Inventory Management system (Review paper)

Khushwant Singh
CSE Dept., UIET, MDU Rohtak, Haryana

Abstract: With the rapid development of network technology, the Web applications are used more widely, and the higher requirements on the Web system has been pursued by many people. This paper describes the Inventory Management System sufficiently to determine the feasibility and usability of a finished system. The core concept is to track the sale of items from the cash registers with additional features for interpreting the data. It uses a client-server model with a connected database to allow multiple stores and warehouses to be connected. This allows for later expansion while still supporting the targeted small businesses. The core features and final framework should be completed within 2 weeks, leaving 5 weeks to implement additional features and testing. According to the practical demands of the Back-state management of the B2C Shopping Mall and Inventory management system.

Keywords- Inventory Management System.

I. INTRODUCTION

With the impulse of advancement for network technique and the rapid development of electronic commerce, online shopping has become the latest consumptive mode for consumers, particular for our younger generations who are more accept new concept and tend to pay more attention to it. In consequence, more and more businessmen establish online malls and make this novel shopping concept accepted by consumers .However, as for these online malls evolved from traditional malls, there are some redundant, even conflict issues existed in online and offline data management and its process. Especially, it is quite important for the adjustments between the former purchase-sell-stock management system and the latter Back state management of the online shops, which has higher requirements for the output of the enterprise workload, the integrity and timeliness of the data. In the past, the client/server architectures are adopted by the purchase-sell-stock management information system, which divides application into two parts. One is the server which is responsible for data management, and the other one is the Client which engages in the interaction with manager. As we known, the client/server architecture is has reflected the superiority, the Inventory management system's upgrade also has been put on the agenda. The improved model and the Back-state management of online mall are very similar, and this will give us opportunities on the integration or mutual expansion between them. The purpose of this paper is to design an online store Back-state management system with the Inventory management system in order to reduce unnecessary data' s duplication for input and output and ensure time consistency.

II. OPERATIONAL CONCEPTS

The purpose of this paper is to construct online mall inventory management integrated system using the concept of purchase sell-stock management system. The Inventory Management System is a real-time inventory database capable of connecting multiple stores. This can be used to track the inventory of a single store, or to manage the distribution of stock between several branches of a larger franchise. However, the system merely records sales and restocking data and provides notification of low stock at any location through email at a specified interval. The goal is to reduce the strain of tracking rather than to handle all store maintenance. Further features may include the ability to generate reports of sales, but again the interpretation is left to the management. In addition, since theft does occasionally occur, the system provides solutions for confirming the store inventory and for correcting stock quantities.

III. SYSTEM REQUIREMENTS

The Inventory Management System uses a web-based interface to display inventory data to the stock manager client. The product will use of open-source software primarily due to cost of implementation. A JSP (Java Server Pages) servlet will be hosted by an Apache Tomcat web server (on top of any choice of operating system, although a flavor of Unix is recommend). The first feature of the Stock Manager Client web interface component allows the Stock Manager Client to view the current stock of products, along with the capabilities of searching and sorting the products. The second feature of the Stock Manager Client web interface will allow the user to modify application settings, such as the threshold for email notifications, frequency of inventory scans (daily at a particular hour, weekly, monthly, etc.), and security settings. The third feature of the Stock Manager Client web interface will allow the user to update the inventory during the restocking process. Since a web interface will be used, a network that supports the HTTP/HTTPS

protocol must exist, whether it is a private network for an isolated customer deployment or an Internet connection for a multi-site customer deployment. The bandwidth of the network depends on the frequency of transactions. A bandwidth of at least 10 Mbps is recommended (small commercial deployment). The database to store the inventory data will use a My SQL database.

Since the software and hardware resources of cash registers are not available due to the variation in software and cost, we will be developing an emulated Cash Register client to interface with actual cash registers. The Cash Register client will emulate purchases by having a simple graphical interface with fields for a barcode number and quantity and a button to make a purchase. As these resources become available, the actual implementation for specific cash register models will be considered for future releases of the Inventory Management System. JavaPOS will be used as a reference for emulating cash registers, and could potentially be used to create the actual implementation of the future release. Alternatively, commercially licensed products are available. In particular, the Microsoft ASP.NET (Active Server Pages) framework can be used. This requires a Microsoft ASP.NET web server and a Microsoft SQL Server database, which also requires a Microsoft operating system. However, these alternative system requirements will increase the overall costs for both implementation and deployment.

IV. System and Software Architecture

The programming language for the Inventory Management System application will be in Java since the JSP/Tomcat architecture will be used. Both types of clients will communicate with the server using Java Remote Method Invocation (RMI) with Secure Socket Layer (SSL). Thus, security will be upheld by HTTPS in both a private network and a network connected to the Internet. The application will use a client-server model. Furthermore, the Java Database Connectivity (JDBC) will use the My SQL Connector/J driver for the server to communicate to the inventory database. Upon receiving requests from the clients, the server will issue transactions to the My SQL database with ACID properties. The Cash Register clients will maintain local log files in the event of communication downtime between Cash Register clients and the server. Upon re-establishing communication, the Cash Register clients will re-synchronize with the server by issuing the requests that occurred during downtime.

Alternatively, if the Microsoft ASP.NET framework is used, then the programming language for the application will be Microsoft C# with a Microsoft SQL Server database. The Microsoft .NET framework includes built-ins for easy database interfacing and .NET Framework emoting (analogous to skeleton and stub), which has the ability to also use HTTPS.
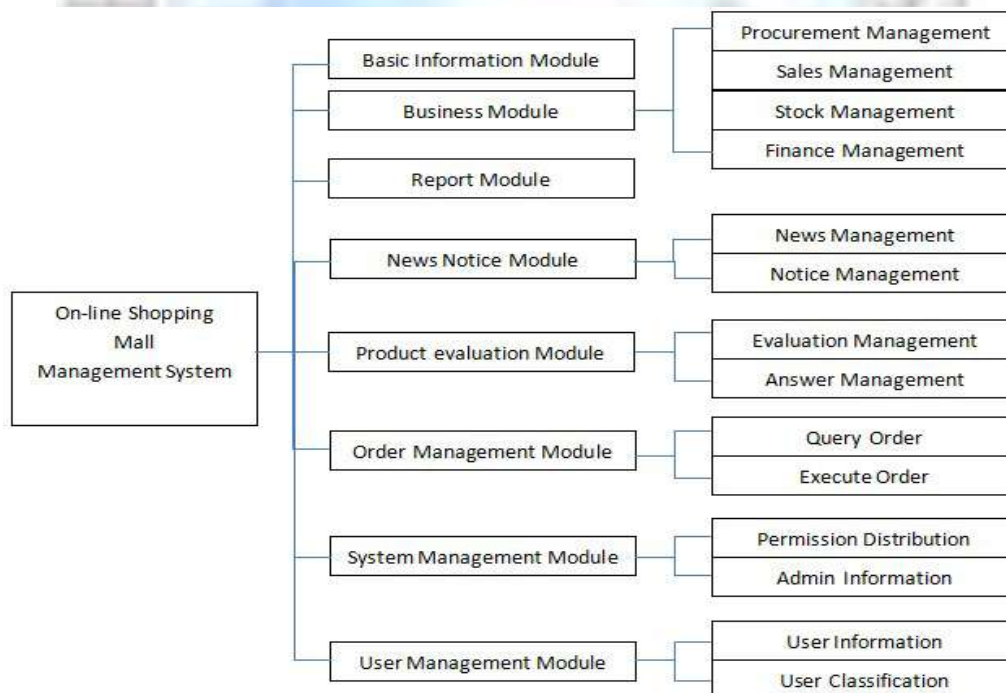
The following portions are the briefly extension to the above, basic information module is mainly responsible for data backup, recovery and browse the system log. The main functions of the financial management are statistical sales revenue, expenditure, the budget, profit distribution and funds occupy, the treatment of providers paid, accounts payable and the capital turnover control are also done in this module. In addition, the financial statistics about online sales, sales calculation within day, month and season are also completed in this module. Sales management is an

indispensable part of the daily operation, which fully embodies the advantages of automation, compatible with all the features of commodity management. There are also some module functions which are describe as follows, inquiries about the mall sales situation; the statistical analysis about sales and gross profit, inquires the commodity quotes, and other special treatment and discount, controlling on the sale of goods and printing the form, the inquiry about the details of the sales, new products on selves and old products off selves, and the management of

Commodity price and other functions

Inventory management is mainly responsible for incoming goods, inventories revision and query, carrying out the inventory reports and so on. The functions of this module are as follows, providing timely inventory goods type, stock, quality, price, storage location and other information. According to the inventory loss, overflow and inventory management, effectively reduce the inventory loss, enhance the effective inventory, support receptionist sales, and obtains reasonable merchandise ordering cycle and quantity. Purchasing management can input, modify, query and analyze in real time, and print a variety of documents according to purchase orders, indent and returned purchase invoice. According to replenishment orders and sales situation to draw up purchasing plan and cooperate with order management department to finish distribution management and return money management.

V. Lifecycle Plan

Objectives:

The main goal of Inventory Management System is to ensure consistent availability of supplies for consumers. Thus, Inventory Management System is directed toward owners of small to large stores and stock managers who are responsible of maintaining sufficient goods on hand in a retail or manufacturing business. It can scale from a single computer running both client and server software up to multiple stores and warehouses.

Schedules:

The time estimated to complete the Inventory Management System project is fairly short. There will be three major Win-Win Spiral Model cycles through our prototyping stage, beta release, and final release. We will need two weeks of designing the architecture and implementing core features and five weeks of adding functionality and testing. The project can be completed by 6-8 people in at most 7 weeks.

Feasibility Rationale Assumptions

The above design should works for the Inventory Management System application. However, we are emulating a cash registers interface to work with our software by simulating barcode inputs. A more realistic design of cash register interface can be done to suit the integration, as the interface does not interfere with the data collected. We expect that the cash registers clients are able to update their data to the current point of sale in case of loss of connection to the server. The particular restocking procedure adopted by each store does interfere with the feasibility and accuracy of the application as it does not remind the stock manager to update the data each time they had done any restocking. We also trust that that the Secure Socket Layer (SSL) is reliable in creating a secure connection between a client and a server.

Risks:

One of the major risks covered by this application is theft breaks synchronization between the inventory and the database. The information could be generated by the data stored in this application. The confidence level of trusting data generated depends on the accuracy of the restocking procedure. Therefore, we are facing a risk of reckless stock manager who could detriment the accuracy of the data. As of the reliability of the SSL encryption, a resolution for this could be by developing SSL and digital certificate policy and configuration guidelines. In addition, giving a choice to the user to set the minimum level of SSL used by not violating the policy should convince them the trustworthiness of the application. Another risk is the competition from other Point of Service software. There are several large competitors in this field including a solution from Microsoft; however, all of these tend to be expensive. Inventory Management System will be a low cost solution mainly targeted at smaller businesses while including the possibility of later expansion.

References

[1]. LI Xiaojun. The design and implementation in Electronic commerce system. BeiJing: Mechanical industry press, 2007.8.

[2]. YANG Shulin, HU Jieping. The latest Java language practical case tutorial. BeiJing: Tsinghua university press, 2006.

[3]. Rob Harrop, Jon Machacek. Professional development guidelines for Spring (Redsaga translation team translation). BeiJing: Mechanical industry press, 2006.8.

[4]. William Crawford, Jonathan Kaplan. J2EE Design Patterns [M]. O'Reilly.2003.9.

[5]. Bruce Eckel. Thinking in Java. 3rd Edition [M]. Prentice Hall,   Inc, 2003.

[6]. Design and Integration in E-Commerce Website based on Framework Technology, XU Jie, WANG Anbao, Published by Atlantis Press.