.1. Abstract 4](#_Toc130850288)

[2.System Analysis 5](#_Toc130850289)

[2.2. Existing System 7](#_Toc130850290)

[2.3. Hardware And Software Requirements 8](#_Toc130850291)

[JAVA 10](#_Toc130850292)

[Creation Of Java 10](#_Toc130850293)

[Java is Portable 10](#_Toc130850294)

[The following is a list of some of java’s features that make it easier to write correct code 12](#_Toc130850295)

[4.1. Data Flow Diagrams: 24](#_Toc130850296)

[System Implementation 29](#_Toc130850297)

[System Testing: 36](#_Toc130850298)

[Testing Analysis: 41](#_Toc130850299)

[Scope Of The Project 42](#_Toc130850300)

[Gantt Chart 43](#_Toc130850301)

[Conclusion: 44](#_Toc130850302)

# Introduction

shop is required for the following purposes: -

1. Capital works
2. Operation and Maintenance Works
3. Other Commercial activities like groceries, shopping malls, Food centers ,etc..

The Shops Management System is targeted to automate almost all of the processes mentioned above to reduce the clerical labor of the staff working in Shops both technical as well as Accounts departments using the software Industry’s latest technologies and cost-effective tools thereby providing better control to the management by avoiding manual errors etc….

In this project, the modules under study the are Material Issues the module, Reports module.

The material Issues module deals with the Island ues functionality of the application. It mainly contains two operations namely Material issues i.e. Issues issued to works based on field requisitions and Inter shop issues i.e. material issues to other shops based on inter-shop requisitions. For these two operations, we have to issue gate passes for both types of operations.

The reports module deals with the Reports provided by the application. This module contains various reports: Monthly SRB Report, Monthly SIB Report, Monthly Section Wise Issues Reports, Priced Ledger, Monthly Shops Abstract, Monthly Work Order wise Details, and Monthly Stock Report.

# 1.1. Abstract

**Project Title:** shop Management System

The system creates a web-based manufacturing system that enables a manufacturing industry to schedule its manufacturing operations based on the daily update of sales from its dealers. Once the sales figures of items for the past week are entered by the dealers over the internet along with the orders for the next delivery, the schedule for the next week’s production will be drawn up. A report of the required raw materials or parts will be drawn up with the product requirements over the internet & asked to quote their rates.

Once the rates are quoted, the order will be placed with the required delivery schedules. Once the parts are supplied the stocks will be updated. Then a production plan will be drawn up taking the bill of materials into consideration. Once the production plan is approved, the stock will be updated when the material is issued. Once the finished products are available the delivery schedules will be drawn up based on the orders placed by the Dealers. The stocks with the dealers will also be maintained.

The Benefits of the Shops Management System is

 It is used as an intranet Application.

* Providing High-Security.
* Easy Business Solutions.

# 2.System Analysis

**2.1. Software Requirement Specification (SRS)**

**What is SRS**?

Software Requirement Specification (SRS) is the starting point of the software development activity. As the system grew more complex, it became evident that the goal of the entire system cannot be easily comprehended. Hence the need for the requirement phase arose. The software projects us initiated by the client needs. The SRS is the means of translating the ideas of the minds of clients (the input) into a formal document (the output of the requirement phase.)

The SRS phase consists of two basic activities:

* 1. **Problem/Requirement Analysis:**

The process is order and more nebulous of the two, dealing with understanding the problem, the goal and constraints.

* 1. **Requirement Specification:**

Here, the focus is on specifying what has been found giving analysis such as representation, specification languages and tools, and checking the specifications are addressed during this activity.

The requirement phase terminates with the production of the validated SRS document. Producing the SRS document is the basic goal of this phase.

**Role of SRS**

The purpose of the Software Requirement Specification is to reduce the communication gap between the clients and the developers. Software Requirement Specification is the medium through which the client and user needs are accurately specified. It forms the basis of software development. A good SRS should satisfy all the parties involved in the system.

**Purpose**

The purpose of this document is to describe all external requirements for mobile task manager. It also describes the interfaces for the system.

# 2.2. Existing System

The existing system for Shops Management System activities uses open source standard & technologies. It had been developed on WINDOWS 2000 PROFESSIONAL platform with „POSTGRESQL‟ database. All the shops of the power distribution company limited are made through the open-source standards & technologies. The user interaction is in GUI (Graphical User Interface) mode.

Shops are required for the following purposes.

1. Capital works
2. Operation and Maintenance Works
3. Other Commercial activities like hiring equipment etc.

The „Shops Management System package‟ is targeted to automate almost all of the processes mentioned above to reduce the clerical labor of the staff working in Shops both technical and as well as Accounts departments using the software Industry’s latest technologies and cost-effective tools thereby providing better control to the management by avoiding manual errors etc.,

# 2.3. Hardware And Software Requirements

**2.3.1. Hardware requirements SERVER:**

|  |  |  |  |
| --- | --- | --- | --- |
| Processor | : |  | intel core i5 |
| Speed | : |  | 2.4 GHz |
| RAM | : |  | 8 GB |
| solid-state Capacity | : |  | 256 GB |
| Monitor Make  **Client:** | : |  | DELL |
| Processor | : |  | intel core i5 |
| Speed | : |  | 2.4 GHz |
| RAM | : |  | 8 GB |
| solid-state Capacity | : |  | 256 GB |
| Monitor Make | : |  | DELL |
| **2.3.2. Software Requirements** | | |
| Database : | | | Oracle |
| Programming Language : | | | Java |

**2**.4. **Feasibility Study**

The existing system is clearly understood the next step is to conduct the feasibility study, which is a high-level capsule version of the entire System Analysis and Design process. The objective is to determine whether the proposed system is feasible. The three tests of feasibility have been carried out:

* + 1. Technical Feasibility
    2. Economic Feasibility
    3. Operational Feasibility
    4. **Technical Feasibility**

In a technical feasibility study, one has to test whether the proposed system can be developed using existing technology or not. It is planned to implement the proposed system using Windows 2000 Professional, JSP and Apache Tomcat Web Server. The Organization already possesses Windows 2000 Professional Operating System. It is evident that the necessary hardware and software are available for the development and implementation of the proposed system. Hence the solution is technically feasible.

* + 1. **Economic Feasibility**

As part of this, the costs and benefits associated with the proposed system are to be compared and the project is economically feasible only if the benefits outweigh the costs. The Organization has already its own satellite link, and a host of SUN FIRE 6800 servers. So it need not invest new for the internet connection and also the organization initiated to use Open Source in project development, hence there is 0 additional cost incurred for the tools that will be used.

* + 1. **Operational Feasibility:**

This test of feasibility checks if the system works with least difficulties when it is developed and installed. The technical staff has sufficient knowledge of the tools being used and the users need just to know how to access and run the programs in the Apache Web Sever. Hence it is concluded that the system is operationally feasible.

**System Specification**

**3.1. System Environment and Tools**

# JAVA

# Creation Of Java

Java was conceived by James Gosling Patrick Naughton, Chris Wrath, Ed Frank, and Mike Sheridan at SUN MicroSystems Incorporation in 1991. It took 18 months to develop the first working version. This language was initially called “oak”, but was renamed “java” in 1995, many more contributed to the design and evolution of the language.

**Java is actually a platform consisting of three components:**

1. Java programming language.

1. Java library of classes and interfaces.

1. Java Virtual Machine.

**The following sections will say more about these components:**

Java is Portable:

One of the biggest advantages Java offers is that it is portable. An application written in Java will run on all the major platforms. Any computer with a Java-based browser can run the applications or applets written in the Java programming language. A Programmer no longer has to write one program to run on a Macintosh, another program to run on a windows machine, still another to run on a UNIX machine, and so on. In other words, with Java, developers write programs only once. Rather than being compiled into machine language, which is different for each operating systems and computer architecture, Java code is compiled into byte codes.

With other languages, the program code is compiled into a language that the computer can understand; the problem is that other computers with different machine instruction sets cannot understand that language. Java code, on the other hand, is compiled into byte codes rather than a machine language. These byte codes go to the Java virtual machine, which executes them directly or translates them into the language that is understood by the machine running it.

In the summary, this means that with the JDBC API extending Java, a programmer writing Java code can access all the major relational databases on any platform that supports the Java virtual machine.

**Java Is Object-Oriented:**

Java programming language is object-oriented, which makes program design focus on what you are dealing with rather than on how you are going to do something. This makes it more useful for programming in sophisticated projects because one can break things down into understandable components. A big benefit is that these components can then be reused.

Object Oriented languages use the paradigm of classes. In simplest terms, a class includes both the data and the functions to operate on the data, all the data members and the functionality of its class. Because of this, you can think of a class as being the like template, with each object being a specific instance of a particular type of a class.

The class paradigm allows one to encapsulate data so that specific data values are those using the data cannot see function implementation. Encapsulation makes it possible to make changes in code without breaking other programs that use that code. If for example the implementation of a function is changed, the change is invisible to the programmer who invokes that function, and it does not affect his/her program, except hopefully to improve it.

Java includes inheritance, or the ability to derive new classes from existing classes. The derived class, is referred to as the parent class. A subclass can add new data members to those inherited from the parent class. As far as methods are concerned, the subclass can reuse the inherited methods as it is, change them, and/or add its own new methods.

**Java Makes It Easy to Write Correct Code:**

In addition to being portable and object-oriented, java facilitates writing correct code. Programmers spend less time writing java code and a lot less time debugging it. In fact, development time reduces by as much as two-thirds.

The following is a list of some of java’s features that make it easier to write correct code:

**Garbage Collection:**

Automatically takes care of allocating and deallocating memory, a huge potential source of errors (Memory Leaks). If an object is no longer being used, then it is automatically removed from memory, or “garbage collected”. Programmers don‟t have to keep track of what has been allocated and deallocated themselves, which makes their job a lot easier, but more importantly it stops memory leaks.

**No Pointers:**

Eliminates big source errors. By using object references instead of memory pointers, problems with pointer arithmetic are eliminated, and problems with inadvertently accessing the wrong memory address are greatly reduced.

**Strong Typing:**

Cuts down on runtime errors. Because java enforces strong type checking, many errors are caught when code is compiled. Dynamic binding is possible and often very useful, but static binding with strict type checking is used when possible.

**Simplicity:**

Makes java easier to learn and use correctly. Java keeps it simple by having just one way to do something instead of having several alternatives, as in some languages. Java also stays lean by not including multiple inheritances, which eliminates the errors and ambiguity that arise when you create a subclass that inherits from two or more classes.

To replace capabilities, multiple inheritances provide, Java lets you add functionality to a class throw the use of interfaces.

**Java Includes a Library of Classes And Interfaces:**

Java platform includes an extensive class library so that programmers can use already existing classes, or implement interfaces to augment the capabilities of classes.

Both classes and interfaces contain data members and functions, but there are major differences. In a class, fields may be either variable or constant, and methods are fully implemented. In an interface, fields must be constants, and methods are prototypes with no implementations. The prototypes give the method signature (the return type, the function name, and the number of parameters with the type for each parameter), but the programmer must supply implementations. To use an interface, a programmer defines a class, declares that it implements the interface, and then implements all the methods in that interface as part of the class.

These methods are implemented in a way that is appropriate for the class in which the methods are being used. Interfaces let one add functionality to a class and give a great deal of flexibility in doing it. In other words interfaces provide most of the advantages of multiple inheritance without its advantages.

A Package is a collection of related java classes and interfaces. The following list, though not complete, gives examples of some java packages and what they cover.

**Java. lang**--the basic classes. This package is so basic that it automatically is +include in any java program. It includes classes dealing with numeric, Strings, objects, runtime, security and threads.

**Java.io--** classes that manage reading data from input streams and writing data to the output streams.

**Java.util**-- miscellaneous utility classes, including generic data structures, bit sets, time, date, the string manipulation, random number generation, system properties, notification and enumeration of data structures.

**Java.net**--classes for network support.

**Java.awt**--classes that manage user interface components such as windows, dialog boxes, buttons, checkboxes, lists, menus, scrollbars, and text fields the “AWT” stands Abstract window Toolkit.

**Java.awt.image**--classes for managing image data, including color models, dropping, color filtering, setting pixel values, and grabbing snapshots.

**Java.applet**--The Applet class, which provides the ability to write applets, this package also includes several interfaces that connect an applet to its document and to resources for playing audio.

It is important that a programmer not be able to write subversive code for applications or applets. This is especially true with the Internet being used more and more extensively for services such as electronic commerce and electronic distributions of software and multimedia content.

**The Java platform builds in security in four ways.**

The way the memory is allocated and laid out: in Java objects location in memory is not determined until the runtime, as opposed to **C** and **C++,** where the compiler marks memory layout decisions. As the result, a programmer cannot look at a class definition and figure out how it might be laid out in memory. Also since, java has no pointers, a programmer cannot forge pointers to memory.

**The Way Classes Are Loaded:**

The java virtual machine doesn’t trust any incoming code and subjects it to what is called Byte code verification. The byte code verifier, part of the virtual machine, checks that

**A:**  The format the f incoming code is correct.

**B:** Incoming code doesn’t forge pointers.

**C:** It doesn’t violate access restrictions. **D:**  It accesses objects as they are.

The Java byte code loader, another part of the virtual machine, checks whether classes loaded during program execution is local or forms across a network. Imported classes cannot be substituted for built in classes, and built in classes cannot accidentally references classes brought in over a network.

**The Way Access Is Restricted For Un trusted Code:**

Java security manager allows users to restrict un trusted Java applets so that they cannot access the local network, local files and other resources.

**Java Performs Well:**

Java performance is better than one might expect. Java‟s many advantages, such as having built in security and being interpreted as well as compiled, do have a cost attached to them. However, various optimizations have been built in, and the byte code interpreter can run very fast the cost it does not have to do any checking. As a result, java has done quite respectably in performance tests. Its performance numbers for interpreted byte codes are usually more than adequate to run interactive graphical end user applications.

For situation that require unusually high performance, byte codes can be translated on the fly, generated the final machine code for the particular CPU on which the application is running at run time.

High level interpreted scripting languages generally offer great portability and fast prototyping but poor performances. Low level compiled languages like **C** and **C++** offer great performances but require large amounts of time for writing and debugging the code.

Because of problems with areas such memory management, pointers and multiple inheritance. Java offers good performance with the advantages of high level languages but without the disadvantages of **C** and **C++.** In the world of design trade-offs, you can think of Java providing a very attractive middle ground.

**Java Server Pages (JSP)**

A JSP page is a text page document that describes how to process request to create a response. JSP is a Java-based technology that simplifies the process of developing dynamic web sites. With JSP, web designers and developers can quickly incorporate dynamic elements into web pages using embedded Java and simple mark-up tags. These tags provide the HTML designer with a way to access data and business logic shop inside Java object.

**Application Model:**

JSP pages can be used in combination with servlets, http, html, xml, applets, java beans components and enterprise java beans components, to implement a wide variety of application architecture(s)or models. To facilitate web-based applications, both servlets and JSP are required elements of j2ee specifications. And while both technologies can be used to dynamically generate html to be send to an end-users web browser, only JSP enforces a strong separation between presentation logic involved in displaying data and the business or programming logic used to generate data in first place. When using JSP, changes to presentation of the data do not require any changes to the underlined server code that supplied data in the first place.

**The java server pages technology offers the following advantages:**

* Write once, run anywhere properties
*  High-quality tool support.
* Reuse of components and tag libraries.
* Separation of dynamic and static content:
* Support for scripting and actions
* Web access layer for N-tier enterprise application architecture.

**Executing JSP:**

**Adding JSP support:**

The most basic requirements for using JSP are a web server implies. Both hardware in the form of a computer accessible over the Internet or a corporate based intranet and s/w in the form of an HTP server running on that hardware. Some of most popular http servers are: apache, Microsoft Internet information sever, web logic etc.

JSP execution start with request for a JSP page, the processing is done based on JSP tag on the page In order to generate contents dynamically the output of the processing is combined with the page static HTML, and presented on the web browser.

**Scripting Languages:**

The default scripting language for JSP is, naturally, java. Since JSP Pages are compiled into java servlets, this assumption makes the translation of the scripts into servlets code very straight forward. To be acceptable for use with JSP, scripting languages must meet three requirements:

1. It must support the manipulation of Java objects.
2. It must include the ability to catch Java exceptions and specify exception handlers.
3. It must be able to invoke methods on Java objects.

**JSP Tags:**

JSP provides four major categories of markup tags:

1. Directives, is a set of tags for providing the JSP container with page specific instructions for how the documents containing the directives are to be processed.

1. Scripting elements are used to embedded programming instructions written in the designated scripting language for the pages which are to be executed each time the page is processed for request.

1. Comments are used for adding documentation strings to a JSP page.

1. Action support different behaviors.

|  |  |
| --- | --- |
| <%=%> | : USED FOR EXPRESSIONS |
| <%%> | : USED FOR SCRIPLETS |
| <%!%> | : USED FOR DECLARATION |
| <%@%> | : USED FOR DIRECTIVES |
| <%@page%> | : USED FOR PAGE DIRECTIVES. |

**Expressions:**

A JSP expression element is a scripting language expression that is evaluated and that can be converted into a string which is succeed emitted into the out object.

Syntax: <%= expression%>

**Script:**

Script can contain any code fragment that is valid for scripting language frfragments

Syntax: <%scrip let %>

**Declarations:**

These are used to declare variables and methods in the scripting language.

A declaration is initialized when the JSP page is initialized and is made available to other declarations and scripts.

Syntax: <%! Declarations%>

**Directives:**

These are messages to JSP engine.

Syntax: <%@directives%>

**Page Directives:**

The page directive defines a number of page dependent attributes and communicates these to the JSP engine.

Syntax: <%@page = directive-attribute-list%>

**About JDBC**

The JDBCTM API provides universal data access from the JavaTM programming language. Using the JDBC 3.0 API, you can access virtually any data source, from relational databases to spreadsheets and flat files. JDBC technology also provides a common base on which tools and alternate interfaces can be built.

**The JDBC 3.0 API is comprised of two packages:**

1. The java.sql package
2. The javax.sql package, which adds server-side capabilities

The JDBC-ODBC Bridge allows applications written in the JavaTM programming language to use the JDBCTM API with many existing ODBC drivers.

The Bridge is itself a driver based on JDBC technology ("JDBC driver") that is defined in the class “ sun.jdbc.odbc.JdbcOdbcDriver ” The Bridge defines the JDBC sub protocol Odbc.

**Status of the JDBC-ODBC Bridge:**

The JDBC-ODBC Bridge should be considered a transitional solution. Sun Microsystems and Mirant are working to make the Bridge more reliable and robust, but they do not consider it a supported product. With the development of pure-Java JDBC drivers, the JDBC-ODBC Bridge should become unnecessary.

**Why not just use ODBC?**

The JDBC-ODBC Bridge offers several advantages over "naked" ODBC.

* The ODBC API is strongly oriented toward C/C++ programmers. The JDBCODBC Bridge makes it unnecessary for programmers using the Java programming language to deal with non-Java language concepts.
* The ODBC API is extremely complex and intermingles high-level and low-level functions. The JDBC-ODBC Bridge allows the programmer to rely on the JDBC API, which is relatively simple and easy to learn.
* The JDBC-ODBC Bridge lets a program deal with ODBC through a standardized JDBC interface. That helps a program avoid being tied to ODBC when a better solution comes along.

**System Design**

# 4.1. Data Flow Diagrams:

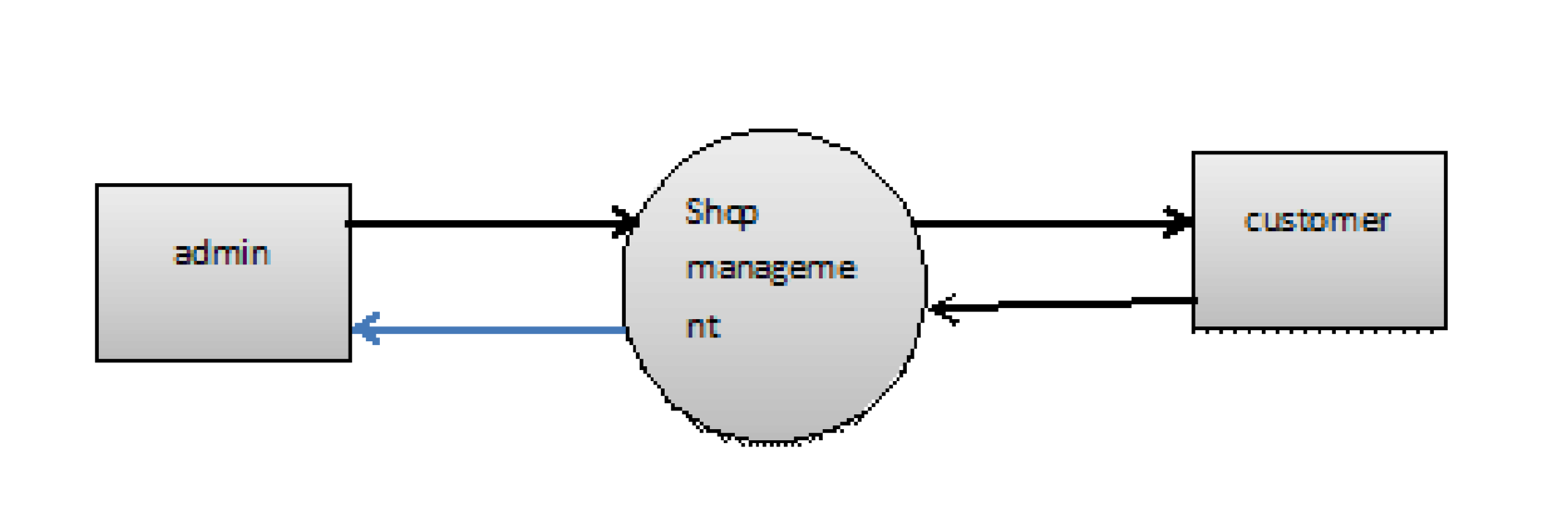
The data flow diagrams from the important modeling tools in the structure system analysis methodologies. The data flow diagrams are on of the most important tools used by system analyst.

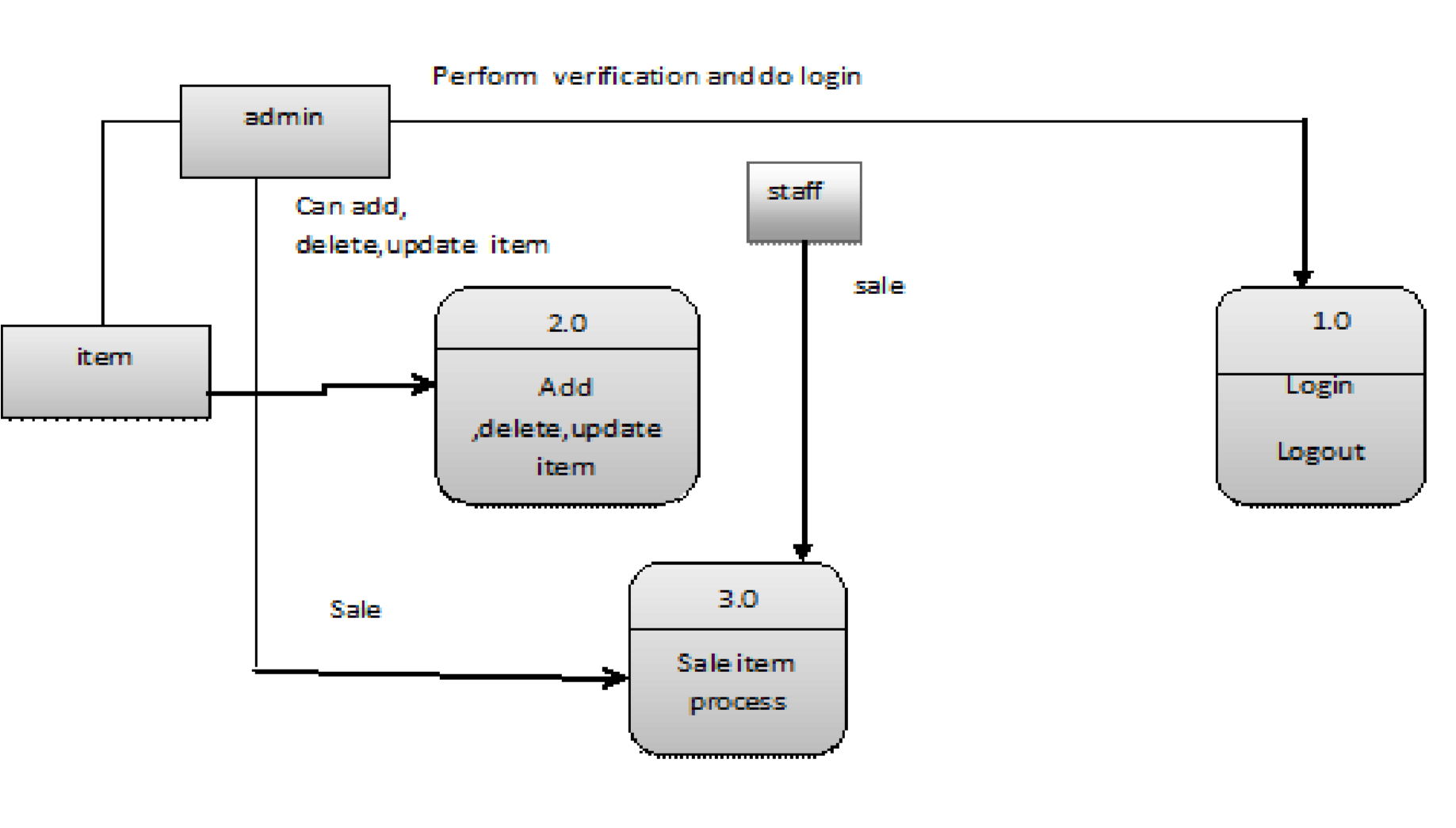
Data flow diagram should be the first tool used by the system analyst to model system components. There are three kinds of system components.

* 1. Process
  2. Entity
  3. Data flow

**Data flows:**

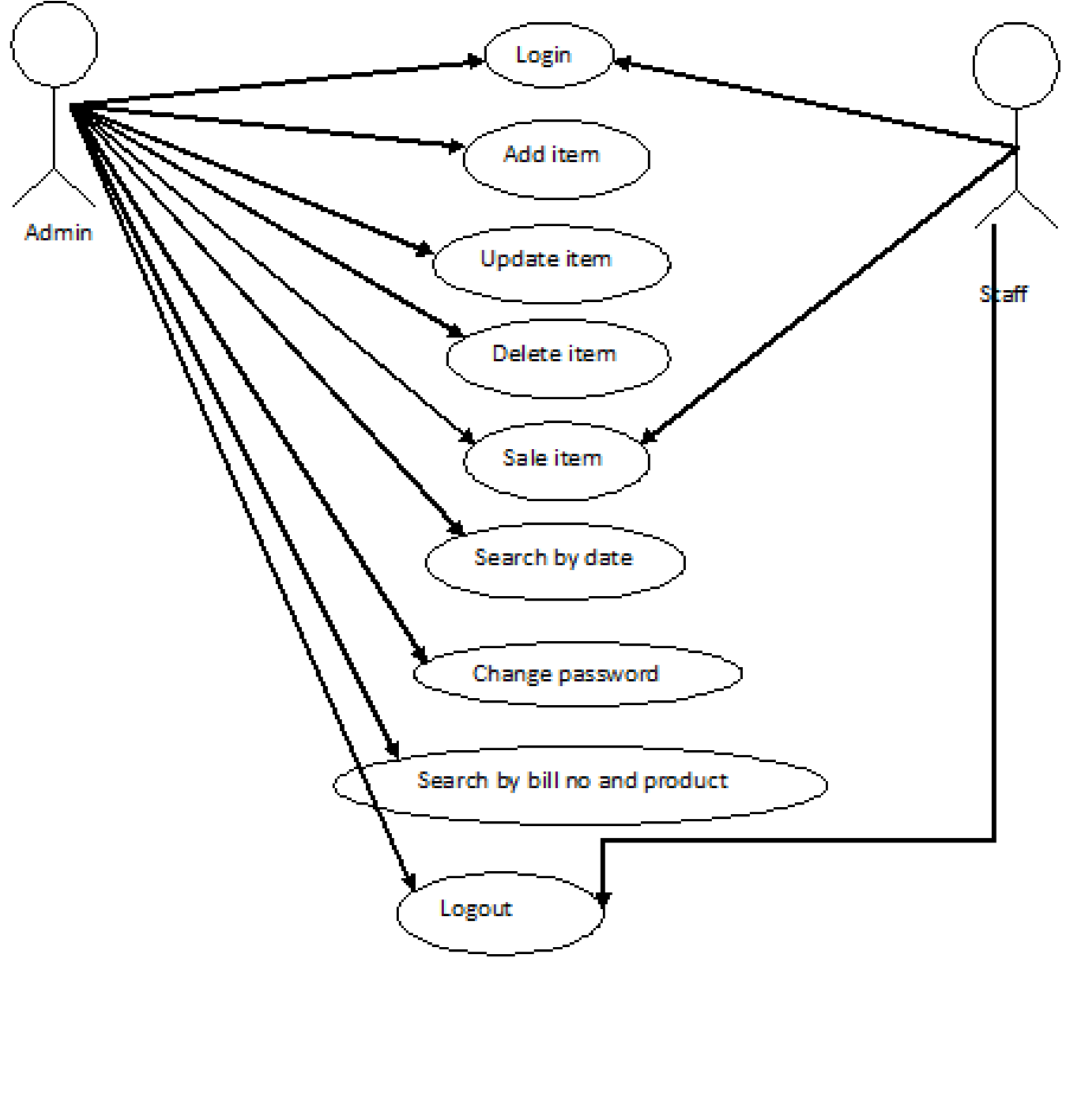
Data flows model the passage of the system and are represented by lines joining the system components. An arrow indicates the direction of the flow and the lines is labeled by the name of the data flow. Flow of data in the system can take place.



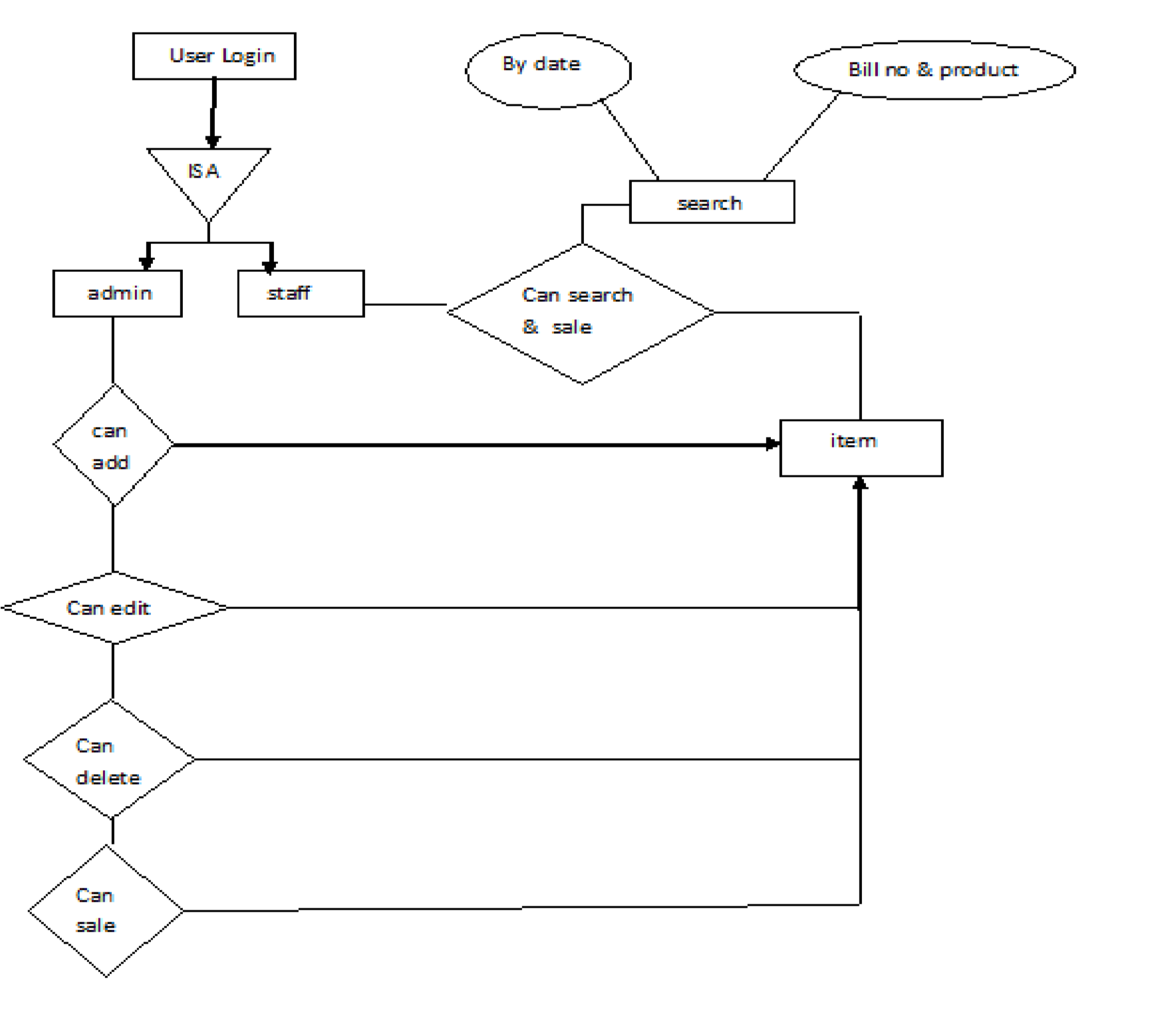


**4.2 Use Case Diagram:**

# 



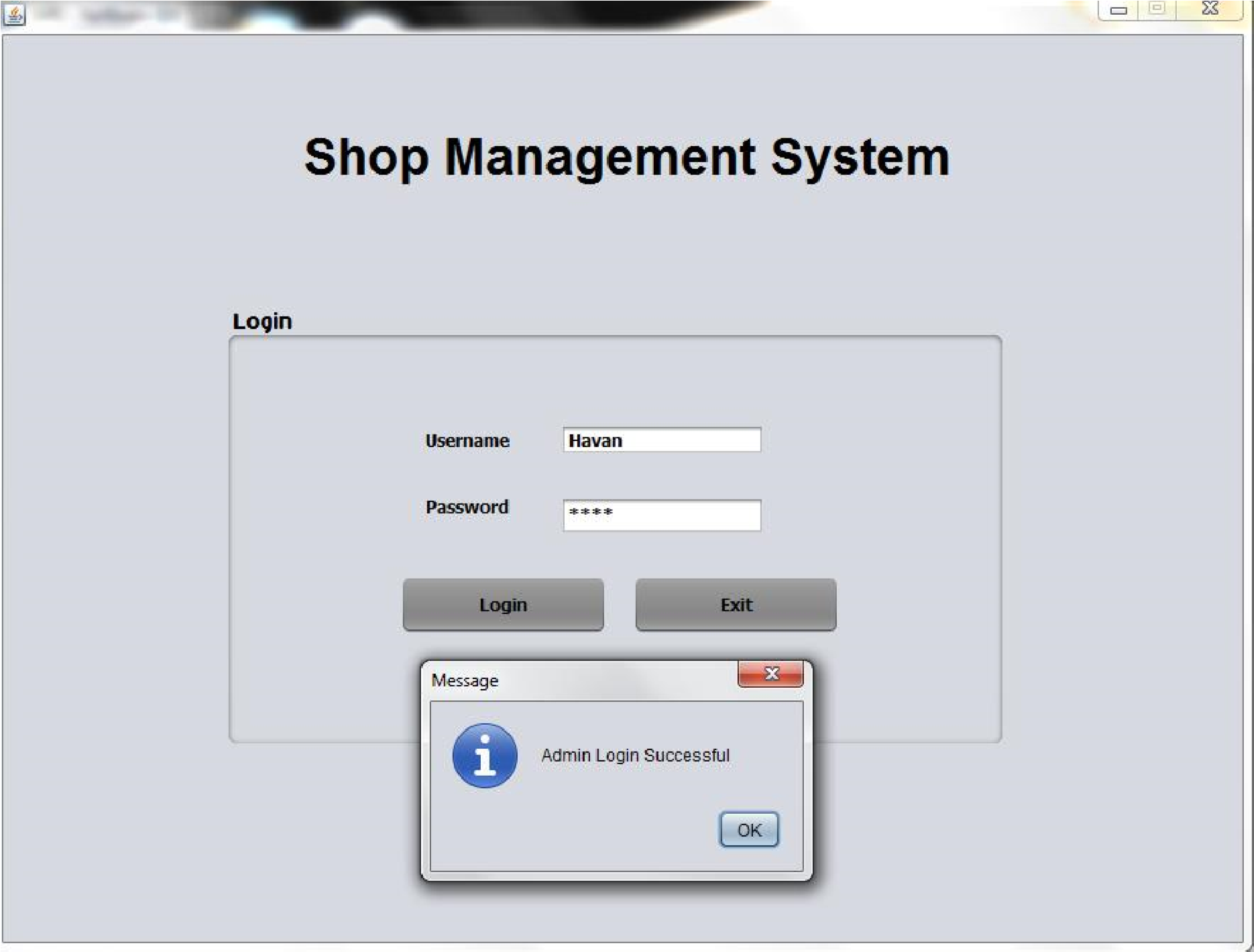
**4.3 E-R Diagram:**



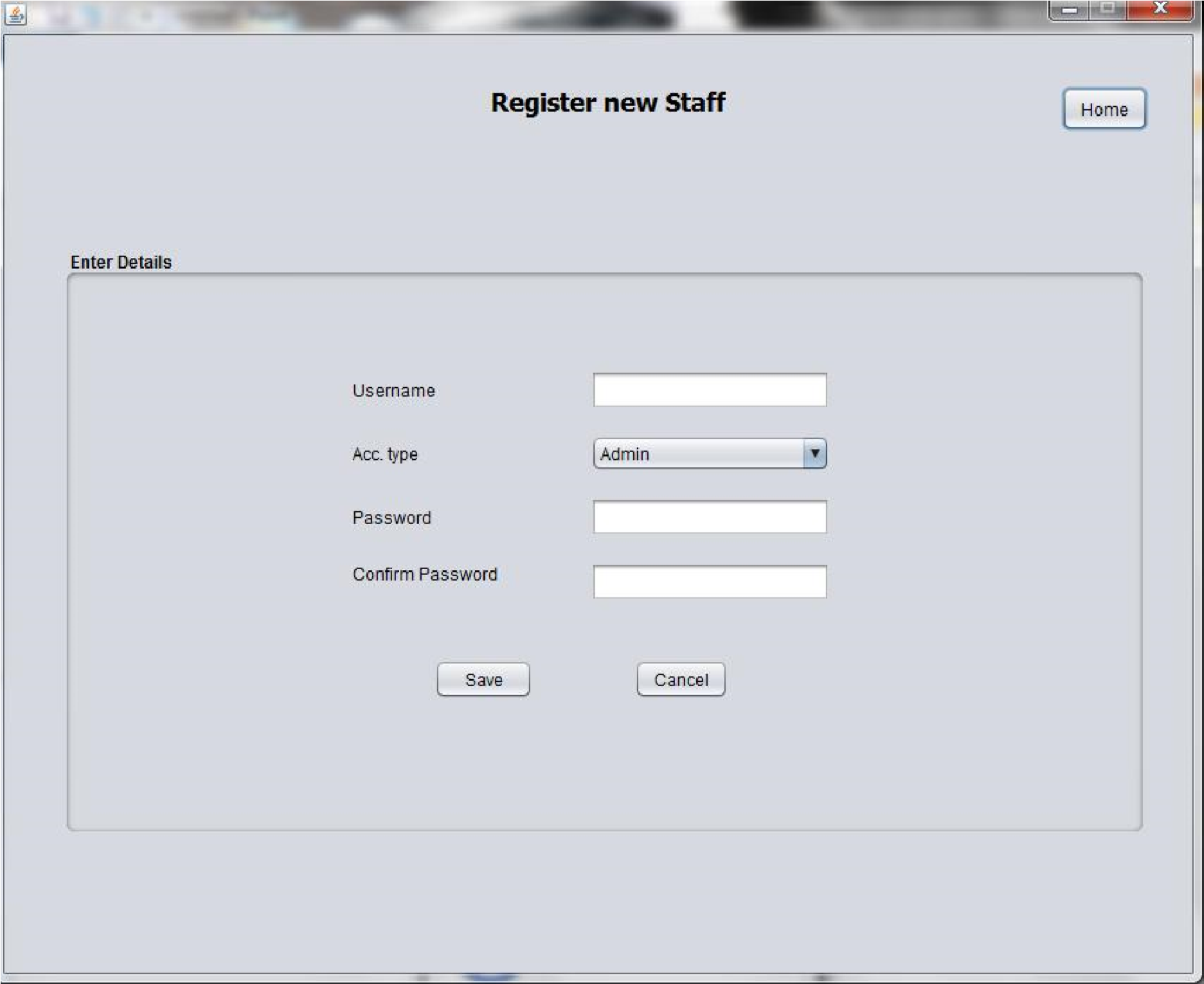
# System Implementation

**5.1. Screens**

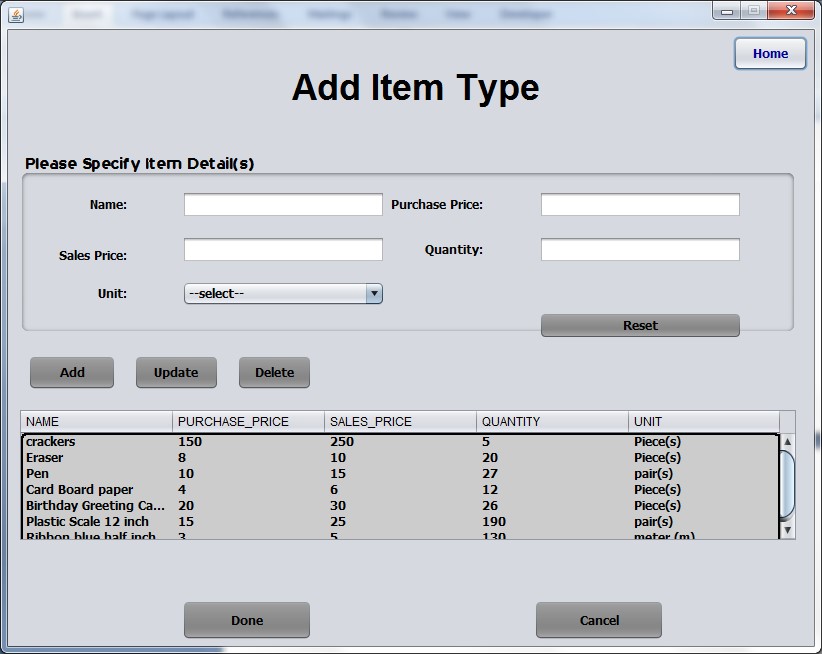
**Login Page:**



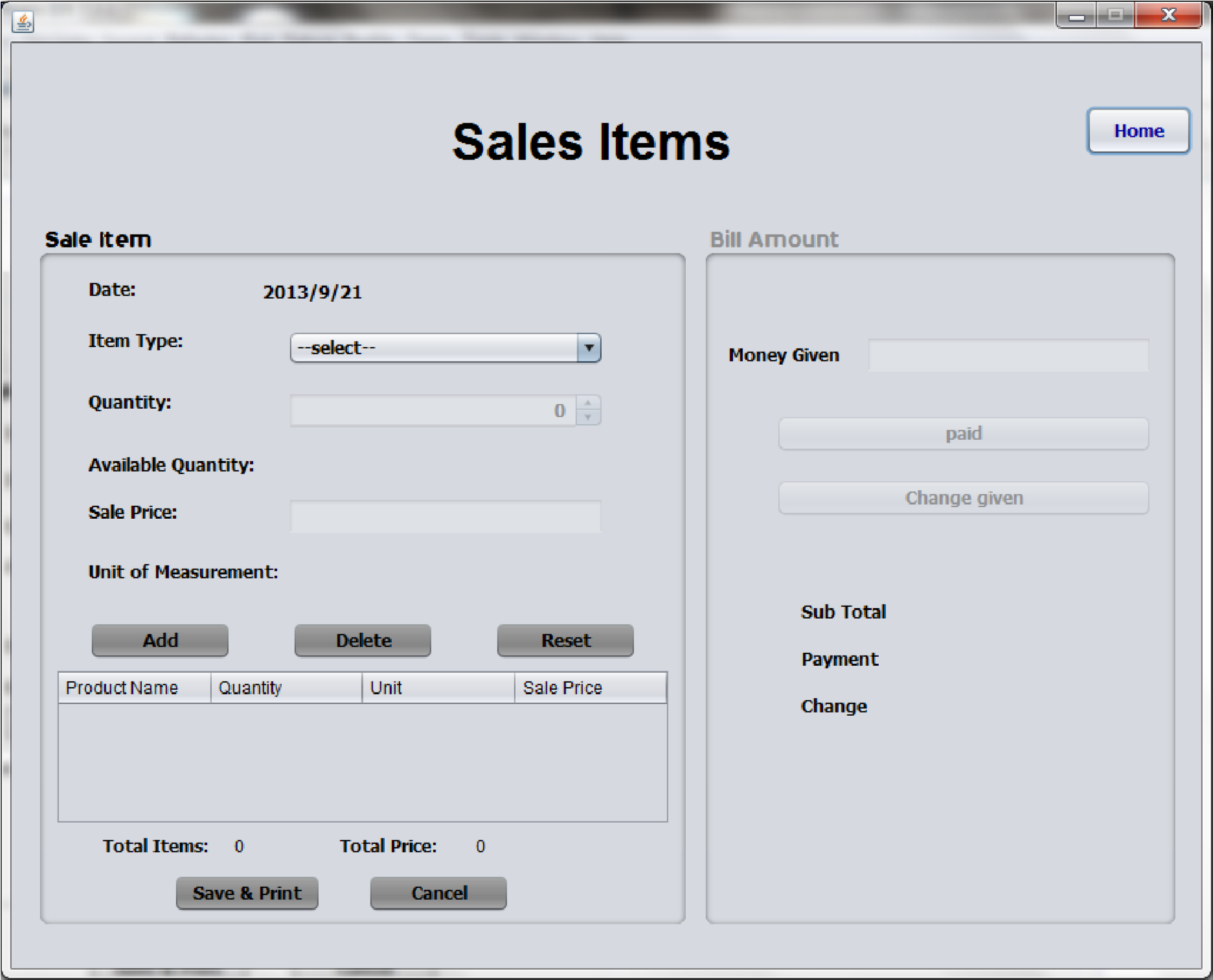
**Administrator/Staff Registration:**



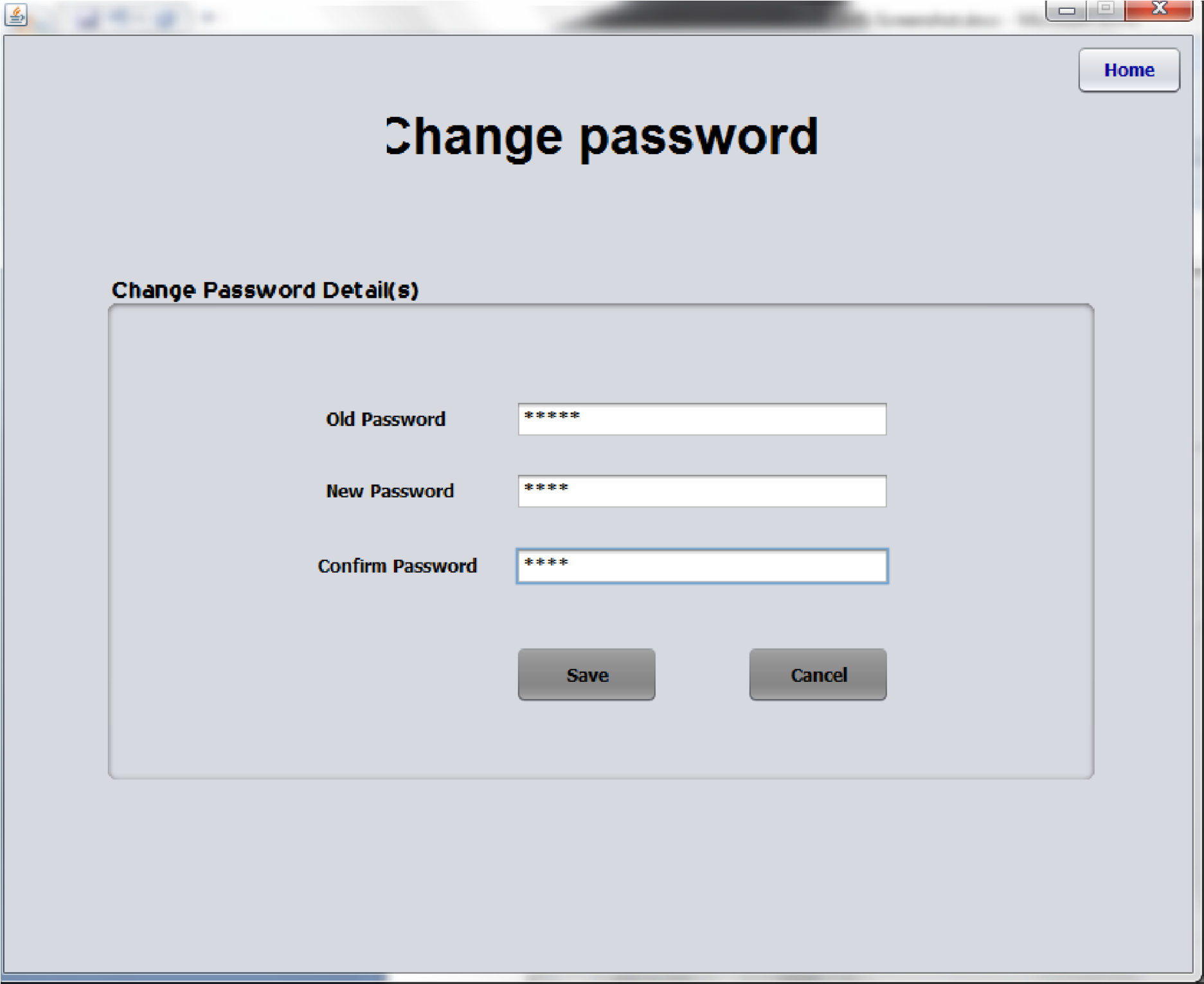
**Add Items:**



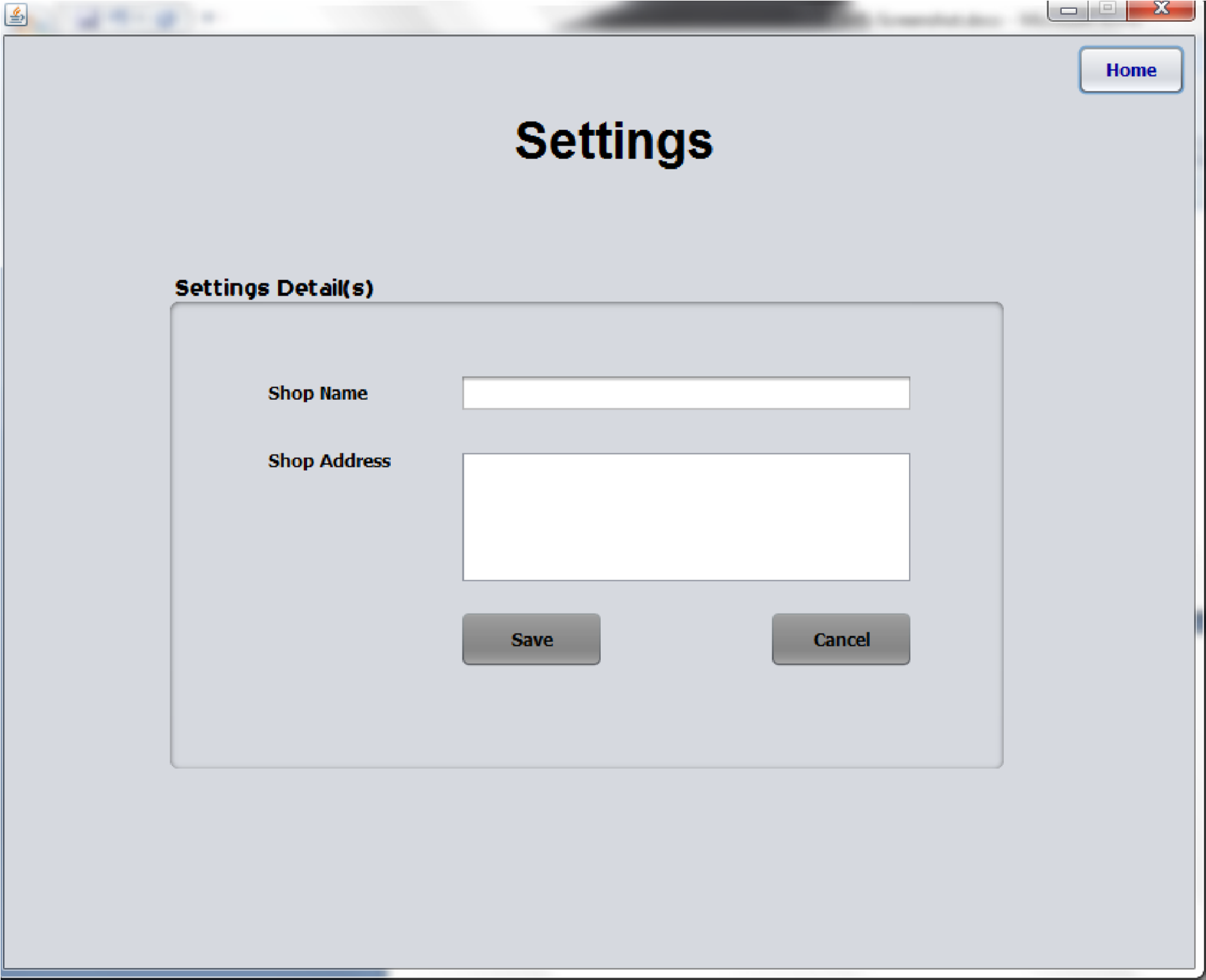
**Sales Item List:**



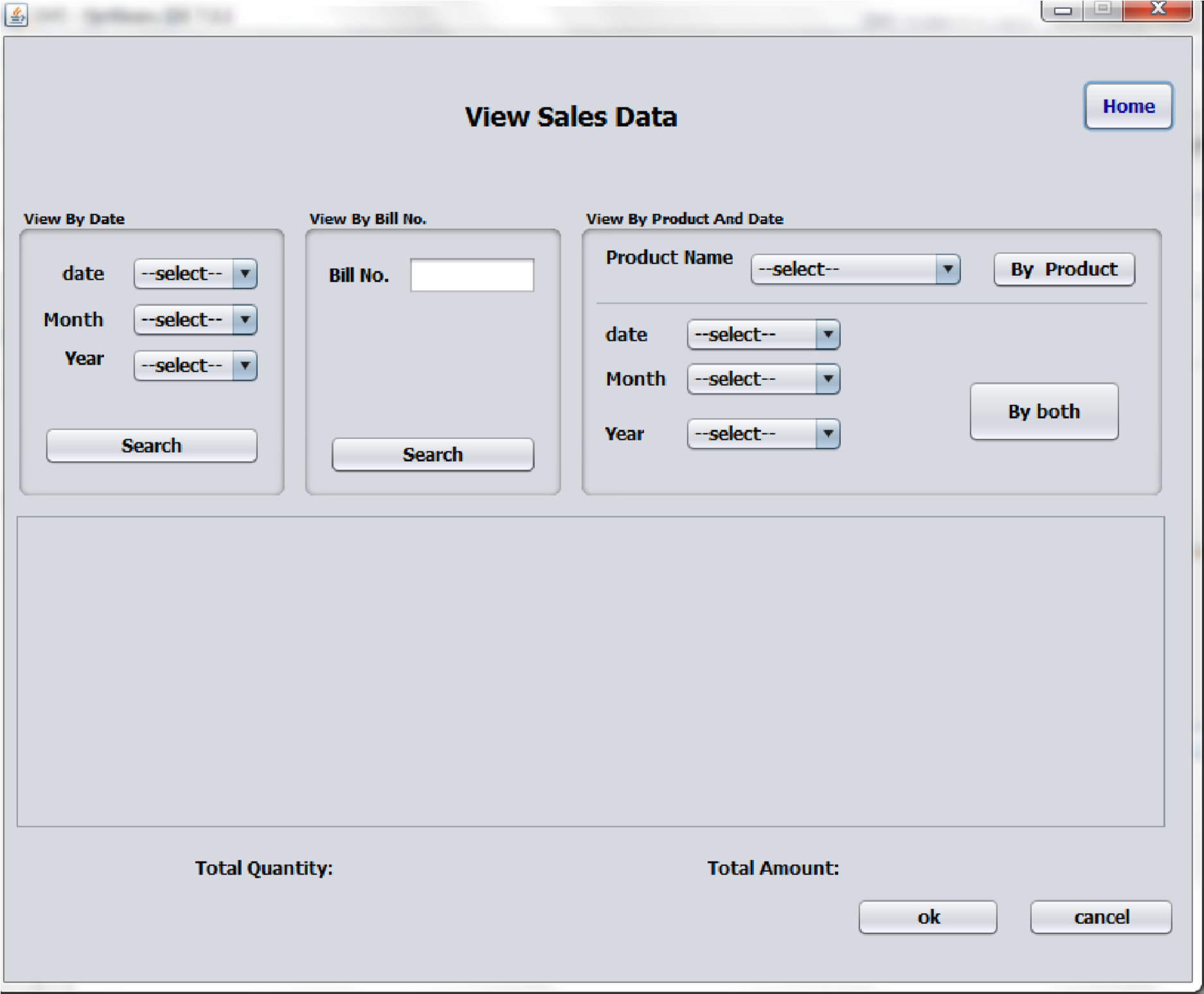
**Change Password:**



**Settings:**



**View Sales Data:**



# System Testing:

The development of Software system involves a series of production activities. There is a chance of errors to occur at any stage. Because of human inability to perform and communicate with perfection, a Quality Assurance Activity accompanies software development.

Software testing is a critical element of software quality assurance and represents the ultimate review of specification, design and code generation.

The increasing visibility of software as a system element and the costs associated with software failure are motivating forces for well-planned, thorough testing.

For testing the system we followed the strategies given below.

**Testing Techniques:**

Different types of testing are

* Boundary Condition Testing
* Integration Testing
* Black Box Testing  Validation Testing
* User Acceptance Testing

During the implementation for the system each module of the system is tested separately to uncover errors within its boundaries. User interface is used as a guide in this process.

The validations have been done for all the inputs using Java Script.

For example to check whether the work allotted among the database correctly without exceeding the schemes which are not validated thoroughly and the internal database has to check the reflections in the database.

**Boundary conditions Test:**

Boundary conditions as in case of generating sequences were tested to ensure that the module operates properly at boundaries establish to limit or restrict processing also it is able to handle incorrect out of the boundary values properly.

**Integration Test:**

The objective of Integration Test is to take the until tested modules and build a program structure that has been defined in the design. We have done top down integration, which is constructing and testing small segments where errors are easier to isolate, and correct. The integration process was performed in three steps:

* The main control was used as test driver.
* Test was conducted as each module was integrated.
* Regression testing to ensure that new errors have not been introduced due to the corrections.

**Block Box Testing:**

It focuses on functional requirements of the software. Block box testing attempts to find errors in the following categories.

Incorrect or missing function

Interface error

Errors in external device access

Performance error

Initialization and termination errors

The above testing was successfully carried out for the developed system.

# 

**Validation Testing:**

At the culmination of integration testing, software is completely assembled as a package, interfacing errors have been uncovered and corrected, and a final series of software tests namely validation tests are performed. Validation succeeds when the software functions in the manner that can be easily accepted by the customer.

After the validation test has been conducted, one of the possible conditions is satisfied. The functions or performance characteristics confirmed to specifications are acceptable. The deviation form specifications are uncovered and a note of what is lacking is made. The developed system has been tested satisfactorily to ensure its performance is satisfactory and it is working efficiently.

**User Acceptance Testing:**

User acceptance of a system is a key factor for the success of any system. The system under consideration was tested for user acceptance constantly, by keeping the users informed of the progress and incorporating changes suggested, at the development time itself.

**Test Case Report:**

Here we specify all the test cases that are used for system testing. The different conditions that need to be tested along with the test case used for testing those conditions and the expected outputs are given. The goal is to test the different functional requirements. Test cases have been selected for both valid and invalid inputs.

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No** | **Test case** | **Condition** | **Expected Output** |
| 1 | Get Systems | Input Domain name | Print list of all system in current domains & response time |
| 2 | Get User | Input Domain name | System id, user id, port no, domain name |
| 3 | Get Processes details | Select process | Output the details of  processes |
| 4 | Get modules details | Select process & select thread opt | Details of modules |
| 5 | Get thread details | Select process & select thread opt | Details of threads |
| 6 | Stop the processes | System id, user id, password | Process close |
| 7 | Stop the system | System id | System close |

# Testing Analysis:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **S.No** | **Testing object** | **Expected value** | **Simulated value** | **Explanation** | **Remarks** |
| 1 | User name &  Password | AEIND  GUEST | AEIND  GUEST | Equal of expected and simulated values | Pass |
| 2 | User name &  Password | AEIND  GUEST | AEIND  GUEST | Unequal of expected and  simulated value | Fail |
| 3 | Change password | GUESS | GUEST  (Old password) | Equal of these two passwords | Pass |
| 4 | Start time and end time |  |  | Equal of these times | Pass |
| 5 | Start date and end date |  |  | Equal of these dates | Pass |

# Scope Of The Project

**Shops Management System:**

The proposed system, Shops Management System ‟can be made broadly available in small scale as well as large-scale shops/stores.

Today not every shop is computerized nor do all of them have an automated billing and access control system. But this application will change its face of it. The current software’s and applications available in the market are also not so user-friendly. This system is GUI based system and is user-friendly. Shops Management System is accessible through the internet. Shops are required for the following purposes.

1. Capital works
2. Operation and Maintenance Works
3. Other Commercial activities like hiring equipment etc..,

The Shops Management System package is targeted to automate the almost all of the processes mentioned above to reduce the clerical labour of the staff working in

Shops both technical and as well as Accounts departments using the software industry‟s latest technologies and cost effective tools there by providing the better control to the management by avoiding manual errors etc.

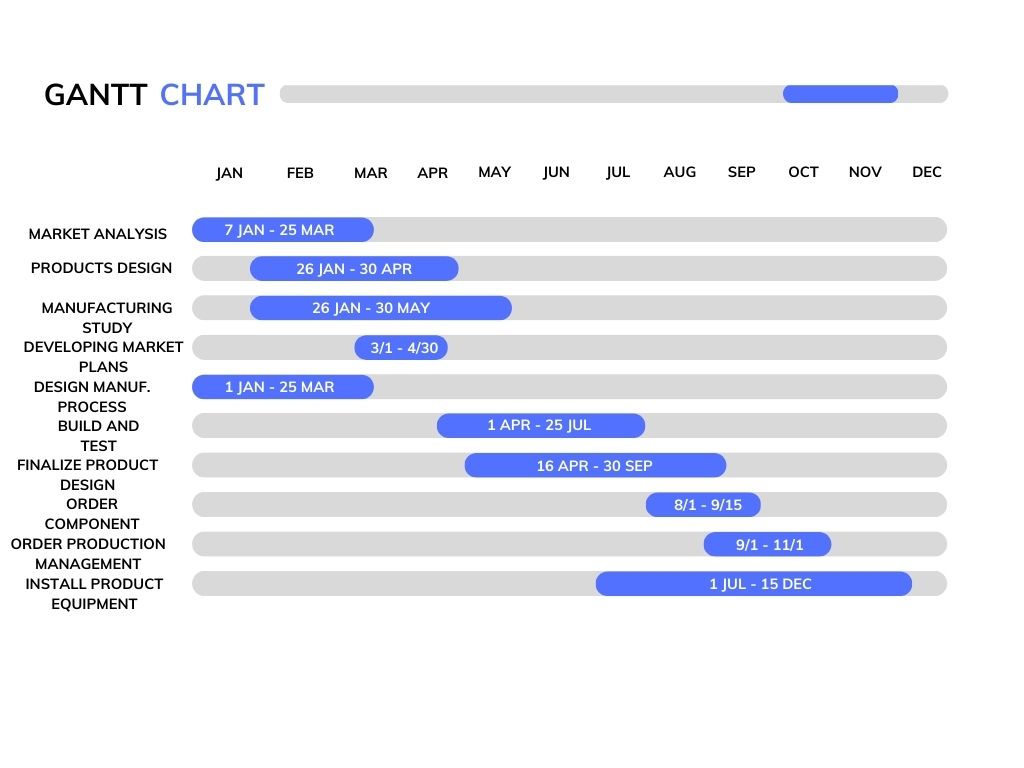
The software package is described in detailed below. The package provides these basic functionalities:

* Access Control
* Manage Items
* Billing of Items
* Bill Printing
* Viewing Of Sales Data -View by Date

-View by Bill No.

-View by Product Name

# Gantt Chart

****

# Conclusion:

Thus, we have successfully completed our diploma final year project of „Shop Management System‟ according to the market needs as well as the guidelines given by our Project Guide and other teaching staff. The experience of completing our project was not only fulfilling but also a great learning curve. We followed the waterfall model of project development. It is one of the best and efficient models for project development.

Testing of the software was done by all the members at different levels and by using the method of role-play so that it was tested more thoroughly and efficiently.

Lastly, we are thankful to the support extended by the project guide and various other people directly or in directly.