CMPE 450

SOFTWARE ENGINEERING

SYSTEM DESIGN DOCUMENT

Version 1.0

Project Name:	Stock Follow Up System Project
Submitted to Instructor :	AYŞE BAŞAR BENER
Asistant:	GÜL ÇALIKLI,ONUR GÜNGÖR
Submitted by:	**Project Group 3** **Group** Design Group 3 Database Group 3 Requirements Group 3 Coding Group 3

Boğaziçi University, İstanbul
October 31th, 2006

Contents

List of Figures

1

Goals and Trade-Offs

The primary goals in the development of ”Storage Follow Up System” can be considered as accuracy, functionality, user-friendliness, performance and security.

1.1 Accuracy

When we consider the importance and the variety of data that storage follow up system will use, accuracy becomes a very important feature for the success of this project. The database will include information in many different fields, and the users will have various filtering options while they are observing the database. Thus, to develop a program which provides accurate information, the use of data should be handled carefully and the program must provide a good error handling mechanism.

1.2 Functionality

Currently, a system like the storage follow up system is being used, so one of the main purposes of this new system is developing a more functional system. To achieve this, a more efficient program should be developed, and the information generated by this program should be useful to provide customer satisfaction. Additionally, since the users of this system are assumed to have no additional knowledge and requirements in using the system, all the necessary functions and use cases must be predefined and ready to use in the program.

1.3 User-Friendliness

The system should be user-friendly to increase its functionality and to attract the users. It should also have a simple interface to simplify the usage of the program and to make it more understandable for the users.

1.4 Performance

Since the system is operating on the internet, it must respond to the user in a short time. The processes between the program and the database should be designed and implemented carefully. Additionally, every user connection must be handled as a separate process for the purpose of preventing any error that could occur because of simultaneous usage.

1.5 Security

Security is an important issue in this system, because of the importance of data kept in the database. The authorization mechanism of the system should be well developed, and the database should be well protected against not permitted access attempts.

2

System Decomposition

2.1 System Decomposition

Stock follow up system is consisted up three subsystems. These subsystems are the loginSystem, which serves the login purposes, the userSystem which serves the ordinary users and the adminSystem which is used by admins of the system.

LoginSystem is the entrance of the stock follow up system. Every user must use the loginSystem in order to be identified by the system and use the system. This system is responsible for checking if the user can enter the system, for identfifying the type of the user and for registering new users. It works like an interface between every user and the system.

LoginSystem has two main tasks. First of all, if a user enters userId and password, it checks if the user is registered to the system and if the user is registered then decides if the user is a regular user or an admin. According to the type of the user, LoginSystem calls the userSystem or the adminSystem. If the user is not registered, the user system does not allow entrance to the system. Second task of loginSystem is registering new users to the system. Using the loginSystem, a new user can create a registration, which will be accepted or declined by an admin.

UserSystem is the subsystem used by ordinary users. If the user that enters the system is not an admin, the userSystem is called and used. It allows the usage of ordinary user operations. There are five user operations. First operation is the approval of transactions, which is only allowed to users from purchase department. Second operation is viewing transactions. This operation can only be done by the rector and users from the purchase department. Third operation is insert transaction, which can be used by the users from purchase department and allows entrance of new transactions to the database. Fourth operation is search operation, which allows searching the database according to certain criteria specified by the user. Fifth operation is viewing pending operations, which shows all not-approved transactions in the database.

AdminSystem is the subsystem used by the users who have admin rights. If the user that enters the system had admin rights, the adminSystem is called and used. It allows all operations that an admin can do. There are twelve admin operations. Add manager and delete manager operations allow the adding and deleting of certain users as managers to certain departments. Add goods and delete goods operations allow the adding and deleting of certain good types to the database. There are three listing operations, one for listing the managers, one for listing the goods and one for listing the users. Add department and delete department operations allow adding and deleting new departments to the system. Finally, there are three operations for new user registration, one for checking new registrationg, one for allowing the registration and one for rejecting the registration.

2.1.1 Layers and partitions

There are two layers in the stock follow up system. First layer is the login layer, consisting of the loginSystem. This layer acts like an interface between the users and system. Second layer is the usage layer, which consists of the userSystem and the adminSystem. This layer is reachable only by registered users and allows all system operations, including regular user operations and admin operations.

2.1.2 System topology

Stock follow up sytem can be represented by the following topology:

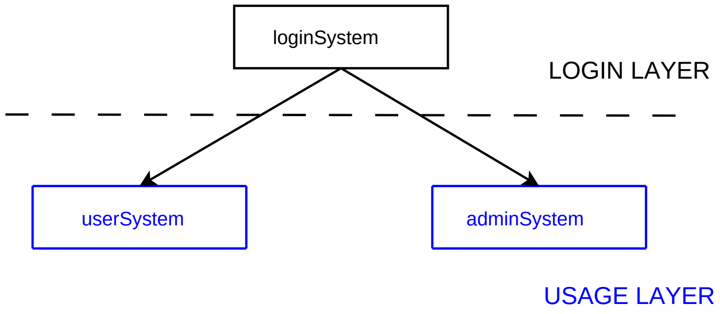

Figure 2.1: TOPOLOGY DIAGRAM

3

Concurrency Identification

3.1 Independency Between Objects in the Object Model

In order to provide the minimum dependency among the subsystems of the StockFollowUpSystem Module, the module is divided into several subsystems. These are loginSystem, userSystem and adminSystem. These subsystems enables the system with the higher level of modularity. Firstly,since many of the operations performed by the subsystems require read-only access to the database and these operations do not bring any dependency between objects, these accesses can performed concurrently.

3.2 Identifiable Threads of Control

The system design of StockFollowUpSystem Module do not require any usage of threads. Threads are not employed in the design since the users access the program from different locations and the operations requested by these users are performed individually and seperately. In addition, since the operations are atomic and do not require any usage of threads, the execution of these processes can be performed without problems.

3.3 User Accesses to the System

The StockFollowUpSystem Module will provide an interface for the users from the internet. Thus, it is inevitable that the program will be inherently multiuser. First, the users trying to register to the system will be presented a login web page prepared and managed by UserInterface module. During login procedure, the password and the username of the user will be compared to the relevant data on the database. Since this process requires read-only access, there will be no problem encountered while the users from different locations are trying to access the system. Thus, this process can be performed simultaneously.

Secondly, after the registration procedure is completed, according to the type of the user (admin, employee, manager etc.), the appropriate access rights will be given to the user while creating the session. According to the access rights of the user, the operations that the user has the right to perform will be active on the page prepared by the UserInterface module. As the user attempts to perform an operation using the system, while finalizing the process, it will possible for the data to be written on the database which requires a read-write access to the database. In that case the necessary fields of the database will be blocked and simultaneous access from different users will be denied.

On the other hand, if the operation, that is intended to be performed by the user , requires read-only access to the database, since this operation does not bring any dependency, it can be performed concurrently without causing any inconsistencies.

3.4 Availability of Multiple Queries in a Single Query

The querying option is available to the authorized users of the Stock Follow Up System. They can construct queries by entering the criterions into the fields on the interface. These fields are sellerOfGood, approvalStatus, typeOfGood, payingDept, usingDept, dateOfOrder, dateOfBuying, dateOfApproval, lowerPrice, upperPrice. The user does not have to know a SQL-like language, by entering the criterions on the appropriate fields, the query is generated and executed. If there is entered no criterion on a field, this field is ignored and not included in the query. Therefore, this feature of the system enables the user create single or multiple queries.

3.5 Parallel Handling of Queries by Different Subsystems

Since the subsystems are mostly independent of each other, queries can be handled in parallel between them. The only exception to this is the situation in which one of the subsystems tries to have a read-write access to one or more of the database tables while another one is trying to access. This situation will be handled by providing a locking mechanism on the table fields during updates, in order to ensure the consistency of the database.

3.6 Concurrency Scheme

The operations that are concurrent between the subsystems can be performed by the StockFollowUpSystem module since these subsystems are independent of each other except in the case of the read-write access to one or more of the tables in the database. In case of updating the tables, the necessary fields of the table will be blocked and concurrent access will be denied. In all other cases, subsystems will operate concurrently without causing any inconsistencies in the database.

4

Hardware / software Allocation

This project will have an Internet based platform for both the implementation and the functional access. As it has been mentioned earlier during the requirements analysis phase, the system will be dealing with three sort of users namely internal employees and managers and the administrator. All of these groups will access the system via Internet pages in different manners.

The hardware allocation of the above described system consists of two main aspects, a closely related couple composed of client and server. The project will initially need two servers to handle the web page operations and database storage. It will be extremely important to separate these two tasks into two different servers in the case of a remarkable increase in user numbers. Doing this, the Web Server will handle the Internet connections and the Data Server will deal with the database operations. Besides the server aspect, the client part of the system will only be asked to have a PC with efficient Internet connection via modem, LAN or other protocols and the MS IE software installed on their machine. An office environment in traditional company idea is not necessary.

The design platform will be based on SQL server to handle secure multi-connection and Java programming to deal with excessive Internet based form operations.

4.1 System Performance

The system performance is based on some important measures: information retrieval speed, connection handling performance, data processing capability and memory usage efficiency.

4.1.1 General System Performance

The information retrieval should be as fast as possible for customer satisfaction. To ensure an efficient response time, the customer should be able to download a page in 5 seconds with a 33.6 Kbps modem. The form submission operations are required to be processed in a short time.

Since the project is an online system, it will have multiple-users to access it. This multiple-user idea will definitely result in a heavy transaction and request traffic. The server should be able to serve 25 percent of registered customers simultaneously. Separation of connection handling and data processing over two different servers would definitely improve the overall system performance as well as the cost incurred.

4.1.2 Input / Output Performance

The project will basically have web forms and web pages as inputs and outputs. The system will not need special hardware for input/output purposes and the input/output performance will be again dependent on the overall system performance mentioned in the general performance measures part, so the response time, the information retrieval speed and the server abilities will play important roles to determine the input/output performance.

Slow data retrieval because of poor programming must absolutely be discarded since the software runs on an online platform.

4.1.3 Processor Allocation

The system is not supposed to handle heavy arithmetic operations or long computations. It will be dealing with simple database queries at most, so it is not crucial for the system to manage a multi-processor environment up until to a certain noticeable database size.

4.1.4 Memory Allocation

The larger the primary memory, the faster the applications would run. As the system performance is basically related to the information retrieval speed and the response time, it would be preferable to have the servers with primary memories in the order of some GB's. Moreover, the size of the secondary storage should be sufficient for data swapping, recovery and backup procedures. An acceptable range for secondary storage should at least cover some tens of GB's.

4.2 Connectivity and Network Architecture

The project has the feature to be online and global over the Internet and web pages as mentioned before. The only physical component of theproject is a couple of connected servers managing the performance of the web page operations and the data storage issues. Admin, managers and staff will all access the servers via TCP-IP.

5

Data Management

The Stock Follow System is where the information about every transactions related to each department in the Bogazii University is held. One of the main aim of the system is to manage all information about each transactions efficiently. In order to achieve that, Stock Follow System uses databases.

5.1 Necessity of Database

Database will be used to keep information about requested, rejected, waiting, bought stuff and registered , waiting users and their access permission. Each of these will be stored in a table. That will allow all system user access necessary information according to their permission.

5.2 Data Distribution

The data included by the GUI interface will be stored into the seperated tables of the grand database.

5.3 Extensibility of the Database

During the improvement of the system, there could be some necessity changes on the database attributes such as creating new tables, adding different fields or changing data types of some fields.

5.4 Frequency of Database Access

Database is accessed whenever there is a user registration, user login, user approval, item request, item rejection, item approval and transactions visualization.

5.5 Query Format and Interface

The query format of the database is SQL stored procedures but this format is strictly hidden from the user with the help of user friendly interfaces.

5.6 Usage of Hidden Location

The server will be located at Purchase Department but the users won't have direct access permission to the server so we do not need a hidden location. The only acces way to the database will be our Stock Follow up System and it provides abstraction and restricted access. In the case of network or system failure it will be directly accessed by the administrator.

5.7 Usage of Relational Database

The database is selected to be relational since it is based on relational algebra. Data is presented as two-dimensional tables. Tables have a specific number of columns and arbitrary numbers of rows to dynamically keep records.

Relational database allows uniquely identifying a row in a table with primary keys and reference to another primary key in another table (foreign key). Also SQL will be used since it is the standard language defining and manipulating tables in relational databases.

5.8 Selection of Archival Data

usertypeID and udescription are stored in the USER TYPE table.

authorityID and adescription are stored in the OPERATIONS table.

usertypeID and authorityID are stored in the USER TYPE OPERATIONS table.

username, usertypeID, departmentID, name, surname, address, phone, email, password, statuID are archived in USERS table.

orderID, itemID, price, amount, orderdate, approvaldate, purchasedate, purchasedepID, seller, total, orderdepID, statuID are archived in the TRANSACTIONS table after a transaction request has sent.

itemID, itemname, status, description are stored in the ALLOWED ITEMS table.

departmentID, departmentname, status are stored in the ALLOWED DEPARTMENTS table.

6

Global Resource Handling

The project will use limited resources. So, there will be some constraints on the usage of global resources. The usage of global resources includes memory management, database management and authentication and security issues. The following are the details about these resources and their usage.

6.1 Memory Management

The system will provide service to many people from several departments. These people may produce huge amounts of data. The primary and secondary storage used for the system should be large enough to handle the transaction data. The system may be designed and implemented in such a way that the space complexity is decreased. Usage of an additional server may be a solution too.

6.2 Database Management

The database will include tables for the users (administrator, managers, employees, and rector), goods and allowable input values. Some of the users have authentication to make changes on the database. The employees will store new data about transactions and register to the system. The administrator will approve the registrations and modify the allowable input values. The users of the system will be able to view the data according to their authority. A back up operation is crucial for the management of the database. The database should be backed up after the last transaction of a day.

6.3 Authentication and Security Issues

It is very important for the security of the system that a non-user of the system should not login the system, view or modify the stored data. The registered users can only login the system by providing their usernames and passwords. There are also restrictions on the registered users. There are four types of users and each type of users can access different parts of the system and perform different tasks. For example, managers who are not from the Purchasing Department should not be able to have any access to the transactions that do not involve their department in any way.

7

Software Control Implementation

7.1 External Control Flow

Control Flow of the Stock Follow-Up System is defined like a simple web application. Although there may be several users using the system at the same time, the control flow of any single user has a pre-defined structure. After the simple login page at the beginning of the system run, the system is composed of a bunch of branched web pages.

7.2 Concurrent Control

Although the application is web based and can have multiple users at the same time (concurrent runs) a single user can visit only a single page at the same time. This is a must due to the fact that data integrity of the stock follow-up system must be maintained.

7.3 Internal Control

The stock follow-up system has a tree shaped internal control using the user request and web page showing structure. The run of a single user may bring up different pages and different methods might be called from the different classes of the system.

7.4 User Interface

User interface is created by web pages which are specialized according to different types of the user. The system design for the user interface imposes different user interfaces for example for an administrator or a department manager as they have different rights. Therefore; an option of adding an allowed good value should not be shown to an ordinary department worker as he does not have the right to take such an action.

8

Boundary Conditions

8.1 Initialization

8.1.1 Dynamic Model of the System Startup

The Stock Flow up System is an event-driven program working on the web server without explicit startup and termination. However, there are two points in time when startup is necessary. First is the installation time and the second is the startup after backup operation.

For the system to start working properly, the program should be installed on the Web server and the database should be installed on the Data Server. Backup process will restart the database so program should response as "denial of service" during this process and therefore there is no need for program to start up. During the installation same approach must be taken. First the database will be launched and then the program.

8.1.2 Description of Data Accessed at Startup

Since the system has operations related with the time and date, these should be gathered at the startup. This is necessary for the transactions table.

Also users table must be available at the start up in case of a new login request. This is particularly important for the robustness and accuracy of the program.

9 Design Rationale

The Stock Follow up System is designed to be simple and user-friendly. As the previous system that is being used is considered, it can be easily seen that our customers do not have any experience of using a complex system, therefore, anything, which can be accessed by user, should be simple and easy to understand. In design process of the system, user-interfaces and object design are structured on this purpose.

As for the user-interfaces, each type of user has a different interface, because a user does not have to deal with many functions or links which are not related to his job. The User-interfaces are designed in a simple way. HTML is preferred as coding language since it is easy to learn and clear to everyone. While being simple, the user interface did not turn into something boring. Its design is enriched by using flash applications in the menu. Another reason for using HTML, it is also compatible with Java Server Pages so that the coder and GUI groups will not have any integration problem. The coder group is using Java Server Pages (JSP) because JSP allows software developer to generate HTML documents dynamically in response to a Web client request. JSP also allows embedding certain pre-defined actions into static content. Therefore, JSP language is more appealing to use with HTML.

As for the object design, classes and their methods are planned to according to different users and their authorization rights. To generate different user-interfaces during login process, user rights are created based on user and his department. It would be easier to code if the user had chance to see all operations that could be done in the system, and to try by selecting operations what he had right to do in the system. However, designing such a complex user-interface would confuse the user. We cannot provide a training program for users for a long time. Therefore the system is designed to create different user-interfaces basing on their rights in sake of being simple and understandable.

The Stock Follow up System can be discussed in terms of scalability, extensibility and modifiability of the design.

As for the scalability, the system can be easily extended and can be used in different departments. The Stock Follow up System will be used by purchasing department, managers (rector and administer) and others. The definition of ”other users” can be extended if a new department would be added to the system.

In terms of networking, the system will not allow additional communications links such as wireless communication because there is no need to access the data from outside the department building. The data should not be accessed from outside as it stores the accounting information of the university.

As for the extensibility, the system can be extended by adding new databases if it is necessary. However, the data that they are storing would not cover much space. In fact, the system data will be backed up periodically. The old data can be kept in secondary storages. There is no need for a second database but if it is discussed in terms of extensibility, the system would allow adding a new database.

As for the modifiability of the design, software does not disrupt the stability of hardware platform. The system does not require any complex calculations which would exhaust operating system. However, the users may log into the system and require data at the same time which may lock the system due to the extension of number of operations which can be done at the same time. However, our customer wants only twenty-five users to log on at a time at most, which will not cause such a capability problem. In future, the system would pursue to satisfy the customer will as a dramatic increase in number of workers is not expected.

The Stock Follow up System is designed with new technology. It is being written with JSP and can handle with simultaneous user connections, database operations and work stations in many departments. As it is mentioned before in detail, the system can be modified and extended in the future easily if it is necessary.

Index