

CSE 201 Advanced Programming (Monsoon Semester)
Course Project Option-2
Superstore Management System

IMPORTANT Instructions:

1. It's mandatory that you attend all the deadlines in this project as per the schedule. No request for rescheduling the demo will be entertained. In case of any unavoidable circumstances you have to take email approval from me well in advance.
2. Each group should choose their project latest by 23:59pm on 08/10. Absolutely no changes are allowed after this deadline. Either of the two members in any group should ensure they choose the project in case another group member is unavailable until this deadline.
3. You **MUST** have a **PRIVATE** git repository for your project and every group member should very frequently check in their code in this repository.
4. No extensions will ever be provided. Any submission after the deadline will not be evaluated. If you see any ambiguity or inconsistency in a question, please seek a clarification from the teaching staff.

Plagiarism: All submitted deliverables are expected to be the result of your individual effort. You should never misrepresent someone else's work as your own. In case any plagiarism case is detected, it will be dealt as per new plagiarism policy of IIITD that was also discussed in the lecture.

The project relates to design and development of a flexible and intelligent management system for a Superstore. The Superstore has multiple warehouses and a chain of stores located in different cities and towns. One or more stores are linked to a warehouse. The linked warehouse ships items to stores on requirement basis as stores have limited space to stock items. The warehouses in turn fulfil the order of stores from their inventoried items or can seek help from another warehouse (i.e. initiates new order to other warehouses, the selection of the other warehouses should be optimal and this optimization algorithm is left for you to decide). Additionally, a warehouse also keeps track of its inventoried items. Both warehouses and stores are identified by id and name.

A. The operational requirements of the system are:

1. A warehouse provides items of various categories viz. grocery, apparels etc. Each category is further divided into subcategories e.g. grocery has the subcategories like beverages, pulses etc. The subcategories contain items. The system should provide GUI to add items in existing subcategories as well as creating new categories and subcategories. Similarly, a store also keeps items based on categories and subcategories. But a store may contain subset of items in linked warehouse.
2. The system should allow updating and deleting the inventory of any items as well as that of categories and subcategories in warehouse as well as in store. Update includes increasing/decreasing number of unites/items i.e. inventory inwards/outwards, inventory details such as description, cost etc.
3. There is a cost associated with inventory of items. The cost includes fixed cost per quarter D and carrying cost per unit per quarter H . Also, the demand of items in units per quarter K .
4. The quantities in #3 above enable to compute economic order quantity (EOQ) i.e. the number of units superstore should be adding to inventory order. The formula to calculate EOQ is,

$$EOQ = \sqrt{\frac{2 * D * K}{H}}$$

5. When the current inventory level of any item reaches zero, the system should generate an alert of ordering EOQ number of items automatically.

B. The system has four different types of users:

1. **Super User:** The super user creates new warehouse and stores, link stores to warehouse, create warehouse administrators. Super user can see data of any warehouse and any store.
2. **Warehouse Administrators:** A warehouse administrator is allowed to manage a warehouse. Management here means addition, updation and deletion of categories, subcategories and items. The administrator is also allowed to manage all the costs, demand of items, usages of each item as mentioned in point#3 and 4 in A above and also handles received orders.
3. **Store Administrators:** A store administrator is allowed to manage categories and subcategories of items in the store. Management here means addition, updation and deletion. The administrator is also allowed to manage all the costs, demand of items, usages of each item as mentioned in point#3 and 4 in A above.
4. **End Users:** The end users of the Superstore system are buyers. They can browse through various categories and subcategories at a store and check the current availability of items. The end users should be provided an option to search any item, sort items alphabetically. The search should support partial matching of search criteria with item names. The end users login into the system as guest users.

C. Some other Salient points are:

1. All the users can also access the system remotely. All administrators have their respective login id and passwords. The login id and password for administrators is created by the super user. The login id and password for super user is created by the application or the system. The end users login to the system as guest users.
2. Whenever a warehouse administrator logs into the system, it automatically shows alert message to him/her for items that needs to be ordered. The alert message should be shown in a new pop window.
3. Whenever the quantity of an item at store reaches zero e.g. when a buyer buys last number of the item, the store sends a message for the new order (with EOQ level as in point#4 in A above) to the linked warehouse. The linked warehouse decrement the number of available items by EOQ amount and sends back an ack to the store stating that the order request has been executed and the quantity of the item sent. The store then increment the items by the quantity received.
4. A warehouse administrator can check data of any other warehouse. A store administrator can only check data of their respective stores.
5. The store administrators should be able to send a message to linked warehouse for required items. The message should contain a list of records where a record is for one item and contains item name, item code (unique code), quantity, expected arrival date.
6. The warehouse administrator should be able to forward a message received from store administrator to other warehouses (the selection as explained in second paragraph in the beginning) if the item is not available in his warehouse.
7. Generate very detailed documentation for each and every class/method/fields of your project by using the JavaDoc tool. You can get information on what/how to use JavaDoc on google.

D. System requirements:

- (i) Different GUIs for different types of users i.e. the main window of Superstore management system should give any option for use type. Also, different views should display data to users based on their access privileges.
- (ii) An end user should be allowed to select the store and check data of any store. Data in the context of end users mean category, subcategory and name of items. Whenever, any user (administrators or end users) selects a category, the subcategories should appear in new window. Similarly, if they select a subcategory, the items should appear in a new window. An option to sort items should be available in all these windows.
- (iii) The application should allow to open multiple client windows simultaneously on one machine.
- (iv) When the SuperStore application starts for the first time, it reads data for warehouses and stores from a configuration file.
- (v) When the super user logs into the system for the first time, the system displays all the registered warehouses and stores information in a new window.

Bonus:

Although we have specified the basic requirements, if you are able to come up with some more interesting features then you would be “eligible” for bonus marks. Although this eligibility we will decide based on factors such as how many other group has also come up with same additional functionality.

Project Deliverables:

1. Deadline 1 (Due on 17th October) – Submit detailed UML class diagrams and use case diagrams for your project. Submit pictures of your diagrams on backpack. We will use these pictures during demo of this deadline.
2. Deadline 2 (Due on 02nd November) – Show static GUI of your project and also some animations components. Submit this code on backpack just like lab deadlines.
3. Deadline 3 (Due on 17th November) – Submit complete project on backpack. Demo after end semester exams.