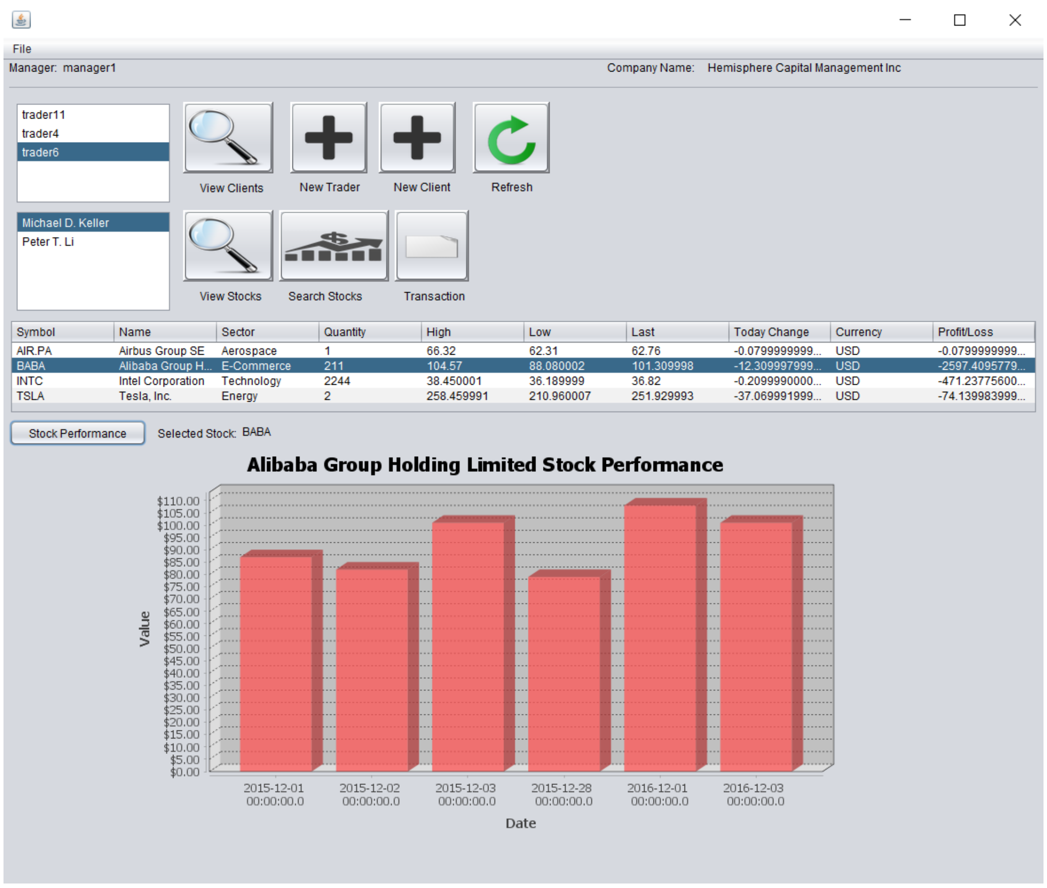


Manager: manager1

Company Name: Hemisphere Capital Management Inc

trader11
trader4
trader6

Michael D. Keller
Peter T. Li

Symbol	Name	Sector	Quantity	High	Low	Last	Today Change	Currency	Profit/Loss
AIR.PA	Airbus Group SE	Aerospace	1	66.32	62.31	62.76	-0.0799999999...	USD	-0.0799999999...
BABA	Alibaba Group H...	E-Commerce	211	104.57	88.080002	101.309998	-12.309997999...	USD	-2597.4095779...
INTC	Intel Corporation	Technology	2244	38.450001	36.189999	36.82	-0.2099990000...	USD	-471.23775600...
TSLA	Tesla, Inc.	Energy	2	258.459991	210.960007	251.929993	-37.069991999...	USD	-74.139983999...

Selected Stock: BABA

Alibaba Group Holding Limited Stock Performance

STOCKIT APPLICATION

CPSC 471 Project Report

ABSTRACT

We proposed a database application which targets investment firms, trading companies and banks. For our project we built a standalone desktop application using Java and a MySQL server. Our application takes in stock market information from a Yahoo Finance API and stores it into our database for further viewing. Our database also stores account information for each type of user of the application, from clients to managers. Depending on the user's role within the company, the application will provide different front-end options, accessible by a login screen.

Andrejs Gusakovs
David Wilkinson
Nizar Maan
Syed Ahmed Zaidi

Introduction:

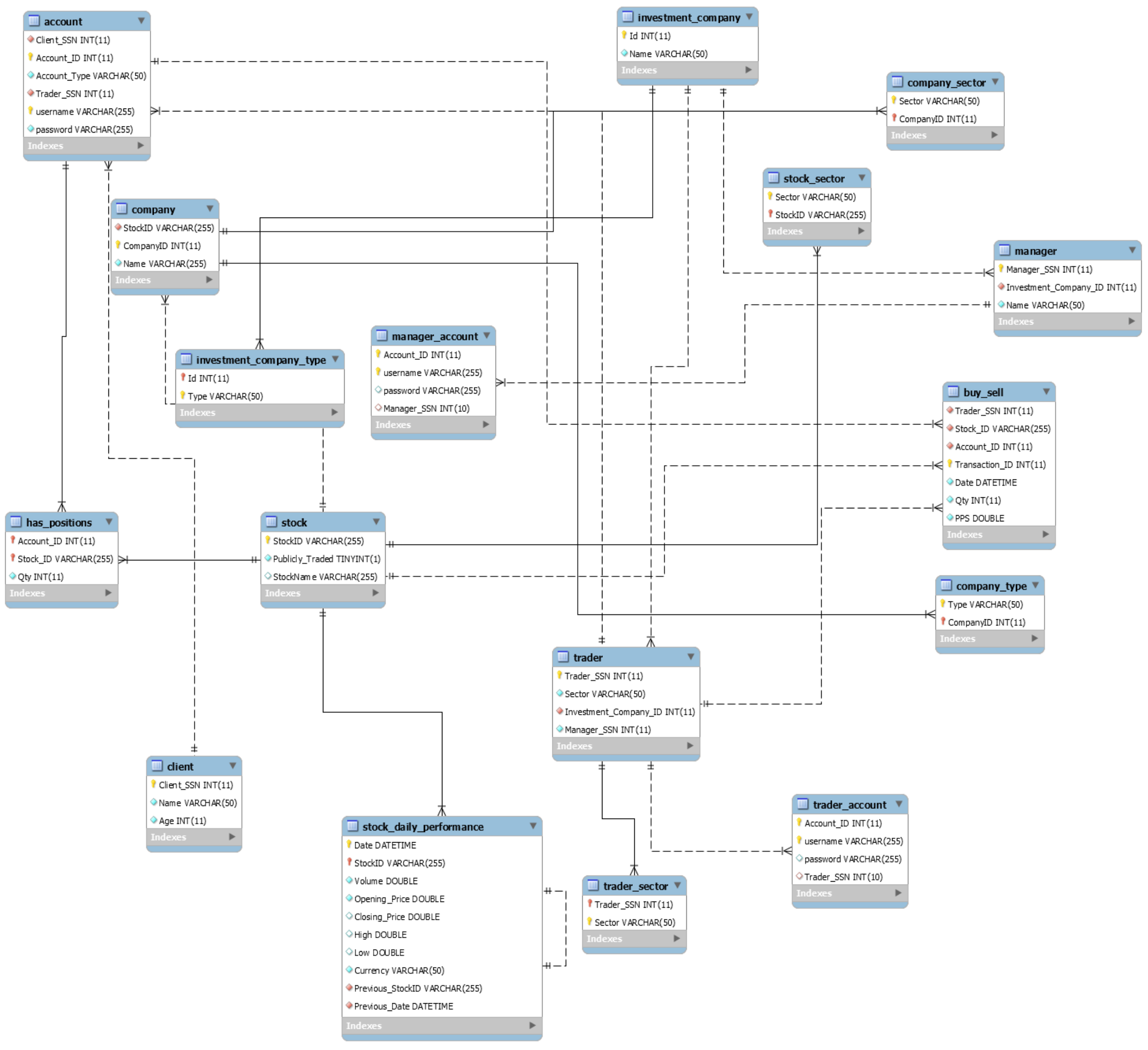
In the financial world, traders and managers within an investment company need access to the historical stock data in order to make informed business decisions for their clients. They rely on the accuracy and simplicity of the data presented to them with a quick overview of their portfolio. Our application intends to provide this feature to traders, managers and clients with the help of a relational database. The database stores all the relevant information regarding stocks and its stakeholders. Traders are able to view their clients' investments with options to buy and sell stocks for their clients' accounts. They also have access to view the current market conditions. Higher level managers are able to add new traders/clients to the system along with having the same set of functionalities as the traders. Clients are able to view the stocks that their trader has invested in on behalf of them. Using a more restrictive application GUI, clients see their overall account information, have the option to change their password, see their account profit/loss and can view which trader is handling their account.

Project Design

The project has three major user types: Traders, Clients and Managers. Traders are added to the database by their managers. The Trader's SSN, their Company and their Manager's SSN are added to the 'trader' table. They also have an account assigned to them. This account information, such as username, password and their Trader's SSN, are added to the 'trader_account' table. Once this information is added to the respective tables in the database, the traders can login to the system using their username and password. The information supplied on the login screen is checked against the stored information in the ‘trader_account’ table. If correct, the trader gets access to their platform. Once logged in, the trader has the ability to analyze the performance of their assigned clients. They also have the option to analyze how the Stock Market is performing. Using this information, the trader is able to buy or sell stocks, on behalf of his/her clients, in order to maximize client profits. This is done by clicking on the transaction button which will allow traders to insert/delete stocks from the client’s ‘had_positions’ table.

For Clients, they are able to view a table of the stocks that their trader has invested in. They can also select the stock and view its historical information. We wanted to give the clients very restricted access to the database so that they are not able to alter most entries. Thus we decided to make a separate table, ‘account’ for clients’ accounts and gave each client access to only change this tables’ password field corresponding to that client. This is done by logging in to the application as a client and going to the settings page. There, a client will be able to view his/her account’s details and will be presented with an option to change their password. If the user wished to change his/her password, he/she will have to first authenticate themselves by providing their current password. Through a set of queries, the password will be matched against the user's current stored password in the ‘account’ table under the password attribute. If the entered password matches the entry in the database for the user, only then the field will be updated to a new password.

For Managers, their personal information is initially inputted by the database operator. This information is added to the ‘manager’ and ‘manager_account’ tables. From here, the manager can login, where the application will check the given username and password with the username and password in the ‘manager_account’ table. When logged in, the manager has access to creating new traders for his/her company, by adding their information to the ‘trader’ table, along with the ‘trader_account’ table. He/she can also add clients in a similar manner, adding their information to the ‘client’ and ‘account’ tables. The manager GUI allows for viewing the list of assigned traders, by matching the manager's SSN with the 'manager_SSN' in the 'trader' table. Similar to the trader GUI, once a trader is selected, the list of that trader's clients can also be obtained by matching the trader's SSN with the 'trader_SSN' in the 'account' table. The manager is then also able to make purchases on behalf of the client, similarly to the trader GUI.



EER DIAGRAM

Implementation Section

For the most part, our conversion from EER diagram to relational schema remained the same when we followed the algorithm presented in class. However, we were having difficulty planning out the authentication process for Managers, Traders and Clients. Thus, we decided to separate them by creating the following tables: ‘manager_account’, ‘trader_account’, and ‘account’ for Managers, Traders, and Clients, respectively. This allowed us to separate the login sessions of the different type of users. Each user selects his/her role, and authenticates his/herself by providing correct login information. We will go in depth with the authentication protocol in the later section. Initially, we planned to use the table ‘buy_sell’ to record a stock transaction (the act of buying or selling a stock). This would have allowed us to build a history of transactions done on a client’s account by a trader. The stock transactions were thought to be dealt via the use of ‘has_positions’ and ‘buy_sell’ tables. The ‘has_positions’ would keep track of an account’s stocks, and the ‘buy_sell’ table would allow us to fetch which Day and Time, the stock was bought. Due to time constraints, we decided to provide the M-V-P (Minimal Viable Product) where we only considered dealing with ‘has_positions’ table. To build and manage our database, we agreed upon MySQL for a database management system. We used the Github to manage our project and divide the tasks amongst each other. Our installation of MySQL included MySQL Server which we configured ourselves. It also came with MySQL Workbench to monitor our database. The database was populated with real world stock information. To get this information, we interfaced with Yahoo Finance, via their API, to provide us with stock information. For simplicity, we decided to limit our stock mini-world to 5 Stocks (although our project is easily extendible to fetch and store as many stocks as required), namely: Airbus, Alibaba, Alphabet, Intel, Tesla, and Yahoo. The information for the stock was stored in the ‘stock’, ‘stock_sector’, ‘stock_daily_performance’, and ‘company’ tables. Our ‘stock_daily_performance’ also has a recursive relationship to itself to link to the previous day’s performance. We tested our SQL statements of each of our transactions, such as getting stock information, on MySQL Workbench to ensure that we were getting correct results back. A complete list of all of the transactions that our application supports on each frame is provided in the next section.

Transaction SQL Statements

1. **Manager Login Frame**
    - Get username and password of a Manager:
        - SELECT username, password FROM manager_account WHERE username= usname AND password = newStr
    - Get investment company of the authenticated Manager
        - SELECT investment_company.Name FROM investment_company, manager, manager_account WHERE manager_account.username = array[0] AND manager.Manager_SSN= manager_account.Manager_SSN AND investment_company.Id = manager.Investment_Company_ID
2. **Trader Login Frame**
    - Get username and password of a Trader
        - SELECT username, password FROM trader_account WHERE username= usname AND password = newStr
    - Get investment company of the authentication Trader
        - SELECT Name FROM investment_company, trader, trader_account WHERE investment_company.Id = trader.Investment_Company_ID AND trader.Trader_SSN = trader_account.Trader_SSN AND trader_account.username = array[0]
3. **Client Login Frame**
    - Get username and password of a Client
        - SELECT username, password FROM account WHERE username= usname AND password = newStr
4. **Manager Frame**
    - Get list of all the traders under a Manager
        - SELECT tc.username FROM trader as t, manager_account as mc trader_account as tc WHERE t.Manager_SSN = mc.Manager_SSN AND

tc.Trader_SSN = t.Trader_SSN AND mc.username = username

- Get list of all the clients under a selected trader
    - SELECT C.Name FROM client as C, trader as t, trader_account as tc, account as a WHERE C.Client_SSN = a.Client_SSN AND tc.Trader_SSN = t.Trader_SSN AND a.Trader_SSN = t.Trader_SSN AND tc.username = selectedTrader
- View a selected Clients invested stocks according to the most recent stock date
    - SELECT stock.stockID, company.Name, stock_sector.Sector, stock_daily_performance.Currency, stock_daily_performance.High, stock_daily_performance.Low,stock_daily_performance.Closing_Price,has_positions.Qty,(stock_daily_performance.Opening_Price-stock_daily_performance.Closing_Price) AS Today_Change, ((stock_daily_performance.Opening_Price-stock_daily_performance.Closing_Price) * has_positions.Qty) AS Profit FROM stock, stock_sector, stock_daily_performance, has_positions, account, client, company WHERE client.Name = selectedClient AND client.Client_SSN = account.Client_SSN AND account.Account_ID = has_positions.Account_ID AND stock_daily_performance.StockID = has_positions.Stock_ID AND stock.StockID = has_positions.Stock_ID AND stock_sector.StockID = has_positions.Stock_ID AND company.StockID = stock.StockID AND Date IN (Select * from (SELECT max(Date) FROM stock_daily_performance, stock where stock_daily_performance.StockID=stock.StockID group by stock.StockID ORDER BY Date) temp_table)
- View a Stock’s closing price according to Dates for graphical purposes:
    - Select stock_daily_performance.Closing_Price, stock_daily_performance.Date FROM stock_daily_performance, stock WHERE stock_daily_performance.StockID = stock.StockID AND

stock.StockID = selectedStock AND Date IN (Select * from(SELECT Date FROM stock_daily_performance WHERE StockID = stockID ORDER BY Date ASC) temp_table)

- Add a new trader:
    - INSERT INTO trader(Trader_SSN, Sector, Investment_Company_ID, Manager_SSN) VALUES(traderssnbox, sectorbox, compid, manssn)
    - INSERT INTO trader_sector(Trader_SSN, Sector) VALUES(traderssnbox, sectorbox)
    - INSERT INTO trader_account(Trader_SSN, username, password) VALUES(traderssnbox,usernamebox,passwordbox)
- Add a new client:
    - INSERT INTO client(Client_SSN, Name, Age) VALUES(clientssnbox, namebox, clientagebox)
    - INSERT INTO account(Client_SSN, Account_Type, username, password, Trader_SSN) VALUES(clientssnbox, accounttypebox, usernamebox1, passwordbox1, traderssn)

5. **Trader Frame**

- List of all the clients under the current trader:
    - SELECT C.Name FROM client as C, trader as t, trader_account as tc, account as a WHERE C.Client_SSN = a.Client_SSN AND tc.Trader_SSN = t.Trader_SSN AND a.Trader_SSN = t.Trader_SSN AND tc.username = username
- View a selected Clients invested stocks according to the most recent stock date:
    - SELECT stock.stockID, company.Name, stock_sector.Sector, stock_daily_performance.Currency, stock_daily_performance.High, stock_daily_performance.Low,stock_daily_performance.Closing_Price,has_positions.Qty,(stock_daily_performance.Opening_Price-stock_daily_performance.Closing_Price) AS Today_Change,

((stock_daily_performance.Opening_Price-stock_daily_performance.Closing_Price) * has_positions.Qty) AS Profit FROM stock, stock_sector, stock_daily_performance, has_positions, account, client, company WHERE client.Name = selectedClient AND client.Client_SSN = account.Client_SSN AND account.Account_ID = has_positions.Account_ID AND stock_daily_performance.StockID = has_positions.Stock_ID AND stock.StockID = has_positions.Stock_ID AND stock_sector.StockID = has_positions.Stock_ID AND company.StockID = stock.StockID AND Date IN (Select * from (SELECT max(Date) FROM stock_daily_performance, stock where stock_daily_performance.StockID=stock.StockID group by stock.StockID ORDER BY Date) temp_table)

- View a selected Stock's closing price according to Dates for graphical purposes:
    - Select stock_daily_performance.Closing_Price, stock_daily_performance.Date FROM stock_daily_performance, stock WHERE stock_daily_performance.StockID = stock.StockID AND stock.StockID = selectedStock AND Date IN (Select * from(SELECT Date FROM stock_daily_performance WHERE StockID = stockID ORDER BY Date ASC) temp_table)

6. **Client Frame**

- List of all the invested stock of a client:
    - SELECT stock.stockID, company.Name, stock_sector.Sector, stock_daily_performance.Currency, stock_daily_performance.High, stock_daily_performance.Low,stock_daily_performance.Closing_Price,has_positions.Qty,(stock_daily_performance.Opening_Price-stock_daily_performance.Closing_Price) AS Today_Change, ((stock_daily_performance.Opening_Price-stock_daily_performance.Closing_Price) * has_positions.Qty) AS Profit

FROM stock, stock_sector, stock_daily_performance, has_positions, account, client, company WHERE client.Name = name AND client.Client_SSN = account.Client_SSN AND account.Account_ID = has_positions.Account_ID AND stock_daily_performance.StockID = has_positions.Stock_ID AND stock.StockID = has_positions.Stock_ID AND stock_sector.StockID = has_positions.Stock_ID AND company.StockID = stock.StockID AND Date IN (Select * from (SELECT max(Date) FROM stock_daily_performance, stock where stock_daily_performance.StockID=stock.StockID group by stock.StockID ORDER BY Date) temp_table)

- View a selected Stock's closing price according to Dates for graphical purposes:
  - Select stock_daily_performance.Closing_Price, stock_daily_performance.Date FROM stock_daily_performance, stock WHERE stock_daily_performance.StockID = stock.StockID AND stock.StockID = selectedStock AND Date IN (Select * from(SELECT Date FROM stock_daily_performance WHERE StockID = stockID ORDER BY Date ASC) temp_table)

7. **allStock Frame**

- View all the Stocks on the market:
  - select stock.StockID, stock.Publicly_Traded, stock.StockName, stock_sector.Sector,stock_daily_performance.Volume,stock_daily_performance.Currency,stock_daily_performance.Opening_Price,stock_daily_performance.Closing_Price,stock_daily_performance.High,stock_daily_performance.Low, stock_daily_performance.Date from stock, stock_sector, stock_daily_performance WHERE stock.StockID = stock_sector.StockID AND stock_daily_performance.StockID = stock.StockID AND stock_daily_performance.Date IN (Select * from (SELECT max(Date) FROM stock_daily_performance, stock where stock_daily_performance.StockID=stock.StockID group by stock.StockID

ORDER BY Date) temp_table)

- View a selected Stock’s closing price according to Dates for graphical purposes:
    - Select stock_daily_performance.Closing_Price, stock_daily_performance.Date FROM stock_daily_performance, stock WHERE stock_daily_performance.StockID = stock.StockID AND stock.StockID = selectedStock AND Date IN (Select * from(SELECT Date FROM stock_daily_performance WHERE StockID = stockID ORDER BY Date ASC) temp_table)

8. **Transaction (Buy/Sell Stock) Frame**

- Buying a Stock:
    - INSERT INTO has_positions (Account_ID, Stock_ID, Qty) VALUES (accountID, stockName, 1) ON DUPLICATE KEY UPDATE Stock_ID= stockName, QTY= QTY + 1
    - UPDATE stock_daily_performance SET Volume = Volume - 1 WHERE stock_daily_performance.StockID = stockName
- Selling a Stock:
    - UPDATE has_positions SET Qty = Qty - 1 WHERE has_positions.Account_ID = accountID AND has_positions.Stock_ID = stockName
    - UPDATE stock_daily_performance SET Volume = Volume + 1 WHERE stock_daily_performance.StockID = stockName
- Set up Stock Market table
    - See Section “allStock Frame” - View all the Stocks on the market
- Set up Client’s Stocks table
    - See Section “Client Frame” - List of all the invested stock of a client

9. **ClientAccountInfo Frame**

- Change Password
    - UPDATE account SET password = dummyPwd where account.Account_ID

= ClientFrame.getAccountId()

User Interface

Managers:

Upon the start of the application, one is presented with a selection screen, as seen in Figure 1. The user selected his/her role, and is prompted to login. The login screen is identical for Manager, Trader, Clients, as shown in Figure 2. It's kept simple and standard, requesting only a username and password.

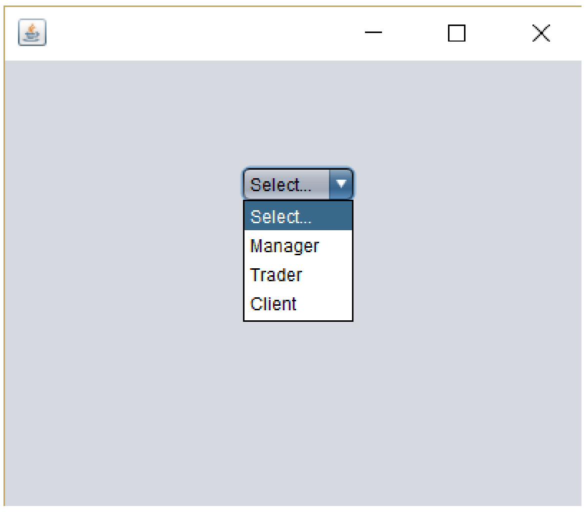


FIGURE 1: SELECTION SCREEN

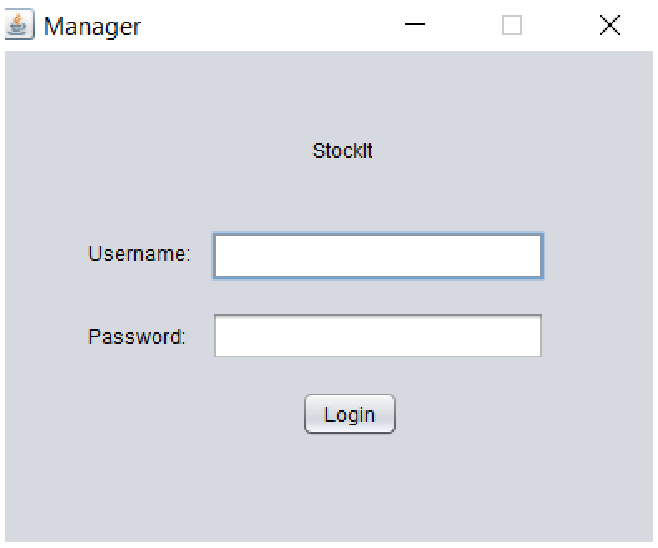
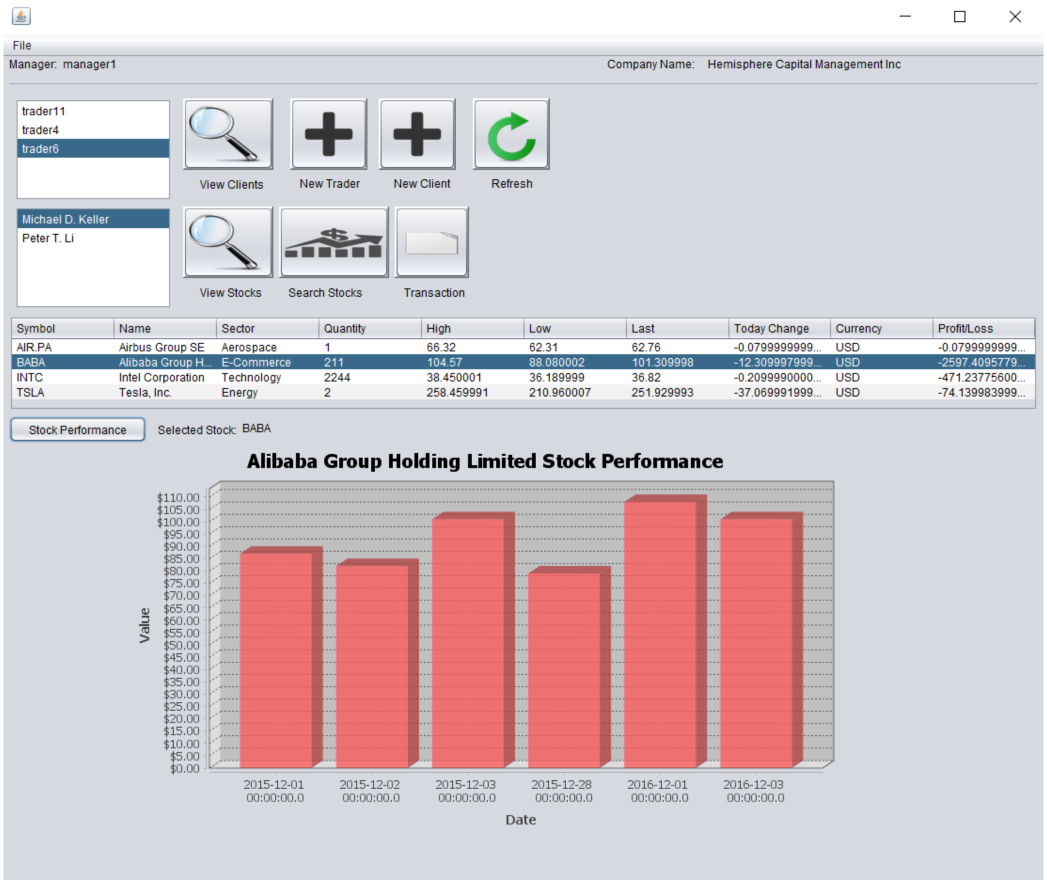


FIGURE 2: LOGIN SCREEN FOR MANAGERS

As shown in Figure 3, upon authenticating as a manager, one is presented with the main screen for a manager. The manager is able to see his/her username and the company he/she works for. The manager also sees the traders they manage, and are able to view the clients that each trader handles. This way, they can see how their traders' clients are performing. Managers are also able to perform transactions for the clients (covered under the Trader section). Managers have escalated privileges that allow them to add traders and clients. To add a trader, the manager has to provide the following information: a username, a password, the trader's SSN and the trading sector. Similarly, to add a client, the manager provides the client's name, username, password, client's SSN and age. Frames for adding a Trader and a Client are shown in Figure 4.1 and Figure 4.2.

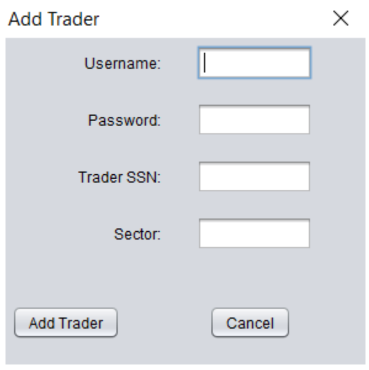

Manager: manager1

Company Name: Hemisphere Capital Management Inc

Symbol	Name	Sector	Quantity	High	Low	Last	Today Change	Currency	Profit/Loss
AIR.PA	Airbus Group SE	Aerospace	1	66.32	62.31	62.76	-0.0799999999...	USD	-0.0799999999...
BABA	Alibaba Group H...	E-Commerce	211	104.57	88.080002	101.309998	-12.309997999...	USD	-2597.4095779...
INTC	Intel Corporation	Technology	2244	38.450001	36.189999	36.82	-0.2099990000...	USD	-471.23775600...
TSLA	Tesla, Inc.	Energy	2	258.459991	210.960007	251.929993	-37.069991999...	USD	-74.139983999...

Selected Stock: BABA

FIGURE 3: MANAGER SCREEN



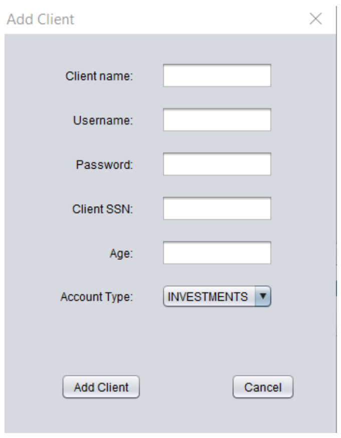
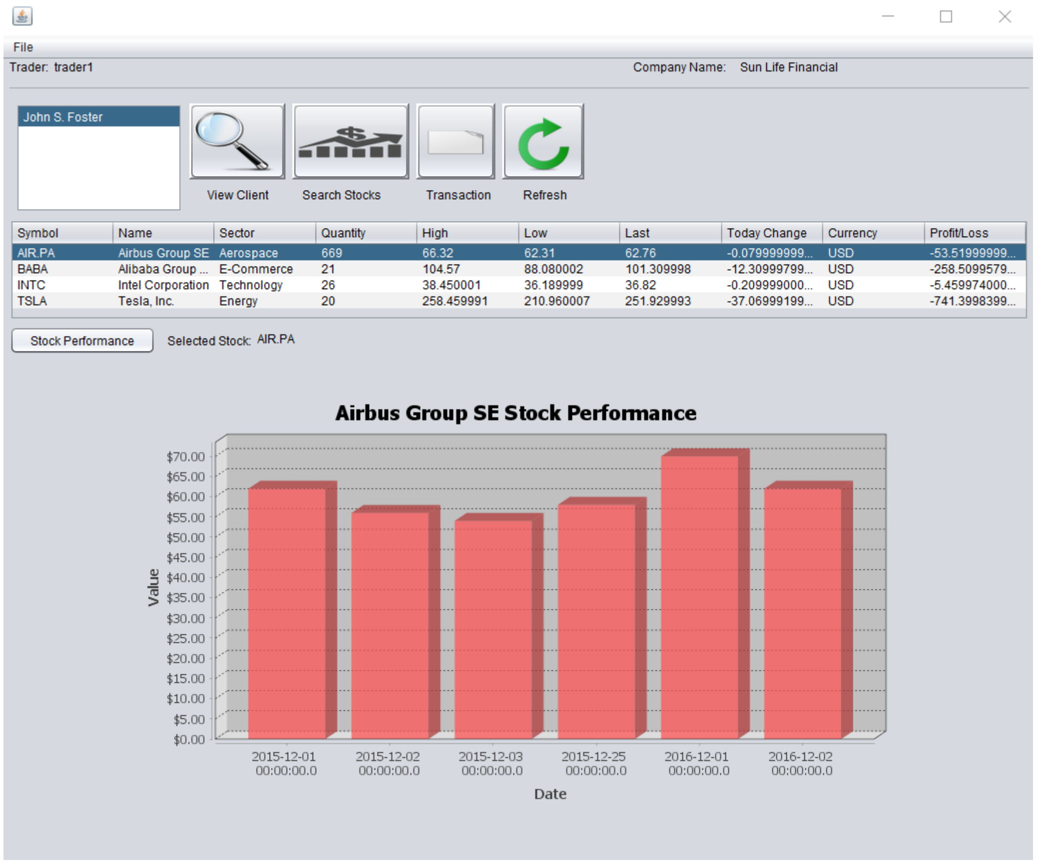


FIGURE 4.1: ADD TRADER

FIGURE 4.2: ADD CLIENT

Traders:

Traders are similar to Managers, with the exception that they are unable to add Traders or Clients to the database, and can only see their own clients. The login interface is identical to a Manager's. The main screen for a Trader is shown below in Figure 5 with 'trader1' as an example.



Trader: trader1

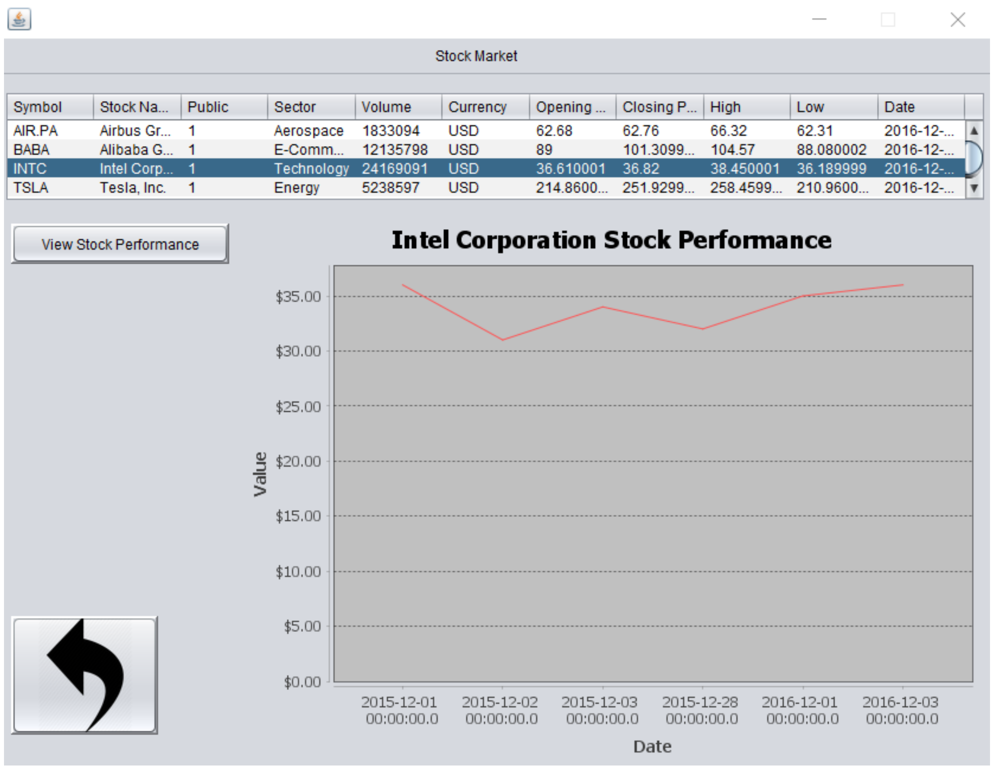
Company Name: Sun Life Financial

Symbol	Name	Sector	Quantity	High	Low	Last	Today Change	Currency	Profit/Loss
AIR.PA	Airbus Group SE	Aerospace	669	66.32	62.31	62.76	-0.079999999...	USD	-53.51999999...
BABA	Alibaba Group ...	E-Commerce	21	104.57	88.080002	101.309998	-12.30999799...	USD	-258.5099579...
INTC	Intel Corporation	Technology	26	38.450001	36.189999	36.82	-0.209999000...	USD	-5.459974000...
TSLA	Tesla, Inc.	Energy	20	258.459991	210.960007	251.929993	-37.06999199...	USD	-741.3998399...

Selected Stock: AIR.PA

FIGURE 5: TRADER MAIN SCREEN

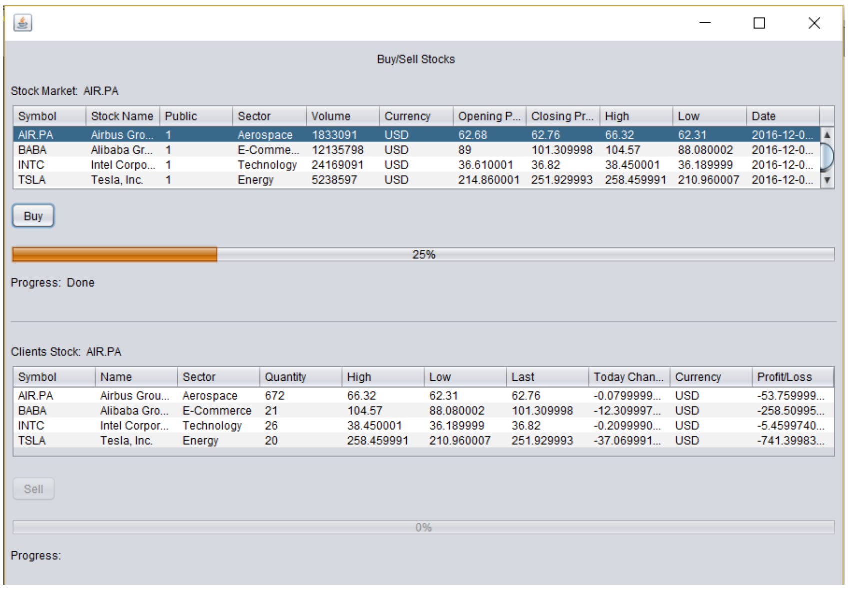
Like Managers, Traders are able to view all the stocks available on the market, and how they are performing (Figure 6). They are also able to perform transactions, as shown in Figure 7. In the stock market screen, the trader can see how a stock is performing historically through the help of a line graph.


Stock Market

Symbol	Stock Na...	Public	Sector	Volume	Currency	Opening ...	Closing P...	High	Low	Date
AIR.PA	Airbus Gr...	1	Aerospace	1833094	USD	62.68	62.76	66.32	62.31	2016-12-...
BABA	Alibaba G...	1	E-Comm...	12135798	USD	89	101.3099...	104.57	88.080002	2016-12-...
INTC	Intel Corp...	1	Technology	24169091	USD	36.610001	36.82	38.450001	36.189999	2016-12-...
TSLA	Tesla, Inc.	1	Energy	5238597	USD	214.8600...	251.9299...	258.4599...	210.9600...	2016-12-...

FIGURE 6: STOCK MARKET

In the transaction screen, the trader is able to see each stock his/her client owns and each stock from the stock market. The trader can select a stock accordingly, and buy or sell it. Immediately, the corresponding table is updated and the new information is displayed.


Buy/Sell Stocks

Stock Market: AIR.PA

Symbol	Stock Name	Public	Sector	Volume	Currency	Opening P...	Closing Pr...	High	Low	Date
AIR.PA	Airbus Gro...	1	Aerospace	1833091	USD	62.68	62.76	66.32	62.31	2016-12-0...
BABA	Alibaba Gr...	1	E-Comme...	12135798	USD	89	101.309998	104.57	88.080002	2016-12-0...
INTC	Intel Corpo...	1	Technology	24169091	USD	36.610001	36.82	38.450001	36.189999	2016-12-0...
TSLA	Tesla, Inc.	1	Energy	5238597	USD	214.860001	251.929993	258.459991	210.960007	2016-12-0...

Progress: Done

Clients Stock: AIR.PA

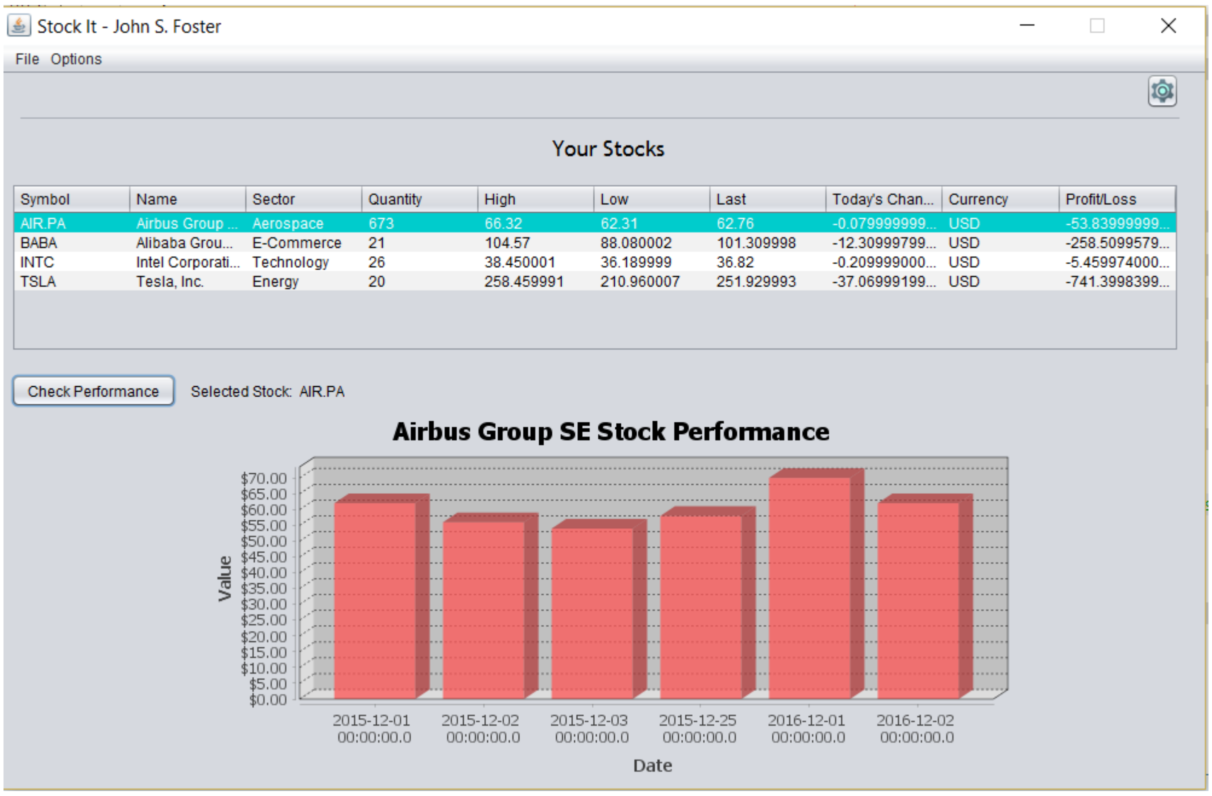
Symbol	Name	Sector	Quantity	High	Low	Last	Today Chan...	Currency	Profit/Loss
AIR.PA	Airbus Grou...	Aerospace	672	66.32	62.31	62.76	-0.0799999...	USD	-53.759999...
BABA	Alibaba Gro...	E-Commerce	21	104.57	88.080002	101.309998	-12.309997...	USD	-258.50995...
INTC	Intel Corpor...	Technology	26	38.450001	36.189999	36.82	-0.2099990...	USD	-5.4599740...
TSLA	Tesla, Inc.	Energy	20	258.459991	210.960007	251.929993	-37.069991...	USD	-741.39983...

Progress:

FIGURE 7: TRANSACTION SCREEN

Clients:

Due to the Client's role within the application, they are provided with the more restrictive front end option. The login interface is identical to the Manager's and Trader's login screen. The main screen for Clients is shown below in Figure 8 with 'client1' (John S. Foster) as an example.



Your Stocks

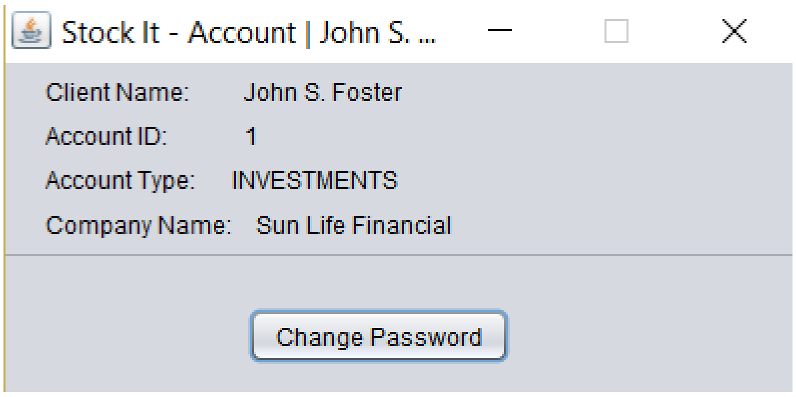
Symbol	Name	Sector	Quantity	High	Low	Last	Today's Chan...	Currency	Profit/Loss
AIR.PA	Airbus Group ...	Aerospace	673	66.32	62.31	62.76	-0.079999999...	USD	-53.83999999...
BABA	Alibaba Grou...	E-Commerce	21	104.57	88.080002	101.309998	-12.30999799...	USD	-258.5099579...
INTC	Intel Corporati...	Technology	26	38.450001	36.189999	36.82	-0.209999000...	USD	-5.459974000...
TSLA	Tesla, Inc.	Energy	20	258.459991	210.960007	251.929993	-37.06999199...	USD	-741.3998399...

Selected Stock: AIR.PA

Airbus Group SE Stock Performance

FIGURE 8: CLIENT - JOHN S. FOSTER

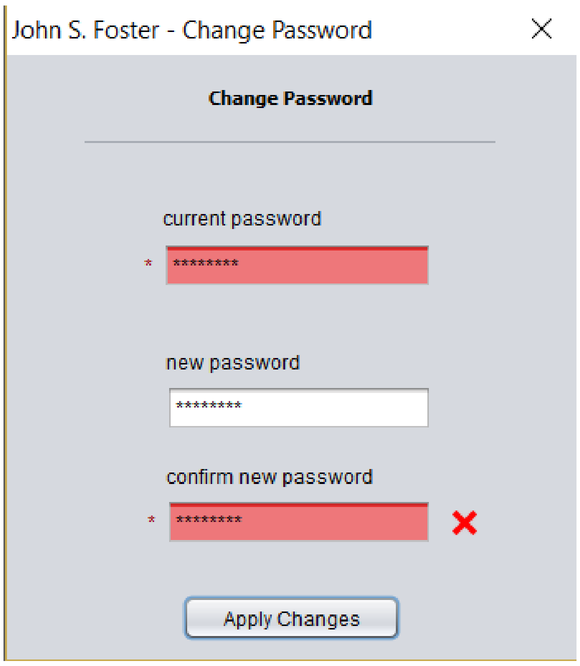
Just like Traders and Managers, Clients can see the stocks their traders have invested in on behalf of them. In addition, they are able to see their account information and change their password by clicking on the settings icon or by selecting "Options" -> "Account" (Figure 9).


Stock It - Account | John S. ...

Client Name: John S. Foster
Account ID: 1
Account Type: INVESTMENTS
Company Name: Sun Life Financial

FIGURE 9: ACCOUNT INFORMATION

If the Client desires to change his/her password, they will be taken to the change password screen as shown in Figures 10.1 and 10.2. Clients must correctly provide all fields, otherwise they will not be able to proceed.


John S. Foster - Change Password

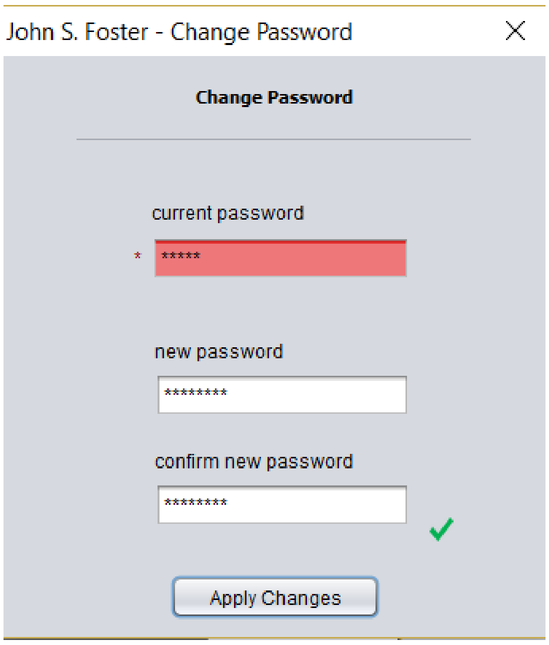
Change Password

current password
* ********

new password
****

confirm new password
* ********

FIGURE 10.1: INCORRECT FIELDS V1


John S. Foster - Change Password

Change Password

current password
* *****

new password
****

confirm new password
****

FIGURE 10.2: INCORRECT FIELDS V2