Capgemini

Warehouse Management Application

High Level Design

Document Control :

Warehouse Management Application				
Guided by- BISWA				
Date	**Version**	**Author**	**Brief Description of Changes**	**Approver Signature**
20-10-22	1.0	Swapnil, Delsy, Sudhansu, Benitta and Sayadeevi		

Introduction

1 Purpose

The purpose of this project is to facilitate the maintenance of a real-time warehouse database. This system can handle large inventories of an organization. The system can also be used to track the inventory of a single store, or to manage the distribution of stock between several stores of a larger franchise.

1.1 Intended Audience:

Our target users would be comprised of suppliers and stockist. Suppliers are the user who would supply items to the Warehouse. Stockist are users who make requests or place orders to take the from the Warehouse.

1.2 Project Goal:

The main goal of the Warehouse Management System is to ensure consistent availability of supplies for consumers.

Hence this System is directed toward owners of small to large stores and stock managers who are responsible for maintaining sufficient goods on hand in a retail or manufacturing business.

The system is designed to reduce the strain of tracking then to handling all the store maintenance.

1.3 Key Project Objectives:

1. Allow users (suppliers and stockist) to register by themselves (ID is system generated).

2. User login.

3. Validate usernames and their respective passwords

4. Allow interactions between admin, stockist and suppliers.

5. Apprising admin for a delivery using a status indicator.

6. Admin's access to all records.

7.Admin can make reports for the items which are in demand.

8. Admin can make reports for the stockist who order regularly.

1.4 Project Scope and Limitation:

Inexperienced users can register, and existing users can login with their respective ID’s and passwords.

In case of the Supplier, he/she will supply only three items at a time. Supplier cannot see the current stock. In the case of Stockist, he/she can order only three items at a time. Stockist can also make requests for unavailable items.

1.5 In Scope:

It provides a general way to manage a warehouse inventory and handling real-time warehouse database of an organization. It can be used to track the inventory of a single store, or to manage the distribution of stock between several stores of a large franchise.

The application is written in C language and is divided into three modules: Warehouse Admin, Supplier and Stockist. Warehouse Admin updates the entire file of warehouse. Suppliers are the user who will provide the supply of items to the Warehouse. Stockist are users who make requests to take the from the Warehouse.

1.6 Functional Overview

The following functions are included in the program:

Supplier:

1) int supplier_add_product( ) -- By using this function, the supplier will add the products.

2) int supplier_edit_product( ) -- By using this function, the supplier will edit the products.

3) int supplier_delete_product( ) -- By using this function, the supplier will delete the products.

4) int supplier_edit_details( ) -- By using this function, the supplier can edit his personal details.

5) int rol_check( ) -- By using this function, the supplier checks the ROL(Re Order Level).

Stockist:

1) int order_items( ) -- By using this function, the stockist orders the products.

2) int view_stock( ) -- By using this function, the stockist can view the stock database.

3) int request_for_items( ) -- By using this function, the stockist requests for the products, which are required.

4) int stockist_edit_details( ) -- By using this function, the stockist can edit his personal details.

Admin:

1) int add_supplier( ) -- By using this function, the admin can add supplier records.

2) int del_supplier( ) -- By using this function, the admin can delete supplier records.

3) int edit_supplier( ) -- By using this function, the admin can edit supplier records.

4) int view_supplier( ) -- By using this function, the admin can view supplier records.

5) int add_stockist( ) -- By using this function, the admin can add stockist records.

6) int del_stockist( ) -- By using this function, the admin can delete stockist records.

7) int edit_stockist( ) -- By using this function, the admin can edit stockist records.

8) int view_stockist( ) -- By using this function, the admin can view stockist records.

9) int add_product( ) -- By using this function, the admin can add product records.

10) int del_product( ) -- By using this function, the admin can delete product records.

11) int edit_product( ) -- By using this function, the admin can edit product records.

12) int view_product( ) -- By using this function, the admin can view product records.

13) int display_report_items( ) -- By using this function, the admin can display reports of high demand products.

14) int display_report_regular( ) -- By using this function, the admin can display the report of registered stockist who order regularly.

2 Design Overview:

Instant Chatters comprises of the following modules:

Name of the Module	Admin , Supplier & Stockist
Handled by	Swapnil Singh
Description	Handles all functions associated with each module

Name of the Module	Admin , Supplier & Stockist
Handled by	J Benitta
Description	Display reports; Validations

Name of the Module	Admin , Supplier & Stockist
Handled by	Delsy Mary W
Description	File handling operations; Validations

Name of the Module	Admin , Supplier & Stockist
Handled by	Sayadeevi N
Description	UT-IT; ROL check; Splint; Valgrind

Name of the Module	Admin , Supplier & Stockist
Handled by	Sudhansu Rai
Description	Handles all functions associated with each module

2.1 Design Objectives:

This project is made with keeping the requirements of the suppliers and stockist in mind. A simple and minimal interface is kept for all users to allow operations in a methodical manner.

The warehouse owners, suppliers and stockist are provided with an interactive menu which allows them to manage their stocks. This application also has an additional feature which allows the admin to check if a product's stock has dropped below a certain level (ROL), so that he/she can manually place an order before it goes out of stock.

2.2 Design Alternative:

We have made use of linked lists to create and write our databases. It consists of data members like ID, name, unit price, quantities etc.

2.2.1 User Interface Paradigms:

The users would be presented with a login menu, where they can register as inexperienced users or login as existing ones. Both parties can edit their personal details. There are a few differences in the menu options between the supplier and the stockist interface.

The suppliers can add products from a list of items provided by the administrator. They can also delete or edit any items that they have already selected. The stockist are provided with an order items option. They can also place a request with the admin for a product.

2.2.2 Error Detection / Exceptional Handling:

Inexperienced users should register before login, or the program displays an error. Registered users have to login with valid credentials. otherwise, they will get invalid username or password.

Various checks are placed at many menus and sub-menus to check for the required formats. Separate functions are integrated into the program to this end.

2.2.3 Performance:

The system will work on the client terminal. The performance depends on the hardware component of the user’s system.

2.2.4 Maintenance:

Extraordinarily little maintenance should be required for this setup. An initial configuration will be the only system required interaction, by the admin, after the system is put together.

Some file maintenance can be required from the administrator regarding the records of the users. Upgrades of hardware and software should have a negligible effect on this project but may result in downtime.

3.Environment Description:

3.1 Time Zone Support: IST- Kolkata

3.2 Language Support: English

3.3 User Desktop Requirements:

a. 64-bit processor, 1 GHz or faster

b. At least 2 GB free hard drive space

c. At least 1 GB RAM

3.4 Server-Side Requirements:

a. 64-bit processor, 1 GHz or faster

b. At least 1 GB free hard drive space

c. At least 1GB RAM

3.4.1 Deployment Considerations:

a. Easy setup: no session storage daemon, use tmpfs and memory caching to enhance performance.

b. Local storage is used.

c. No network latency to consider.

d. To scale buys a bigger CPU, more memory, larger hard drive, or additional hardware.

3.4.2 Application Server Disk Space:

a. No such disk space is required as the program is fully functional on online IDE(s) as well. The Local Operating System is required and a few files to store the records of processes.

3.4.3 Database Server Disk Space:

a. No such disk space is required as the program is fully functional on online IDE(s) as well. The Local Operating System is required and a few files to store the records of processes.

3.4.4 Integration Requirements:

1. Language: C
2. Tools: Splint, Valgrind, Makefile
3. Complier: gcc
4. Linux Environment

3.4.5 Jobs:

We can establish connections between supplier, stockist and admin who are connected to the system. We can supply the product according to requirements.

3.5 Configuration:

3.5.1: Operating System: Linux environment