STRUCTURED ANALYSIS AND STRUCTURED DESIGN

On-Line Sales Portal Software

Data Dictionary

User name: string

User address: string

User email-id: string

User IM ID (Optional for Customers): string

User Gender: string

Manager Biometric ID: string

Manager DOB: string

Item Name: string

Item Price: double

Item Company: string

Item Photo: (.jpg, .png)

Item Description: string

Item City: string

Amount Paid: float

Structured Analysis of the Software

Initially the user can register as a Customer or as a Manager
For seeing the List of Items along with the category Login is not required
but in order to make a request for buying Login is necessary.

User can now Login as a Customer or as Manager, for this he had to provide UserId and Password
This userid and password will be validated from the database by validateCustomer() and validateManager() functions
which return a bool value.

If the returned value from the above two function is "true" then the account exists hence a Customer/Manager
Object will be created accordingly.
If the returned value is "false" then no object will be created.

Database is a class whose only one object will exist which would be made when the software opens.
Any change in the Database will be made in this single object and it will be updates in the file when software closes (i.e. in case of online when the window tab closes then file will be updated centrally).

As soon as Customer Object is created then database object will be passed then listItem variable field will be updated in the constructor.
Customer can edit (add/remove) his added items in the database, these functions will be performed in the database file when the customer object is being destroyed.
Hence deleteItem() and addItem() won't use Database object.

Now,

1. When Customer wants to see his own items

He Obviously cannot buy his own item hence purchase option won't be called.
he can view his items uploaded which will call showItem() function, in which he can do following operations

a).He can delete the item hence deleteItem() will be called

b).He can press the back button hence showListItem() will be called

c).He can see the bid price which is set by any interested buyer, it's up to seller whether he wants to lower the price pf item or not

If he agrees then he will update the current price of item, then setprice() will be called and he can change the status of item in setprice() function only

Suppose If he don't want to negotiate with the price of the item then he can set status of the item to "NotNegotiable" In such cases

Buyer wont be able to bid on the price of the item he can only purchase the item.

2. When Customer wants to see searched Item then search() function in database will be called and it will return the list of items matched

now call showitemList(), there user will click on specific item

this item id will be returned and now showItem function will be called which will again return an int value stating {back,buy}

now if return value corresponds to buy then system will check whether status is Negotiable or not if yes then buyer can set bid price,

if not then he cannot set the bid price and he will have to either buy that item or go back in the previous menu.

When Manager Object is created (on the case of successful Login) he can send a request to the database Object to

1.remove Customer

2.add/remove categories

3.change categories.

4.review/remove item.

Manager can also send email to a Customer by making an object of class Email, by this he can send any email

to any of his Customer email.

Manager can also see the list of purchased items by calling the function of PurchaseBill.
He can also delete any purchased item in the history through removePurchase() function in the Database.

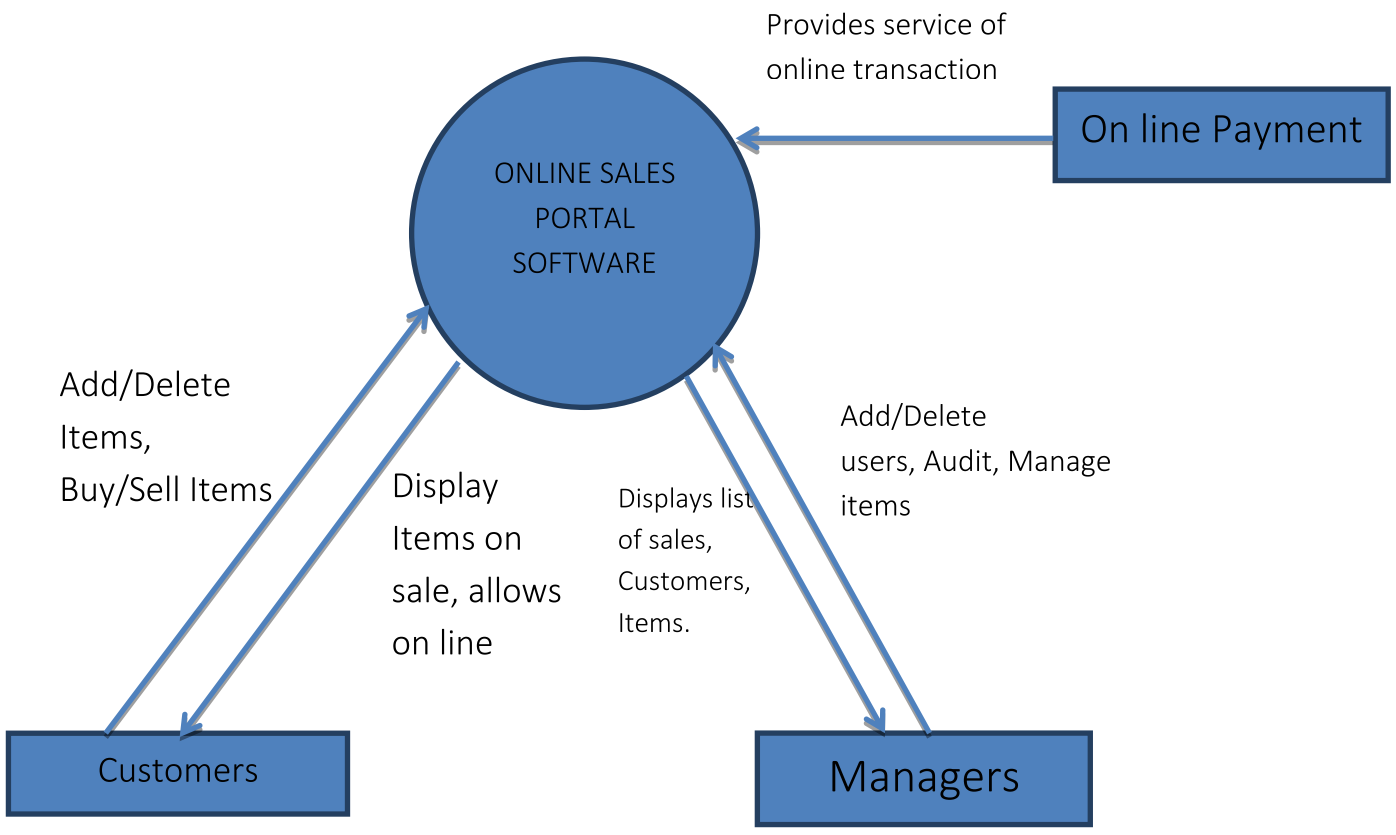
In addition to above all functions the Customer and Manager can delete their accounts respectively through member functions deleteAccount()
which will send request to removeCustomer()/removeManager() function in Database class and hence Database function will be updated.

Item is a class whose object is created for each and every object listed in the database.
Each item will have a price which is fixed by the seller
Each item will have a bid price which will be decided by the buyer.
The buyer will set the bid price and corresponding emaill notification will be send to the seller
Now, its upto seller whether he want to negotiate with the price of the item or not
Accordingly he can status of item as negotiable and non-negotiable.
If seller agrees with the bid price of the item then he can reduce the cost of the item for that buyer, and a corresponding email notification will be send to the buyer on his email address.

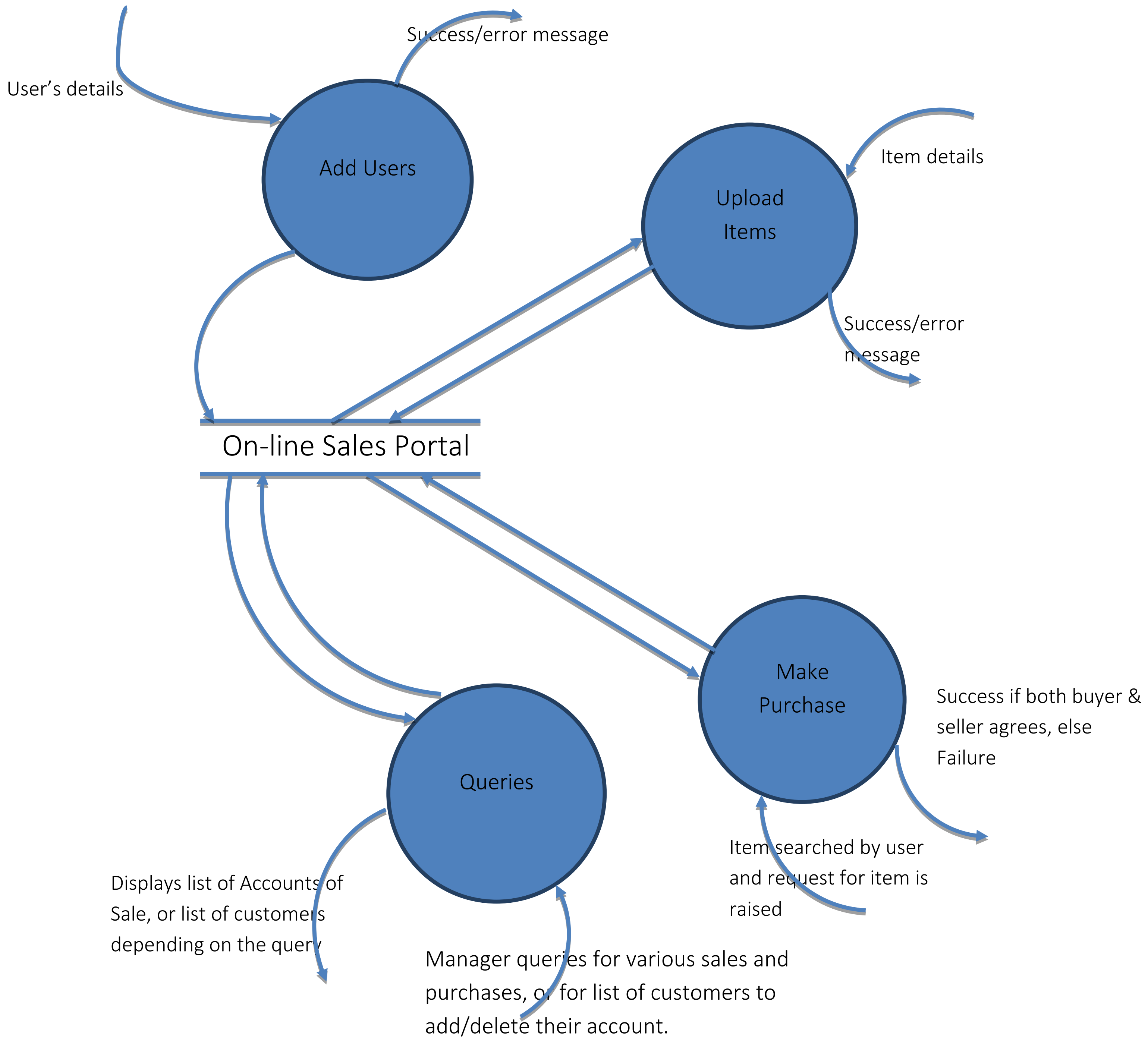
When Buyer and seller both agrees on some price then buyer can pay the money through
Pay class which will have function that will open a secure gateway for the transaction of money,
Corresponding notification of payment will be sent to the seller so that he can arrange for
the delivery of item.

Each item will store separately its buyer and seller respectively.

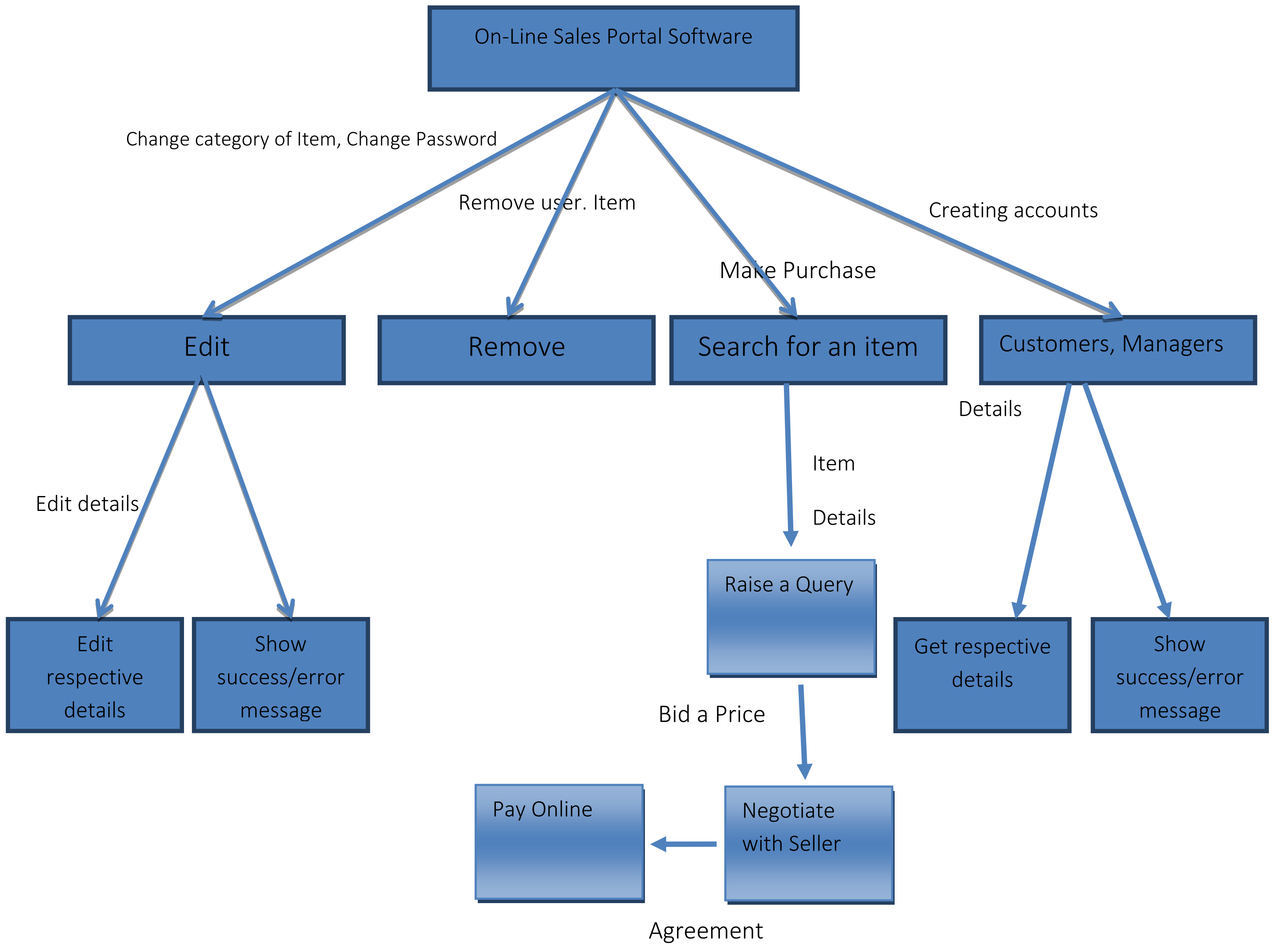
CONTEXT DIAGRAM


Provides service of online transaction
On line Payment
ONLINE SALES PORTAL SOFTWARE
Add/Delete Items, Buy/Sell Items
Display Items on sale, allows on line
Displays list of sales, Customers, Items.
Add/Delete users, Audit, Manage items
Customers
Managers

DATA FLOW DIAGRAM


Success/error message
User's details
Add Users
Upload Items
Item details
Success/error message
On-line Sales Portal
Make Purchase
Success if both buyer & seller agrees, else Failure
Queries
Item searched by user and request for item is raised
Displays list of Accounts of Sale, or list of customers depending on the query
Manager queries for various sales and purchases, or for list of customers to add/delete their account.

STRUCTURE CHART


On-Line Sales Portal Software
Change category of Item, Change Password
Remove user. Item
Make Purchase
Creating accounts
Edit
Remove
Search for an item
Customers, Managers
Details
Item
Details
Edit details
Edit respective details
Show success/error message
Raise a Query
Get respective details
Show success/error message
Bid a Price
Pay Online
Negotiate with Seller
Agreement