CS 558: Computer Systems Lab

Assignment – 2: Socket Programming

Client Server Trading System using socket programming

SUBMITTED BY -

GROUP 17

214101050 - Senapathi Akhila Preethi

214101051 - Shambhavi Shanker

214101052 - Sherin Abraham

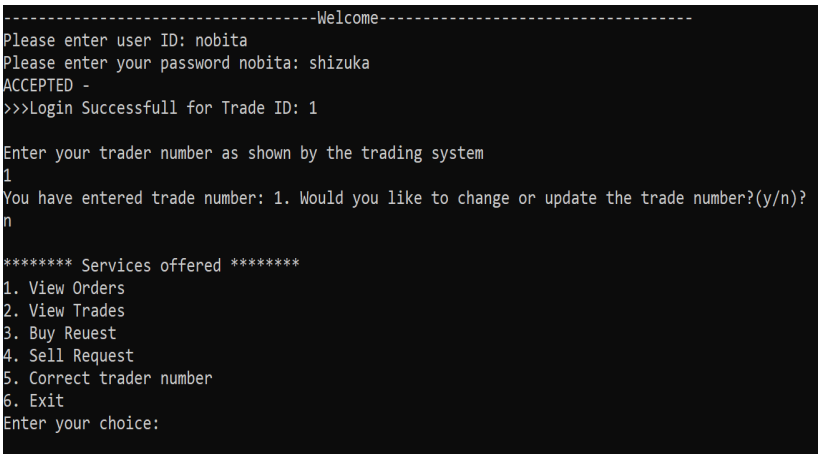
INTRODUCTION

- This is a client server trading system application using socket programming.
- This application facilitates a set of traders to trade with each other by buying or selling items in an automated system.

Functionalities that a client can perform include:

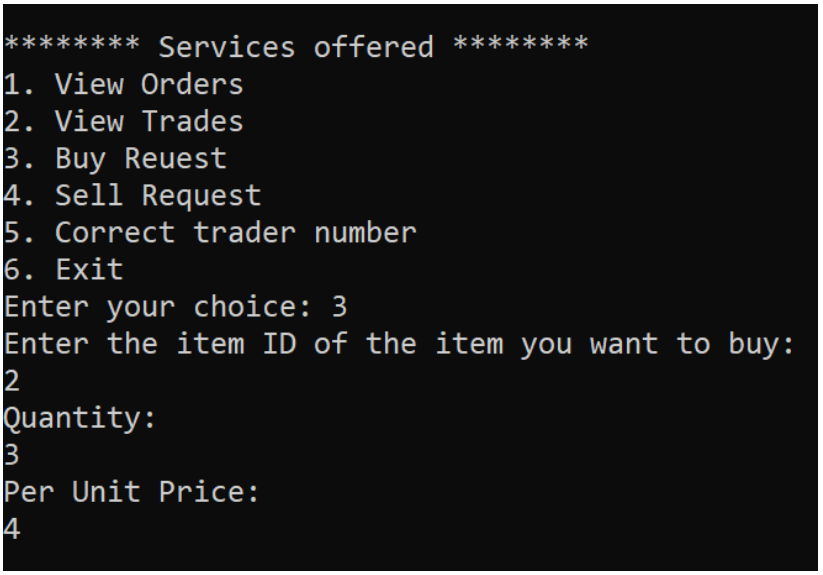
1. Login to the system
2. Send buy request
3. Send sell request
4. View order status
5. View trade status

Index	Details
Functionality	Login to the System
Overview	The trader will execute the client, give the trader number and will be logged in. After that he/she will have the following options in a menu. Several clients will login (from different terminals).
Description	➔ After the client connects to the server, the client is asked for username and password . ➔ A message is sent to the server of the form : **<username> <password> L $** ➔ Where username and password are the one entered by the user. ➔ With the L in the message, the server gets to know that it's a login request. ➔ Then the entered username and passwords are matched by the server with the five usernames and passwords stored in credentials.txt file. ➔ First the contents of credentials.txt is read and put in an array of strings of size 5. ➔ Then the usernames and passwords are matched and if it matches with any of the five credentials then the user is allowed to login otherwise not. ➔ After logging in, the user is provided with a user id or trader id.

	➔ The user is then asked to enter the trader id and if entered correctly, he/she is allowed to trade and is shown a list of options to perform. ➔ If a user mistakenly enters a wrong trader id then he/she can re-enter the correct trader id by selecting option 5.
Output	

Index	Details
Functionality	Send Buy Request
Overview	The trader will send a buy request by stating the item code, the quantity and unit price.

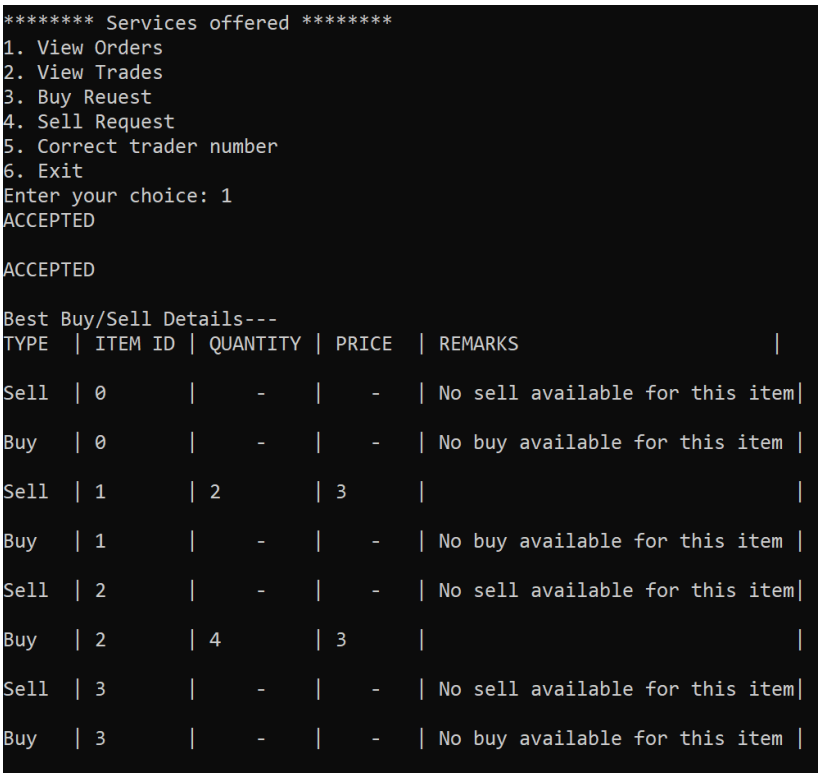
Description	➔ For buying an item, the trader has to enter 3 after the menu is shown. ➔ Then the user is asked for the item number, its quantity and unit price. ➔ A message of the following form is sent to the server <trader_id> <password> B <item no.> <quantity> <unit price> ➔ With B in the message, the server gets to know that the request is for buying ➔ The server maintains a buy queue and a sell queue for each item. A maximum of 1000 requests can be put in each queue and the queue is a circular queue. ➔ We create a buy request with the details provided by the trader and then we check if there is a matching sell request in the sell queue. ➔ We scan the sell queue for that item and find the request with the lowest price. ➔ Since the sell queue is sorted in increasing order of price the first request will be the one with the lowest price. ➔ If the price is less than or equal to the price which the trader is willing to give, a trade is created for that item and then the quantity of that item is updated in the sell queue. ➔ If the sell request is completely bought, The entry for that sell request is removed from the queue . ➔ If the required quantity of items is not bought, we look for other sell requests and repeat the same process until all the requests are seen. ➔ If still some quantity could not be bought then the remaining is put in the buy queue in increasing order of prices. If more than one request have the same price then requests are put in FCFS order.

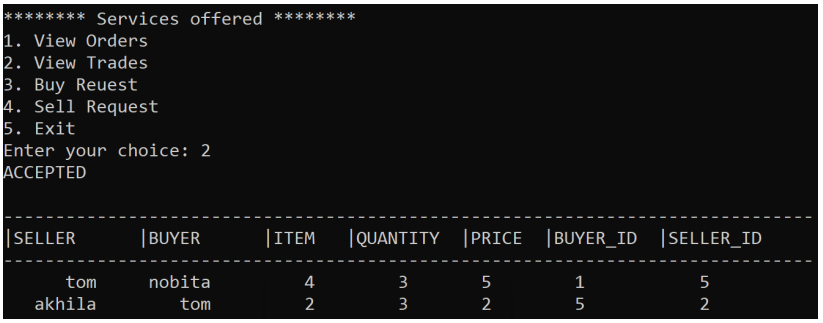
Output	

Index	Details
Functionality	Send Sell request
Overview	The trader will send a sell request by stating the item code, the quantity and unit price.
Description	➔ For selling an item, the trader has to enter 4 after the menu is shown. ➔ Then the user is asked for the item number, its quantity and unit price. ➔ A message of the following form is sent to the server <trader_id> <password> S <item no.> <quantity> <unit price> ➔ With S in the message, the server gets to know that the request is for buying ➔ The server maintains a buy queue and a sell queue for each item. A maximum of 1000 requests can be put in each queue and the queue is a circular queue. ➔ We create a sell request with the details provided by the trader and then we check if there is a matching sell request in the sell queue.

	➔ We scan the buy queue for that item and find the request with the maximum price. ➔ Since the sell queue is sorted in increasing order of price the last request will be the one with the maximum price. ➔ If the price is more than or equal to the price at which the trader is willing to sell, a trade is created for that item and then the quantity of that item is updated in the buy queue. ➔ If the quantity of buy request is less than the quantity for sale , The entry for that buy request is removed from the queue . ➔ If the required quantity of items is not sold, we look for other buy requests offering more price than what the seller is willing to sell at and repeat the same process until all the requests are seen. ➔ If still some quantity could not be sold then the remaining is put in the sell queue in increasing order of prices. If more than one request has the same price then requests are put in FCFS order.
Output	******** Services offered ******** 1. View Orders 2. View Trades 3. Buy Reuest 4. Sell Request 5. Correct trader number 6. Exit Enter your choice: 4 Enter the item ID of the item you want to sell: 1 Quantity: 2 Per Unit Price: 3

Index	Details
Functionality	View Order Status
Overview	The Trader can view the position of buy and sell orders in the system. This will display the current best sell (least price) and the best buy (max price) for each item and their quantities.

Description	➔ The Trader can view the position of buy and sell orders in the system. This will display the current best sell (least price) and the best buy (max price) for each item and their quantities. ➔ To view order status, the trader has to enter 1 when the menu is shown. ➔ A message is sent to the server of the form : <username> <password> VO $ ➔ Here VO tells the server that the request is for viewing orders. ➔ For each item the best buy and best sell is printed. ➔ For the best buy of an item , the buy request with the highest price is shown along with its quantity. ➔ For the best sell of an item, the sell request with the minimum price is shown along with its quantity. ➔ For some items, there may not be any buy or sell requests.
Output	

Index	Details
Functionality	View Trade Status
Overview	The trader can view his/her matched trades. This will provide the trader with the details of what orders were matched, their quantities, prices and counterparty code.
Description	➔ The trader can view his/her matched trades. This will provide the trader with the details of what orders were matched, their quantities, prices and counterparty code. ➔ To view trade status, the trader has to enter 2 when the menu is shown. ➔ A message is sent to the server of the form : ➔ <username> <password> VT $ ➔ Here VT tells the server that the request is for viewing orders. ➔ The server then checks the tradeDetails array which stores all the trades. ➔ If in any trade, the buyer or seller is the current trader then the trade details for that trade are printed.
Output	

```
******** Services offered ********
1. View Orders
2. View Trades
3. Buy Reuest
4. Sell Request
5. Exit
Enter your choice: 2
ACCEPTED

-------------------------------------------------------------------------------
|SELLER      |BUYER      |ITEM   |QUANTITY  |PRICE  |BUYER_ID  |SELLER_ID
-------------------------------------------------------------------------------
      tom     nobita         4        3       5        1           5
   akhila        tom         2        3       2        5           2
```