

Database Management Project Report

[TITLE: Portfolio Management System](#)

Link to the Project source code :

<https://github.com/NitishDixit/CS5200Project.git>

Team Members:

Nitish Dixit

Janki Bhimani

Scope of Project:

Nowadays, share market is on the top of the list for investing money, be it a small or a large investor, so there is a lot of money coming in and out in the form of stocks.

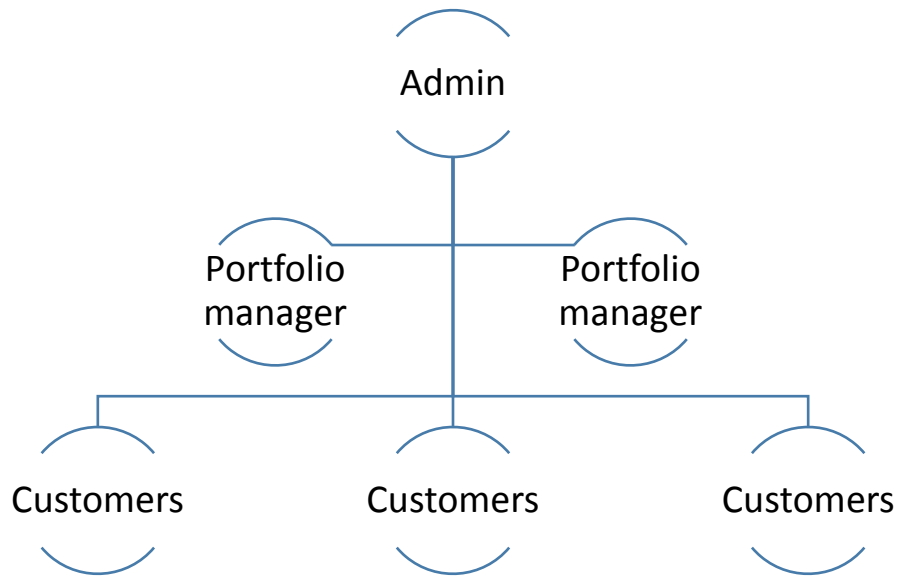
Now, due to such a great demand there is a need for a proper portfolio management for such a system which tracks and also guides you in the matter of stocks is necessary.

Components:

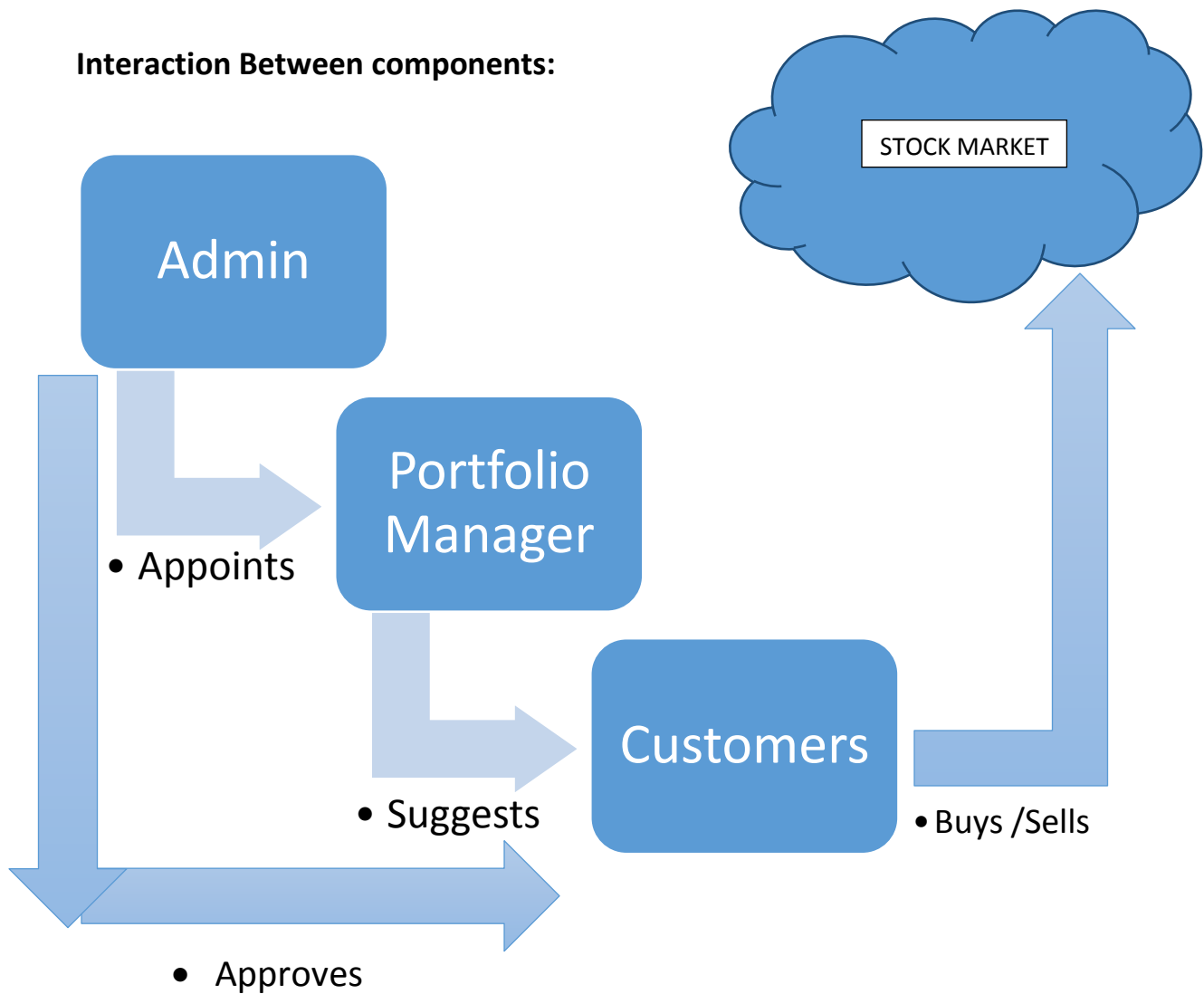
The portfolio management system consists of 3 components as follows:

- Customers
- Portfolio Manager
- Administrator

Hierarchy of the components:



Interaction Between components:



Technologies Used:

- MySQL
- Java
- JSF frame work
- JPA
- Apache Tomcat
- Eclipse Luna
- YQL (Yahoo Finance API) (<https://developer.yahoo.com/yql/console/>)

Number of Tables : 14

Number of Fields : 55

Number of Foreign Keys : 10

Number of Mapping tables : 4

Number of classes : 35 (this includes all the classes made including the model which have classes from JPA)

Number of CRUD operations : 20

API used : YQL (Yahoo Finance API) (<https://developer.yahoo.com/yql/console/>). We used two APIs Yahoo.finance.quote and yahoo.finance.quotes (notice quote and quotes). The first one gives a basic details for stock while the latter gives a very detailed of the stock.

Use Cases for Admin:

- Approve a New Customer
- Delete an Existing Customer
- Add a Portfolio Manager
- Delete a Portfolio Manager
- Assign Portfolio Manager to Customer
- Create a Portfolio Manager

Use Cases for Portfolio Manager:

- Stock suggestions to the assigned customers
- Update profile

Use Cases for Customers:

- Buy/Sell Stocks
- View Portfolio
- View Transaction History

- Manage Account Balance
- See Stock Suggestions.
- Update Profile
- View Portfolio Manager managing the account

Explanation:

Admin creates the portfolio manager and then appoints to the previously approved customer to portfolio manager. Portfolio manager suggest the market trends to the customers using the REST provided by **YQL -Yahoo.finance**.

Customers are approved depending on their bank credit history by Admin. They can inquire about any stock using their TICKR symbol and then accordingly they can either buy or sell that stock using their account details after considering the suggestions given by portfolio manager.

Portfolio manager researches the market using YQL data and then suggests the customers to buy or sell the stock accordingly.

Login Page:

Welcome to Portfolio Management System

[Get Stock Quote](#)

Username	<input type="text"/>
Password	<input type="password"/>

[New User? Signup Now](#)

Getting Stock Quote:

Enter Ticker Symbol:

[Login Page](#)

On clicking get stock quote:

Details for the stock symbol AAPL

Name	Apple Inc.
Days Low	95.61
Days High	96.88
Last Trade Price	95.97
Change	-0.02
Year High	99.44
Year Low	63.8886
Stock Exchange	NasdaqNM

[Get key stats for a particular stock](#)

[Login Page](#)

On clicking key stats for a particular stock get more details:

KeyStats for the stock AAPL

Statistics for AAPL	Term	Content
Market Cap	intraday	574660000000
Enterprise Value	Aug 12, 2014	567890000000
Trailing PE	ttm, intraday	16.10
Forward PE	fye Sep 28, 2015	13.65
PEG Ratio	5 yr expected	1.24
Price Sales	ttm	3.23
Price Book	mrq	4.75
Enterprise value revenue	ttm	3.19
Most Recent Quarter	mrq	Jun 28, 2014
Profit Margin	ttm	21.64%
Operating Margin	ttm	28.84%
Revenue	ttm	178140000000
Revenue Per Share	ttm	28.81
Quarterly Revenue Growth	yoy	6.00%
Gross Profit	ttm	64300000000
NetIncomeAvltoCommon	ttm	38560000000
DilutedEPS	ttm	5.96
QtrlyEarningsGrowth	yoy	12.30%
TotalCash	mrq	37930000000
TotalCashPerShare	mrq	6.33
TotalDebt	mrq	31040000000
TotalDebtEquity	mrq	25.67
CurrentRatio	mrq	1.47
BookValuePerShare	mrq	20.19
OperatingCashFlow	ttm	56370000000
LeveredFreeCashFlow	ttm	37430000000
52WeekHigh	Jul 29, 2014	99.44
52_WeekLow	Sep 16, 2013	63.89
ShortRatio	as of May 30, 2014	1.70
ShortPercentageofFloat	as of May 30, 2014	1.90%

New Customer Registration:

Portfolio Management System Customer Sign up page

Customer Name	<input type="text"/>
Address	<input type="text"/>
Contact Number	<input type="text"/>
Account Number	<input type="text"/>
Email Id	<input type="text"/>
Security Question: Which city were you born?	<input type="text"/>
Username	<input type="text"/>
Password	<input type="password"/>
Retype Password	<input type="password"/>

[Login Page](#)

Admin Module:

Home screen:

Welcome Admin...!!!

[Create Portfolio Manager](#)

[Approve Customer](#)

[Delete Customer](#)

[Delete Portfolio Manager](#)

[Assign Portfolio Manager to Customer](#)

[Delete PM assignment](#)

[Logout](#)

Create Portfolio manager:

Portfolio Manager Name	<input type="text"/>
Address	<input type="text"/>
Contact Number	<input type="text"/>
Username	<input type="text"/>
Password	<input type="password"/>
Retype Password	<input type="password"/>

[Home](#)
[Logout](#)

Approve Customers:

Inactive Customers

Customer ID	Customer Name	Contact Number	Customer Account Number
14	test	9876543210	7876456765

Enter Customer Id to activate

[Home](#)
[Logout](#)

Delete Customers:

All Customers

Customer ID	Customer Name	Contact Number	Customer Account Number	Customer Activation Status
9	dummy	9876543210	9876543245	a
11	Testing	9876543216	9876543245	a
12	Janki	9876543219	9876765645	a
13	Krati	8786756452	9876564521	a
14	test	9876543210	7876456765	d
15	Dixit	8767456724	8767456345	a

Enter Customer Id to delete

[Home](#)

[Logout](#)

Delete Portfolio Manager:

All Portfolio Manager

PM Id	PM Name	Contact Number	PM Address	PM Username
6	Karan Kumar	6179824219	khanna, Punjab	karankk
7	Sahil Khandwala	6179829547	Mumbai	sahilk
8	Nitish Dixit	9876543210	75 St Alphonsus St, Boston, MA	nitish

Enter PM Id to delete

[Home](#)

[Logout](#)

Assign portfolio manager to customer:

Customers

Customer ID	Customer Name
9	dummy
11	Testing
12	Janki
13	Krati
14	test
15	Dixit

Portfolio Managers

Portfolio Manager ID	Portfolio Manager Name
6	Karan Kumar
7	Sahil Khandwala
8	Nitish Dixit

Active Portfolio Managers Allocations

Portfolio Manager ID	Customer ID
8	6
8	9
8	12
8	13

Enter the Customer Id

Enter the Portfolio Manager Id

[Home](#)
[LogOut](#)

Delete PM assignment:

Active Portfolio Managers Allocations

Portfolio Manager ID	Customer ID
8	6
8	9
8	12
8	13

Enter the Customer Id

Enter the Portfolio Manager Id

[Home](#)
[LogOut](#)

Portfolio Manager Module

Home Screen:

Welcome Portfolio Manager...!!!

[Stock Suggestion to Customers](#)

[Update Profile/Change Password](#)

[Logout](#)

Stock Suggestion to Customers

Select which customer to suggest:

List Of Customer

Customer ID	Customer Name	Customer Contact Number	Customer Account Number
9	dummy	9876543210	9876543245
12	Janki	9876543219	9876765645
13	Krati	8786756452	9876564521

Enter the Customer Id

[Home](#)

[Logout](#)

Enter the stock Ticker and suggestions to be given to customers:

Suggest Stock Options to Customer

Enter Stock Ticker	<input type="text"/>
Comments	<input type="text"/>

[Home](#)

[Logout](#)

Customer Module

Home Screen

Welcome Home dixit

What Operation would you like to perform?

Order Shares

View Portfolio

View Transaction History

Account Details

See Stock Suggestion

Update Profile

View Portfolio Mananger

Logout

Order Shares:

Symbol	AAPL
Name	Apple Inc.
Days Low	93.28
Days High	94.82
Last Trade Price	94.74
Change	0.26
Year High	99.44
Year Low	63.8886
Stock Exchange	NasdaqNM

Buy shares for this stock

quantity	<input type="text" value="10"/>
----------	---------------------------------

Place Order

View Portfolio:

DashBoard for the user dixit

Company Symbol	Quantity	Bought At	Current Price	Profit/Loss %	Column
AAPL	10	94.74	94.74	0.0	Sell

View Transaction History:

Transaction History for the user dixit

Company Symbol	Transaction Type	Buy/Sell price per share	Transaction Date	Transaction Date
AAPL	Buy	94.74	10	Sun Aug 10 00:00:00 EDT 2014
AAPL	Sell	94.74	5	Sun Aug 10 00:00:00 EDT 2014

Account Details:

Account Details for the customer dixit

Account Number	8767456345
Balance	0.0

Transfer Funds to PFM System?

Account Number	8767456345
Amount	1000000

Initiate Transfer

See Stock Suggestion:

Stock Suggestions

Suggested Stock	Suggested Date	Comments
AAPL	Sun Aug 10 18:00:47 EDT 2014	Buy more stocks. Market is good.

[Home](#)

[Logout](#)

Update Profile:

Customer Name	<input type="text" value="Dixit"/>
Address	<input type="text" value="Boston"/>
Contact Number	<input type="text" value="8767456724"/>
Username	<input type="text" value="dixit"/>
Password	<input type="password"/>
Retype Password	<input type="password"/>

[Home](#)

[Logout](#)

View Portfolio Manager:

Portfolio Manager Name	Sahil Khandwala
Portfolio Manager Address	Mumbai
Portfolio Manager Contact Number	6179829547

[Home](#)

[Logout](#)

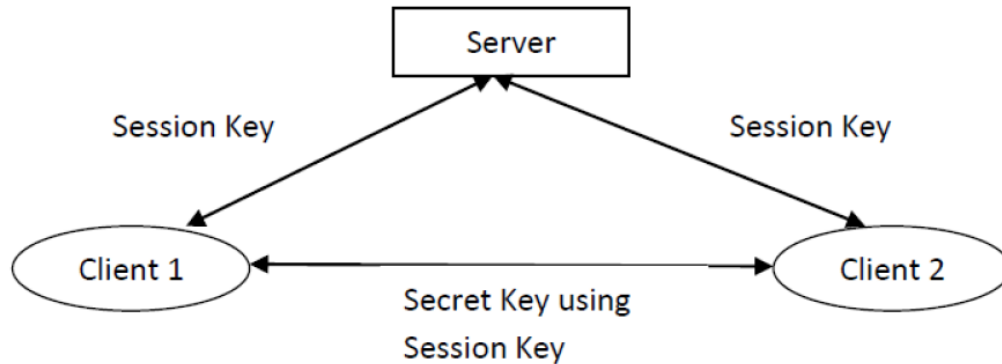
Place Sell Order:

Place Sell Order

Company Id	<input type="text" value="AAPL"/>
Quantity	<input type="text" value="5"/>

WOW Feature:

Highly secure chatting system between different entities



Service Provided:

- Mutual Authentication – Both parties are authenticated each time.
- Identity Hiding – Identity Information (List of online clients) is not sent in clear text.
- Perfect Forward Secrecy – Both parties during client-server and client-client communication set up a Diffie-Hellman key and forget it after session is over.
- Resilience to Denial of Service attacks – Server sends a challenge question which is computationally expensive for the attacker and this makes the system resilient to DoS attacks.
- Message Integrity – Both clients send a MAC along with the message to ensure integrity.

Screenshot of secure chatting system(WOW feature) :

Login to the secure chatting system :

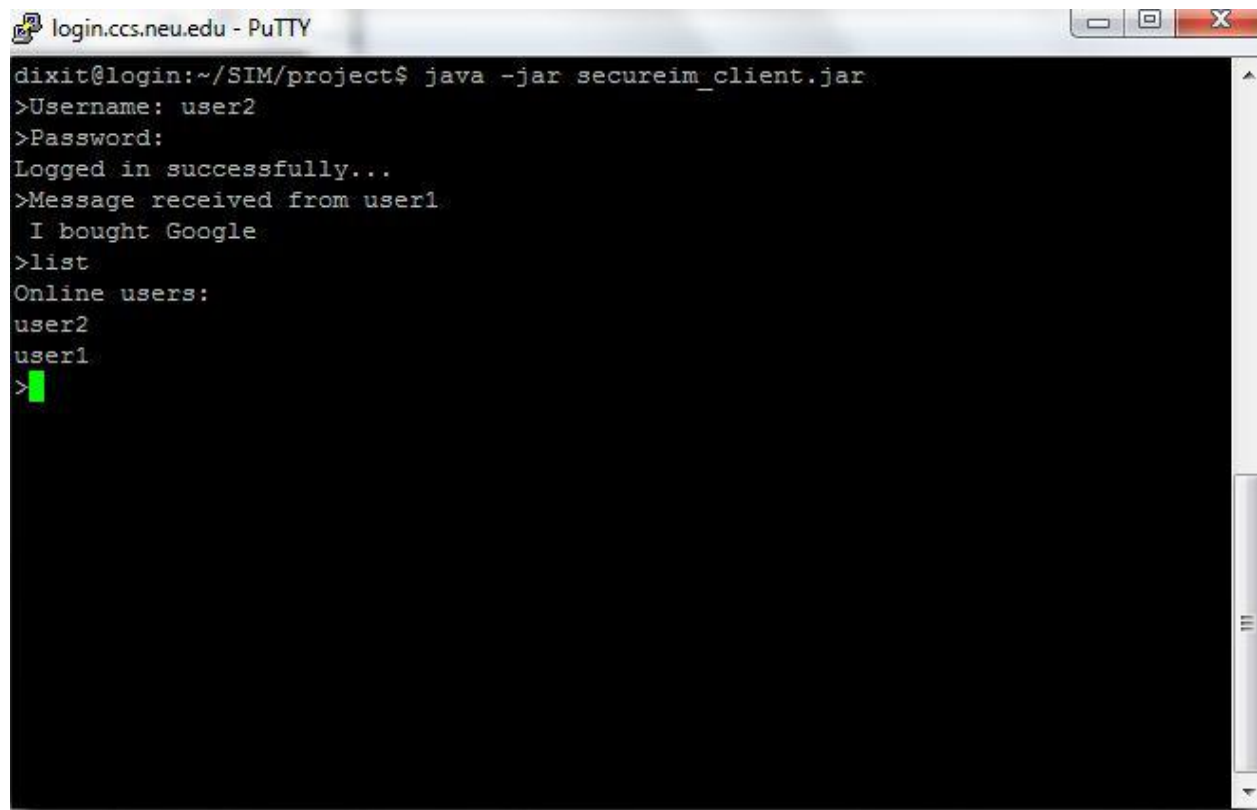
The image shows three terminal windows from a PuTTY session on login.ccs.neu.edu. The top-left window shows the server starting and authenticating two users. The top-right window shows a client logging in as user2. The bottom window shows a client logging in as user1 after navigating through the directory structure.

```
login.ccs.neu.edu - PuTTY
dixit@login:~/SIM/project$ java -jar secureim_server.jar
Reading configuration...
Starting server... DONE
user1 authenticated and added
user2 authenticated and added
█

login.ccs.neu.edu - PuTTY
dixit@login:~/SIM/project$ java -jar secureim_client.jar
>Username: user2
>Password:
Logged in successfully...
>█

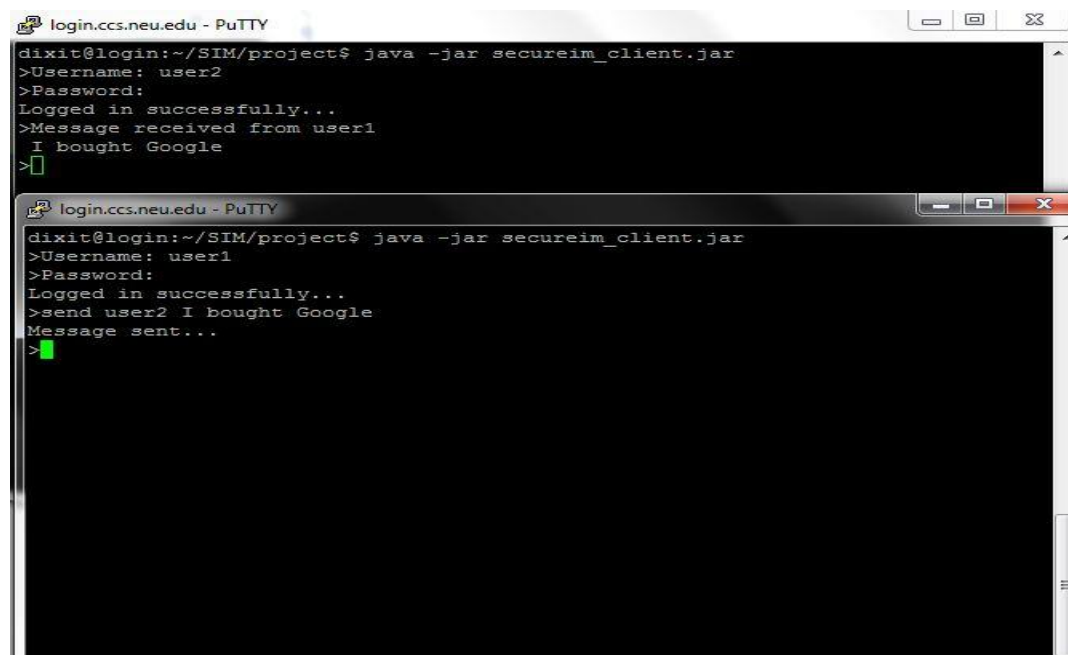
login.ccs.neu.edu - PuTTY
dixit@login:~$ cs SIM
-bash: cs: command not found
dixit@login:~$ cd SIM
dixit@login:~/SIM$ cd project/
dixit@login:~/SIM/project$ java -jar secureim_client.jar
>Username: user1
>Password:
Logged in successfully...
>█
```

List of online users :



```
login.ccs.neu.edu - PuTTY
dixit@login:~/SIM/project$ java -jar secureim_client.jar
>Username: user2
>Password:
Logged in successfully...
>Message received from user1
  I bought Google
>list
Online users:
user2
user1
>
```

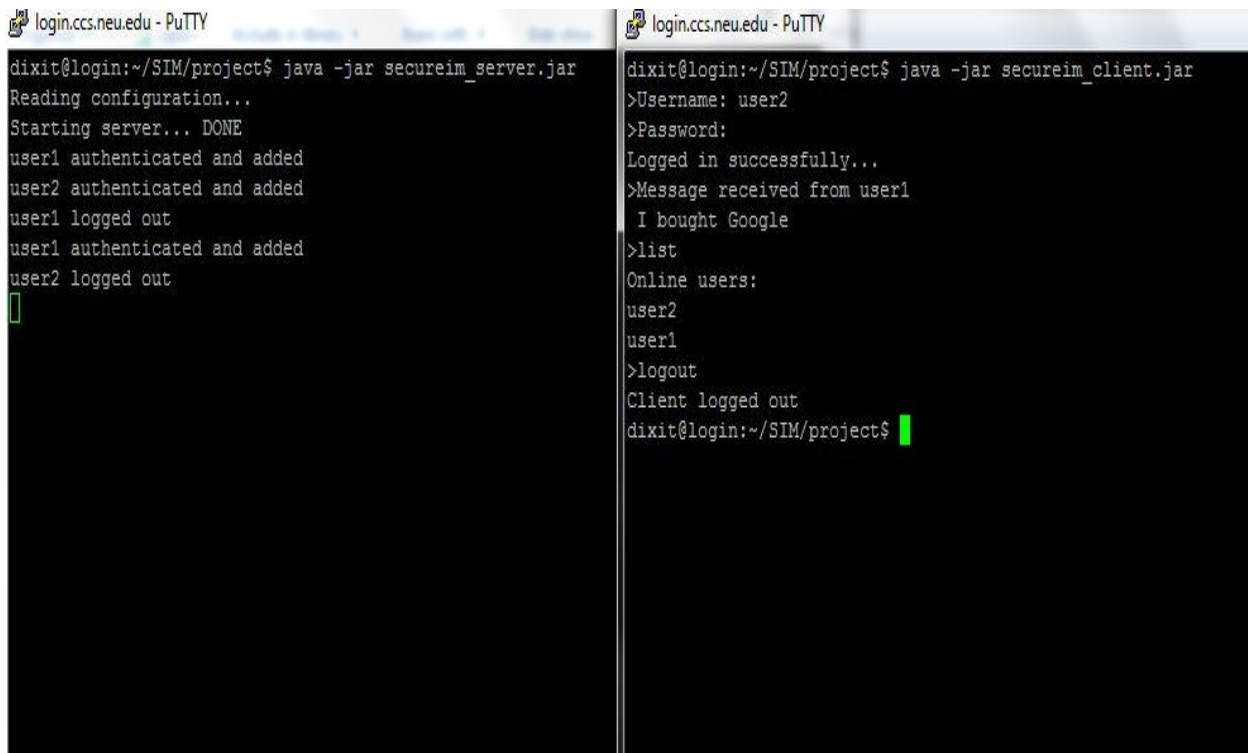
Sending message to another online user :



```
login.ccs.neu.edu - PuTTY
dixit@login:~/SIM/project$ java -jar secureim_client.jar
>Username: user2
>Password:
Logged in successfully...
>Message received from user1
  I bought Google
>

login.ccs.neu.edu - PuTTY
dixit@login:~/SIM/project$ java -jar secureim_client.jar
>Username: user1
>Password:
Logged in successfully...
>send user2 I bought Google
Message sent...
>
```

Logging out :



The image shows two side-by-side terminal windows from a PuTTY session on login.ccs.neu.edu. The left window shows the execution of the secureim_server.jar file. It starts by reading configuration, then starts the server. It receives authentication for user1 and user2, logs user1 out, and then receives authentication for user1 and user2 again, finally logging user2 out. The right window shows the execution of the secureim_client.jar file. It prompts for a username (user2) and password, logs in successfully, receives a message from user1 ("I bought Google"), lists online users (user2 and user1), and then logs out. The prompt changes from login: to dixit@login: after logging out.

```
dixit@login:~/SIM/project$ java -jar secureim_server.jar
Reading configuration...
Starting server... DONE
user1 authenticated and added
user2 authenticated and added
user1 logged out
user1 authenticated and added
user2 logged out
█

dixit@login:~/SIM/project$ java -jar secureim_client.jar
>Username: user2
>Password:
Logged in successfully...
>Message received from user1
  I bought Google
>list
Online users:
user2
user1
>logout
Client logged out
dixit@login:~/SIM/project$ █
```

Challenges Faced:

- Faced problems when interacting with postgresSQL using JPA therefore shifted to MySQL.
- All the tables initially didn't have primary key which was required by the objects created by JPA.
- Understanding JSF framework.

Implementation Differences:

- Planned to have multiple admins controlling a set of portfolio managers but have just one admin who controls all the portfolio managers.
- Planned on using PostgreSQL as a backend but faced problems using JPA with this so shifted to MySQL.