Application Project Phase 3

Title

Stock trading system

Team

- Anubola Sai Abhinay (180010006)
- Mohan Chandrakanth (180010003)
- Chilaka Avinash (180010011)

Introduction

LOGIN interface will have forms for username, password and login option. After login using login credentials if he is customer, then go to customer interface. After login using login credentials if he is employee, then go to employee interface.

CUSTOMER interface will have account info, portfolios, orders, stocks and logout. On selecting account info, we will get our account information. On selecting portfolios, we will get information about the stocks that we currently own. On selecting orders, we will get your orders. On selecting stocks, we can search for stocks and place orders. On selecting logout, we can logout of our account.

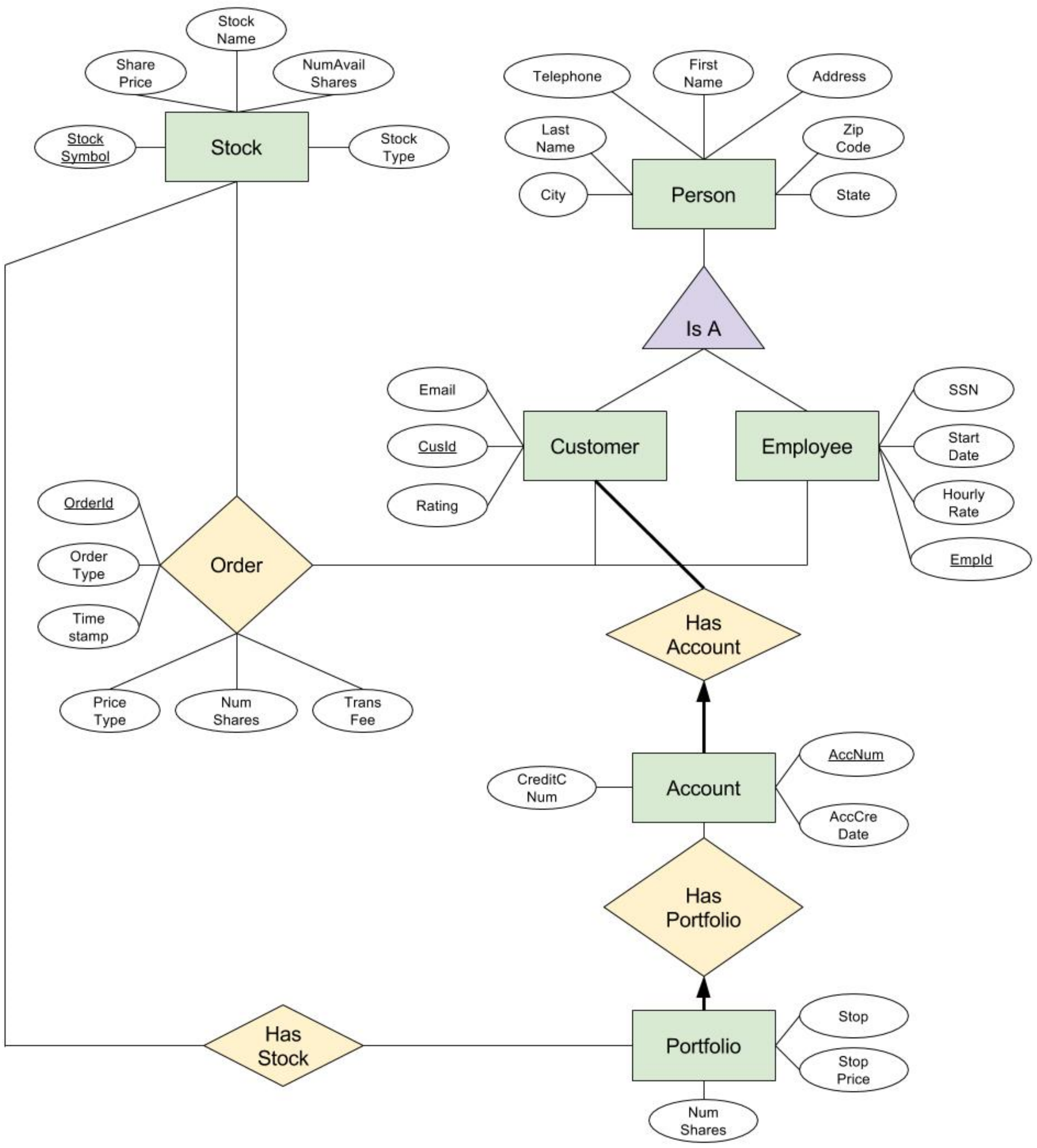
EMPLOYEE interface is divided into two parts which are Customer Representative interface and Manager interface.

CUSTOMER REPRESENTATIVE interface will have account info, customers, orders, stocks and logout. On selecting account info, we will get our account information. On selecting customers, we can view, add, edit, and delete Customers. On selecting orders, we can view orders. On selecting stocks, we can search for stocks and add stocks. On selecting logout, we can logout of our account.

MANAGER interface will have account info, employees, orders, stocks and logout. On selecting account info, we will get our account information. On selecting employees, we can view, add, edit, and delete employees. On selecting orders, we can view and record orders. On selecting stocks, we can search for stocks and set stock prices. On selecting logout, we can logout of our account.

Implementation overview: module diagram and DB diagram

ER Model:



Relational database design:

```
CREATE TABLE Stock (
	StockSymbol		VARCHAR(5) NOT NULL,
	StockName		VARCHAR(20) NOT NULL,
	StockType		VARCHAR(20),
	SharePrice		FLOAT(2) NOT NULL,
	NumAvailShares		INTEGER NOT NULL,
	PRIMARY KEY (StockSymbol),
	UNIQUE (StockName)
);

CREATE TABLE Employee (
	SSN				CHAR(9) NOT NULL,
	LastName		VARCHAR(20),
	FirstName		VARCHAR(20),
	Address			VARCHAR(50),
	City			VARCHAR(20),
	State			VARCHAR(20),
	ZipCode				CHAR(5),
	Telephone		CHAR(10),
	StartDate		TIMESTAMP,
	HourlyRate		FLOAT(2),
	EmpId			SERIAL NOT NULL,
	Position_		VARCHAR(7) NOT NULL,
	PRIMARY KEY (EmpId),
	UNIQUE (SSN)
);


CREATE TABLE Customer (
	LastName		VARCHAR(20) NOT NULL,
	FirstName		VARCHAR(20) NOT NULL,
	Address			VARCHAR(50),
	City			VARCHAR(20),
	State			VARCHAR(20),
	ZipCode				CHAR(5),
	Telephone		CHAR(10),
	Email			VARCHAR(50),
	Rating			INTEGER NOT NULL,
	CusId			SERIAL NOT NULL,
	PRIMARY KEY (CusId)
);
```

```
-- AccType: 1 for customer, 2 for employee, 3 for manager
CREATE TABLE Login (
        Usr                     VARCHAR(20) NOT NULL,
        Pwd                     VARCHAR(20) NOT NULL,
        AccType                 INTEGER NOT NULL,
        Id                      INTEGER,
        PRIMARY KEY (Usr)
);

CREATE TABLE Account_ (
        AccNum                  SERIAL NOT NULL,
        AccCreDate          TIMESTAMP,
        CreditCNum          VARCHAR(16) NOT NULL,
        CusId               INTEGER NOT NULL,
        PRIMARY KEY (AccNum),
        FOREIGN KEY (CusId) REFERENCES Customer (CusId)
                ON DELETE NO ACTION
                ON UPDATE CASCADE
);

CREATE TABLE Order_ (
        OrderId             SERIAL,
        StockSymbol         VARCHAR(5),
        OrderType           VARCHAR(4) NOT NULL,
        NumShares           INTEGER NOT NULL,
        CusAccNum           INTEGER DEFAULT 0,
        Timestamp_          TIMESTAMP WITH TIME ZONE DEFAULT CURRENT_TIMESTAMP,
        PriceType           VARCHAR(15) NOT NULL,
        StopPrice           FLOAT(2) DEFAULT 0,
   StopDiff             FLOAT(2),
        CurSharePrice   FLOAT(2),
        EmpId               INTEGER DEFAULT 0,
        Recorded            BOOLEAN DEFAULT '0',
   Completed            BOOLEAN DEFAULT '0',
        PRIMARY KEY (OrderId),
        UNIQUE (StockSymbol, Timestamp_, CusAccNum, EmpId),
        FOREIGN KEY (StockSymbol) REFERENCES Stock (StockSymbol)
                ON DELETE SET NULL
                ON UPDATE CASCADE,
        FOREIGN KEY (CusAccNum) REFERENCES Account_ (AccNum)
                ON DELETE SET NULL      -- changed from SET DEFAULT
                ON UPDATE CASCADE,
        FOREIGN KEY (EmpId) REFERENCES Employee (EmpId)
                ON DELETE SET NULL
                ON UPDATE CASCADE
);
```

```
CREATE TABLE Transact (
        Id                              SERIAL,
        OrderId                 INTEGER,
        TransFee                FLOAT(2),
        TimeStamp_              TIMESTAMP WITH TIME ZONE DEFAULT CURRENT_TIMESTAMP,
        PricePerShare   FLOAT(2),
        PRIMARY KEY (Id),
        FOREIGN KEY (OrderId) REFERENCES Order_ (OrderId)
                ON DELETE SET NULL
                ON UPDATE CASCADE
 );


CREATE TABLE Portfolio (
        AccNum                          INTEGER,
        StockSymbol             CHAR(5),
        NumShares               INTEGER,
        Stop_                   VARCHAR(8) NOT NULL,
        StopPrice               FLOAT(2),
        PRIMARY KEY (AccNum, StockSymbol),
        FOREIGN KEY (AccNum) REFERENCES Account_ (AccNum)
                ON DELETE NO ACTION
                ON UPDATE CASCADE
);

CREATE TABLE ConditionalPriceHistory (
        OrderId                 INTEGER,
        CurSharePrice   FLOAT(2),
        PriceType               VARCHAR(15) NOT NULL,
        StopPrice               FLOAT(2),
        Timestamp_              TIMESTAMP WITH TIME ZONE DEFAULT CURRENT_TIMESTAMP,
        PRIMARY KEY(OrderId, PriceType, Timestamp_),
        FOREIGN KEY(OrderId) REFERENCES Order_ (OrderId)
                ON DELETE CASCADE       -- fix
                ON UPDATE CASCADE
);

CREATE TABLE StockPriceHistory (
        StockSymbol             VARCHAR(5),
        SharePrice              FLOAT(2),
        Timestamp_              TIMESTAMP WITH TIME ZONE DEFAULT CURRENT_TIMESTAMP,
        PRIMARY KEY(StockSymbol, Timestamp_),
        FOREIGN KEY(StockSymbol) REFERENCES Stock (StockSymbol)
                ON DELETE CASCADE
                ON UPDATE CASCADE
);
```

Data used for execution

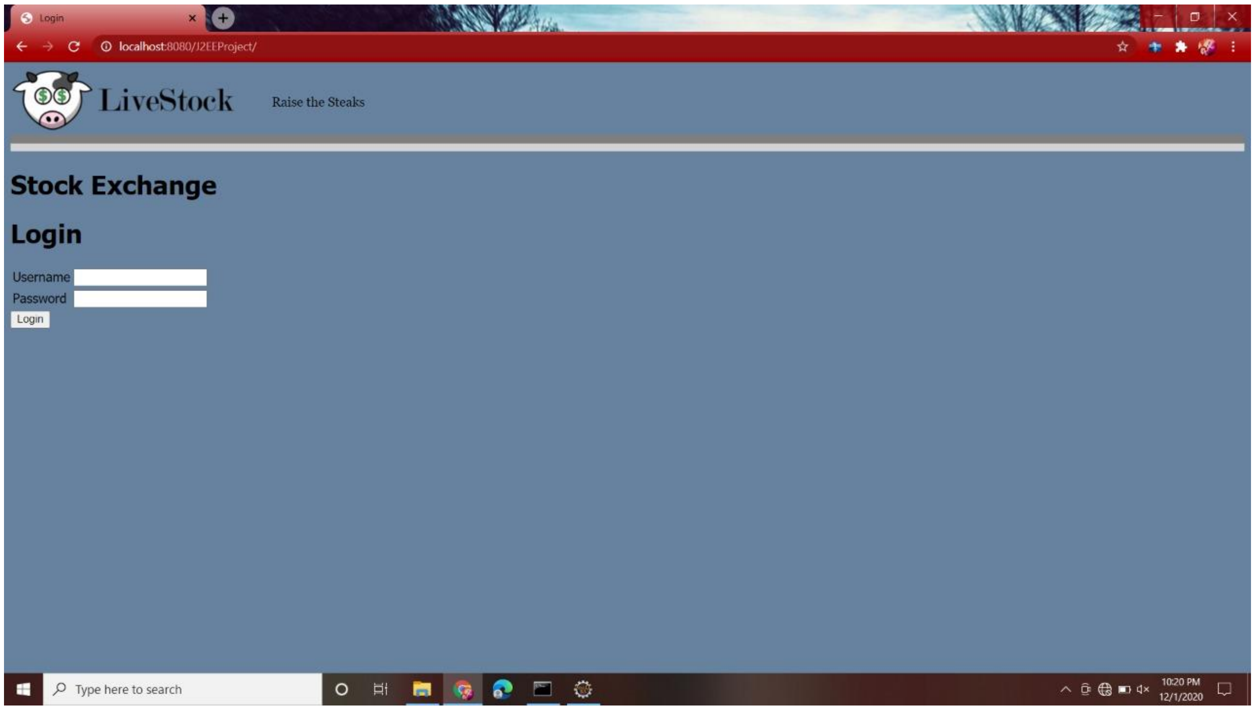
```
INSERT INTO Customer(LastName, FirstName, Address, City, State, ZipCode, Telephone, Email,
Rating)

VALUES ('Yang', 'Shang', '123 Success Street', 'Stony Brook', 'NY', '11790', '5166328959',
'syang@cs.sunysb.edu', 1);


INSERT INTO Customer(LastName, FirstName, Address, City, State, ZipCode, Telephone, Email,
Rating)

VALUES ('Du', 'Victor', '456 Fortune Road', 'Stony Brook', 'NY', '11790', '5166324360',
'vicdu@cs.sunysb.edu', 1);


INSERT INTO Customer(LastName, FirstName, Address, City, State, ZipCode, Telephone, Email,
Rating)

VALUES ('Smith', 'John', '789 Peace Blvd.', 'Los Angeles', 'CA', '93536', '3154434321',
'jsmith@ic.sunysb.edu', 1);


INSERT INTO Customer(LastName, FirstName, Address, City, State, ZipCode, Telephone, Email,
Rating)

VALUES ('Philip', 'Lewis', '135 Knowledge Lane', 'Stony Brook', 'NY', '11794', '5166668888',
'pml@cs.sunysb.edu', 1);


INSERT INTO Account_(AccCreDate, CreditCNum, CusId)

VALUES ('2006-10-01 00:00:00', '1234567812345678', 4);


INSERT INTO Account_(AccCreDate, CreditCNum, CusId)

VALUES ('2006-10-15 00:00:00', '5678123456781234', 2);


INSERT INTO Account_(AccCreDate, CreditCNum, CusId)

VALUES ('2016-10-15 00:00:00', '5432123456781234', 1);
```

```
INSERT INTO Stock (StockSymbol, StockName, StockType, SharePrice, NumAvailShares)
VALUES ('GM', 'General Motors', 'automotive', 34.23, 1000);

INSERT INTO Stock (StockSymbol, StockName, StockType, SharePrice, NumAvailShares)
VALUES ('IBM', 'IBM', 'computer', 91.43, 500);

INSERT INTO Stock (StockSymbol, StockName, StockType, SharePrice, NumAvailShares)
VALUES ('F', 'Ford', 'automotive', 9.0, 750);

INSERT INTO Portfolio (AccNum, StockSymbol, NumShares, Stop_, StopPrice)
VALUES (1, 'GM', 250, 'None', NULL);

INSERT INTO Portfolio (AccNum, StockSymbol, NumShares, Stop_, StopPrice)
VALUES (1, 'F', 100, 'None', NULL);

INSERT INTO Portfolio (AccNum, StockSymbol, NumShares, Stop_, StopPrice)
VALUES (2, 'IBM', 50, 'None', NULL);

INSERT INTO Portfolio (AccNum, StockSymbol, NumShares, Stop_, StopPrice)
VALUES (3, 'GM', 50, 'None', NULL);

INSERT INTO Employee (SSN, LastName, FirstName, Address, City, State, ZipCode, Telephone,
StartDate, HourlyRate, Position_)
VALUES ('123456789', 'Smith', 'David', '123 College Road', 'Stony Brook', 'NY', '11790',
'5162152345', '2005-11-01 00:00:00', 60, 'CusRep');

INSERT INTO Employee (SSN, LastName, FirstName, Address, City, State, ZipCode, Telephone,
StartDate, HourlyRate, Position_)
```

```
VALUES ('789123456', 'Warren', 'David', '456 Sunken Street', 'Stony Brook', 'NY', '11794',
'6316329987', '2006-02-02 00:00:00', 50, 'Manager');

INSERT INTO Order_ (StockSymbol, OrderType, NumShares, CusAccNum, Timestamp_,
PriceType, StopPrice, EmpId, Recorded)
VALUES ('F', 'Sell', 30, 1, NOW(), 'Market', NULL, '1', '0');

INSERT INTO Login (Usr, Pwd, AccType, Id)
VALUES ('CoolPerson', '2cool4school', 1, 1);

INSERT INTO Login (Usr, Pwd, AccType, Id)
VALUES ('DuVic', 'horse', 1, 2);

INSERT INTO Login (Usr, Pwd, AccType, Id)
VALUES ('Wordsmith', 'pen>sword', 1, 3);

INSERT INTO Login (Usr, Pwd, AccType, Id)
VALUES ('Clark', 'adventure', 1, 4);

INSERT INTO Login (Usr, Pwd, AccType, Id)
VALUES ('Dsmith', '12345', 2, 1);

INSERT INTO Login (Usr, Pwd, AccType, Id)
VALUES ('Boss', 'password', 3, 2);

INSERT INTO Order_ (StockSymbol, OrderType, NumShares, CusAccNum, Timestamp_,
PriceType, StopPrice, EmpId, Recorded)
```

```
VALUES ('F', 'Sell', 200, 1, NOW(), 'Trailing Stop', 5, '1', '0');

INSERT INTO Order_ (StockSymbol, OrderType, NumShares, CusAccNum, Timestamp_,
PriceType, StopPrice, EmpId, Recorded)
VALUES ('F', 'Buy', 200, 3, NOW(), 'Market', null, '1', '0');

INSERT INTO Order_ (StockSymbol, OrderType, NumShares, CusAccNum, Timestamp_,
PriceType, StopPrice, EmpId, Recorded)
VALUES ('IBM', 'Buy', 100, 1, NOW(), 'Market', null, '1', '0');

INSERT INTO Order_ (StockSymbol, OrderType, NumShares, CusAccNum, Timestamp_,
PriceType, StopPrice, EmpId, Recorded)
VALUES ('IBM', 'Sell', 25, 2, NOW(), 'Market', null, '1', '0');

INSERT INTO Order_ (StockSymbol, OrderType, NumShares, CusAccNum, Timestamp_,
PriceType, StopPrice, EmpId, Recorded)
VALUES ('GM', 'Buy', 100, 1, NOW(), 'Market', null, '1', '0');

INSERT INTO Order_ (StockSymbol, OrderType, NumShares, CusAccNum, Timestamp_,
PriceType, StopPrice, EmpId, Recorded)
VALUES ('GM', 'Sell', 25, 3, NOW(), 'Trailing Stop', 10, '1', '0');

INSERT INTO Order_ (StockSymbol, OrderType, NumShares, CusAccNum, Timestamp_,
PriceType, StopPrice, EmpId, Recorded)
VALUES ('GM', 'Sell', 25, 1, NOW(), 'Hidden Stop', 10, '1', '0');

INSERT INTO Order_ (StockSymbol, OrderType, NumShares, CusAccNum, Timestamp_,
PriceType, StopPrice, EmpId, Recorded)
VALUES ('GM', 'Sell', 25, 1, NOW(), 'Trailing Stop', 10, '1', '0');
```

Screenshots



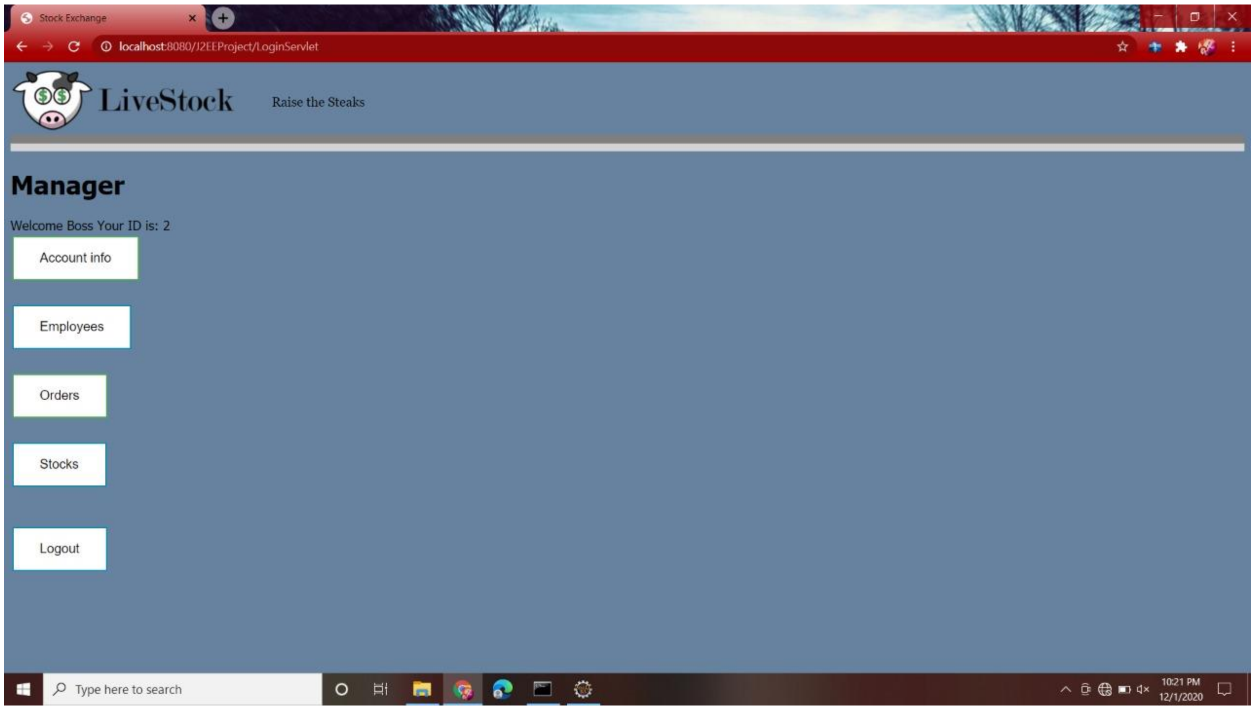
Stock Exchange

Login

Username

Password

Login



Manager

Welcome Boss Your ID is: 2

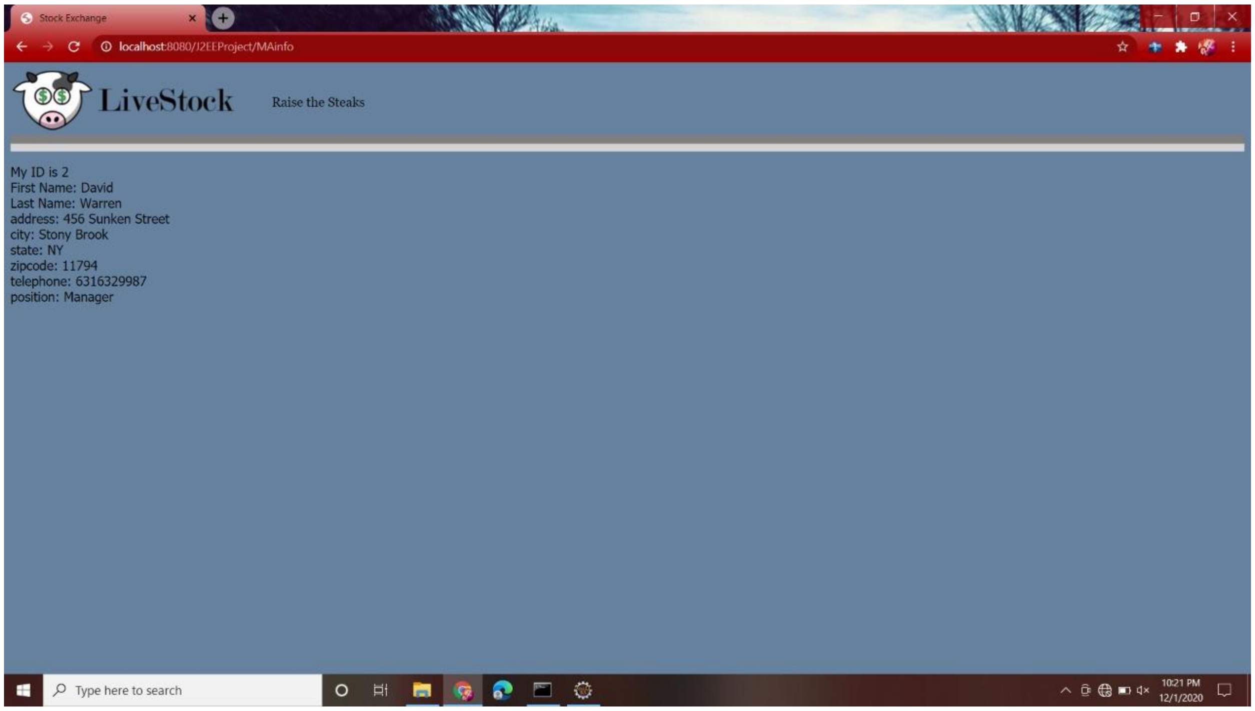
Account info

Employees

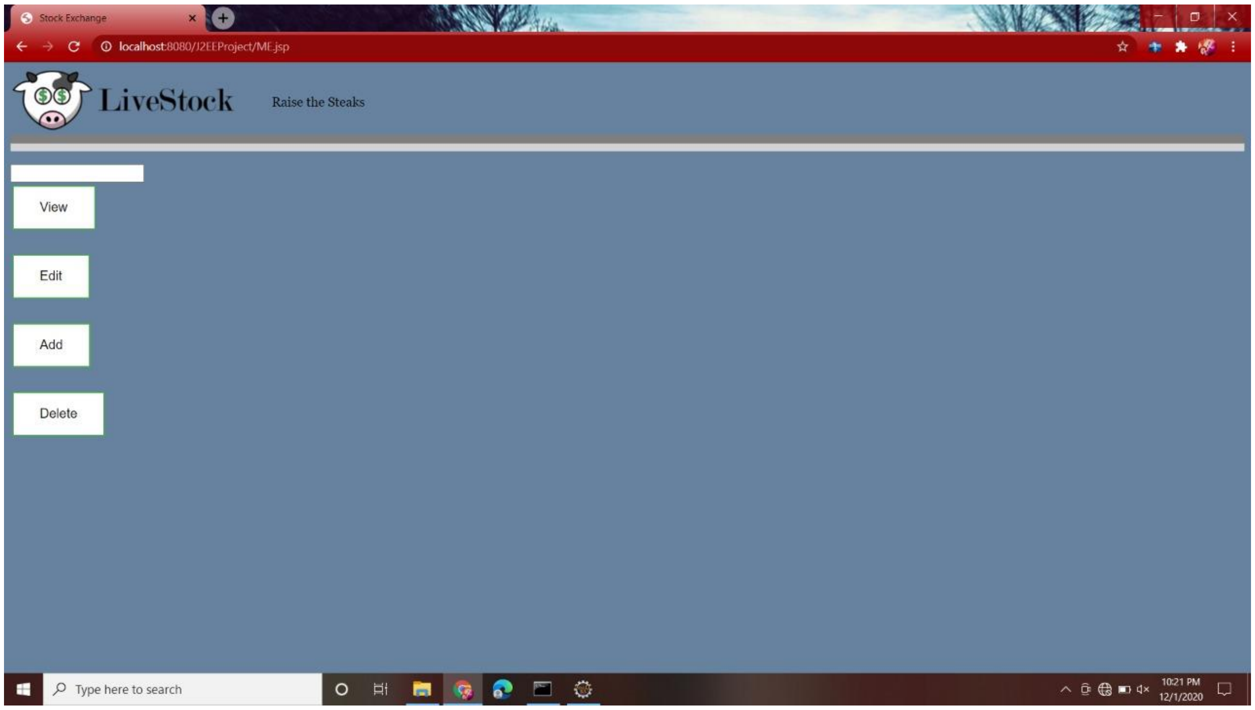
Orders

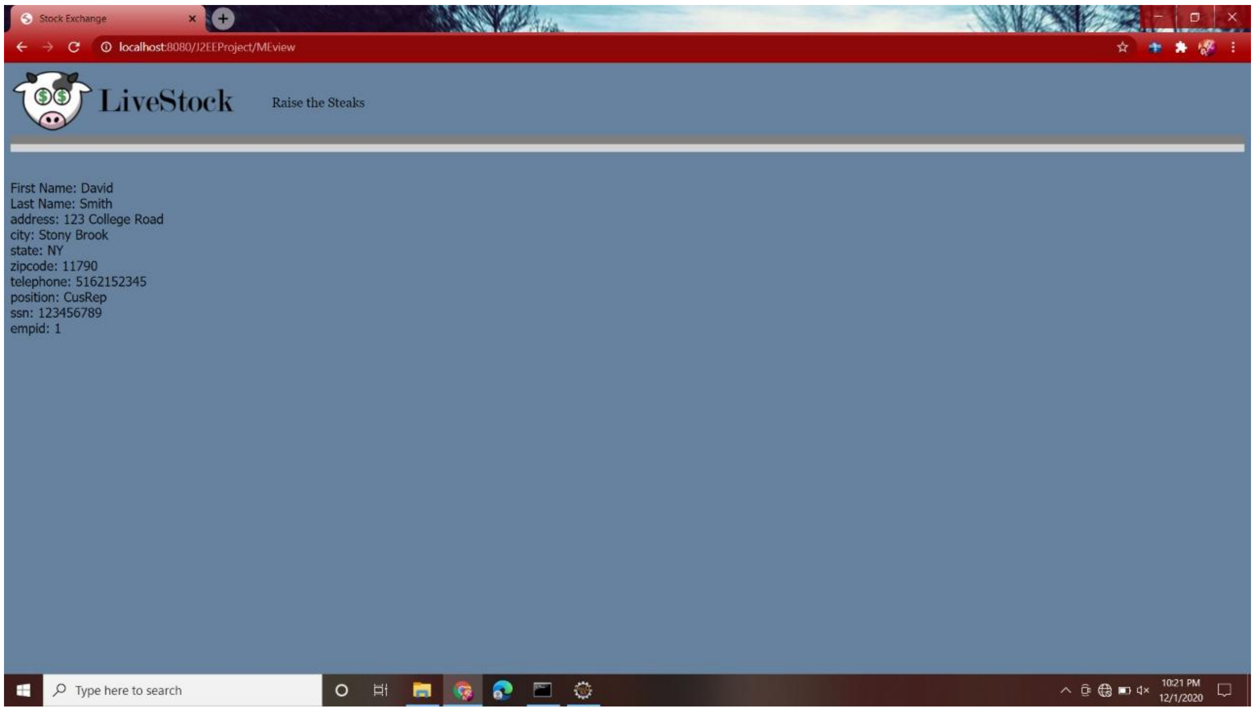
Stocks

Logout

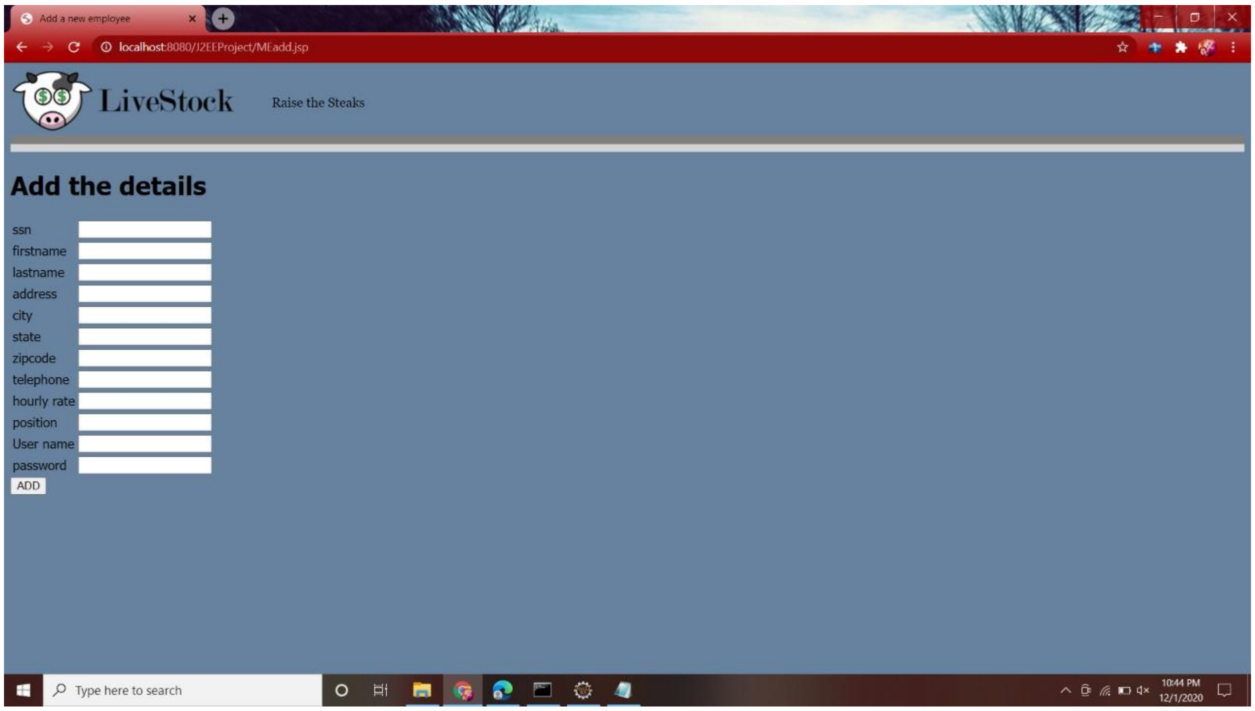


My ID is 2
First Name: David
Last Name: Warren
address: 456 Sunken Street
city: Stony Brook
state: NY
zipcode: 11794
telephone: 6316329987
position: Manager





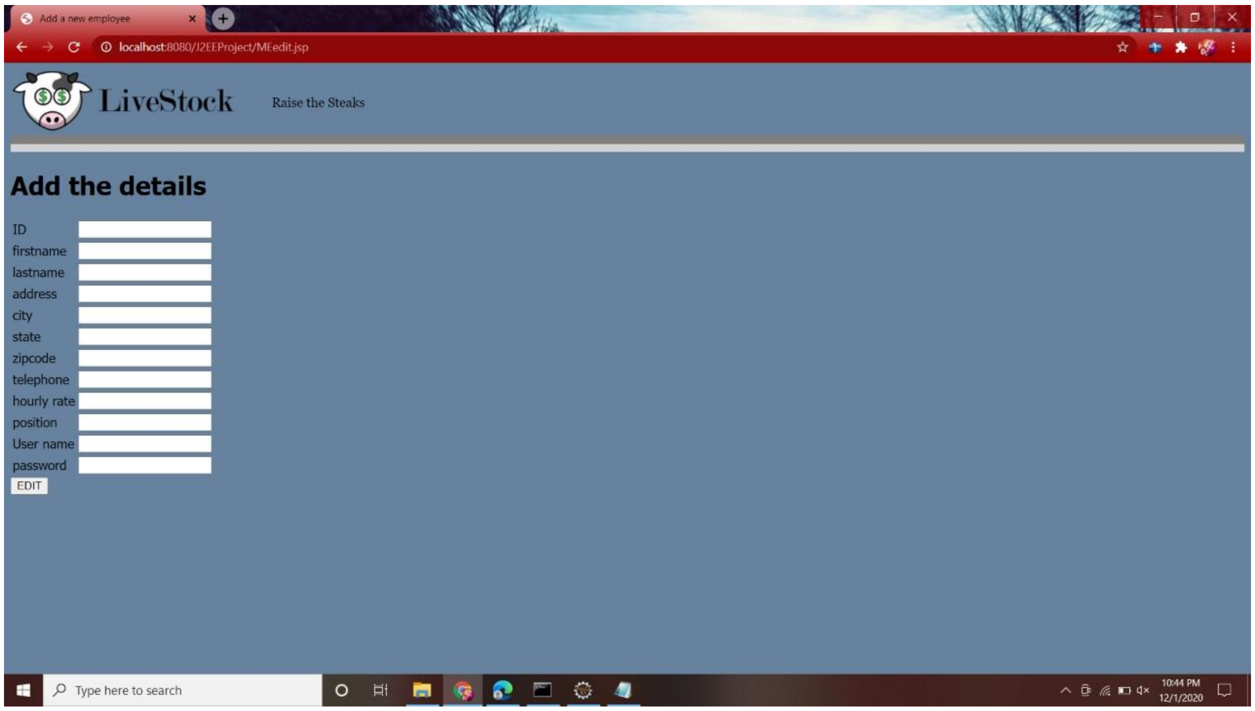
First Name: David
Last Name: Smith
address: 123 College Road
city: Stony Brook
state: NY
zipcode: 11790
telephone: 5162152345
position: CusRep
ssn: 123456789
empid: 1



Add the details

ssn	
firstname	
lastname	
address	
city	
state	
zipcode	
telephone	
hourly rate	
position	
User name	
password	

ADD


Add a new employee

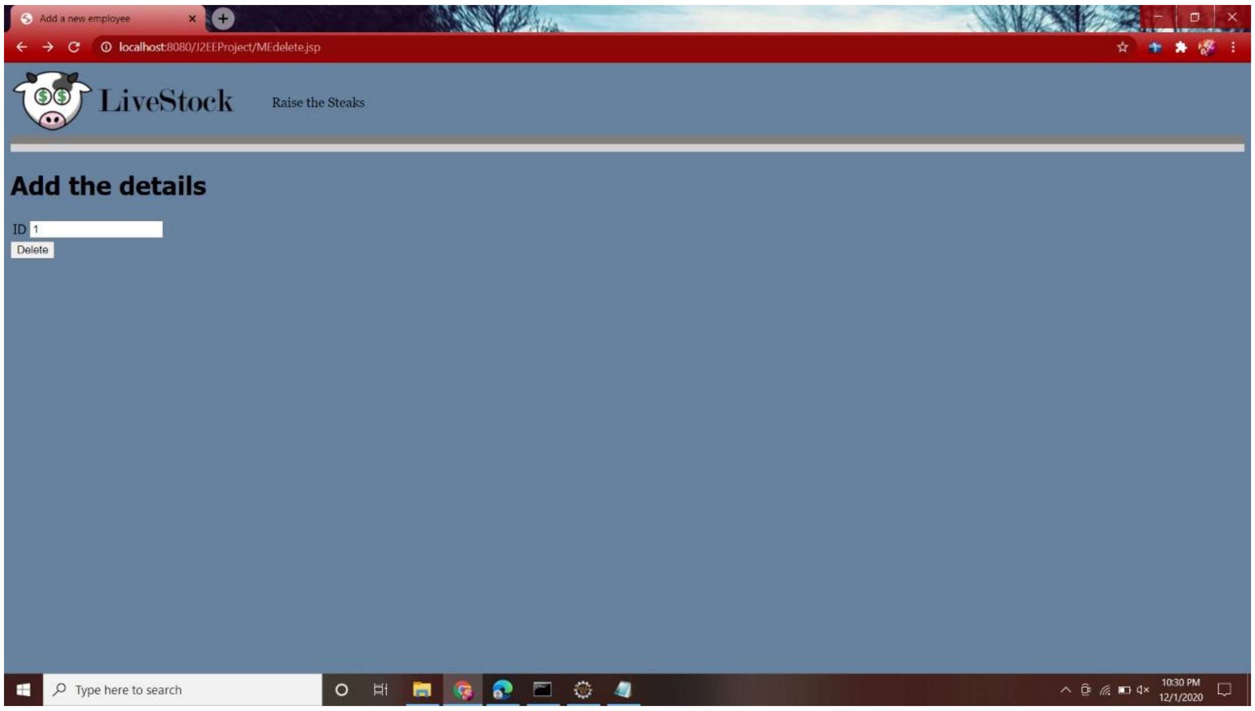
localhost:8080/J2EEProject/MEedit.jsp

LiveStock Raise the Steaks

Add the details

ID	
firstname	
lastname	
address	
city	
state	
zipcode	
telephone	
hourly rate	
position	
User name	
password	

EDIT


Add a new employee

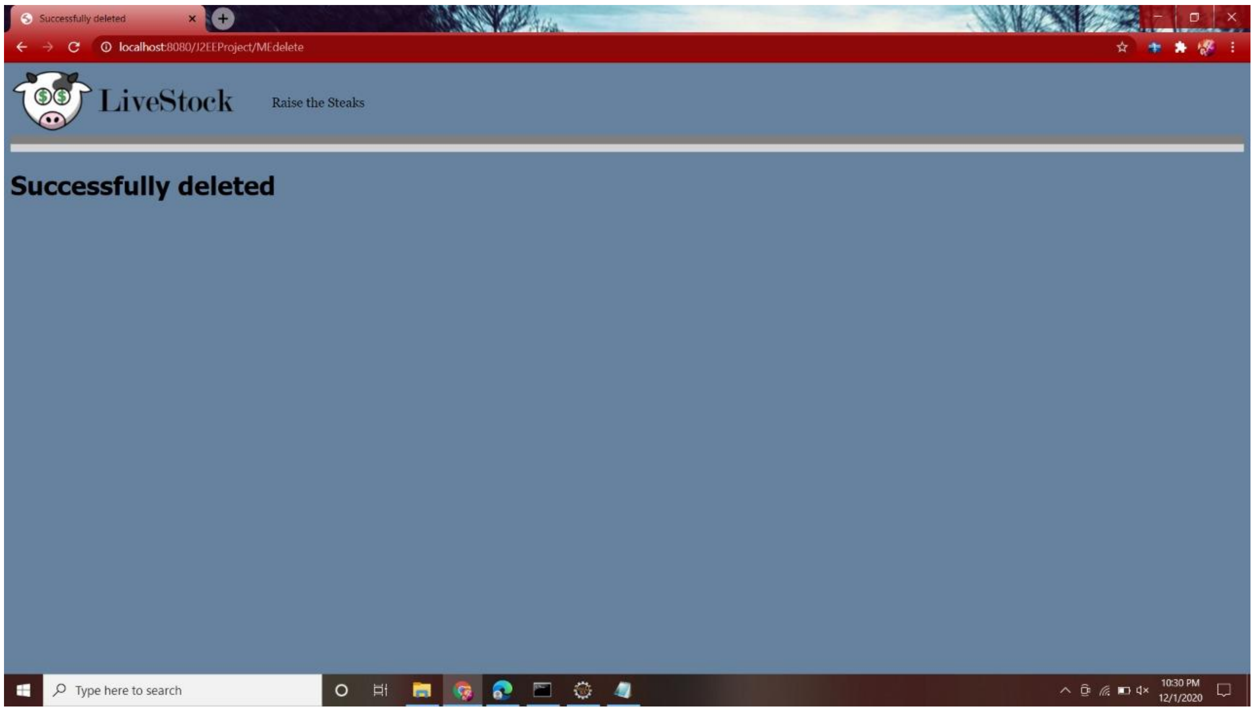
localhost:8080/J2EEProject/MEdelete.jsp

LiveStock Raise the Steaks

Add the details

ID 1

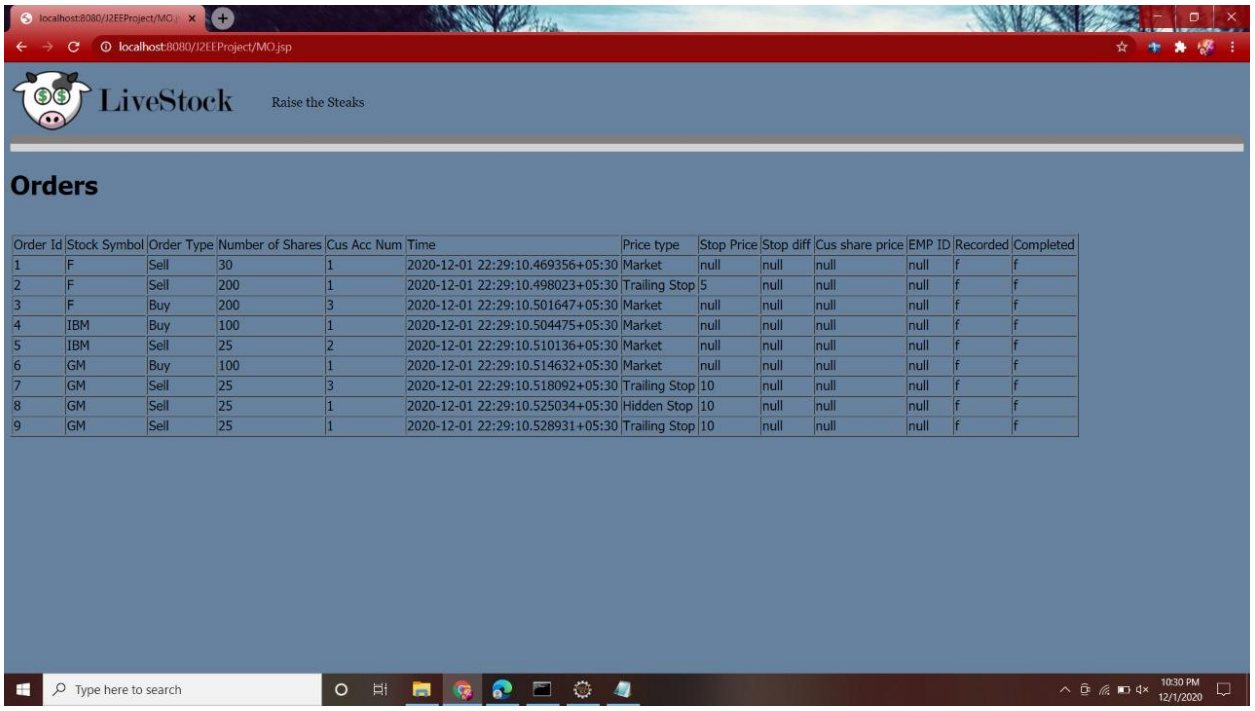
Delete


Successfully deleted

localhost:8080/J2EEProject/MEdelete

LiveStock Raise the Steaks

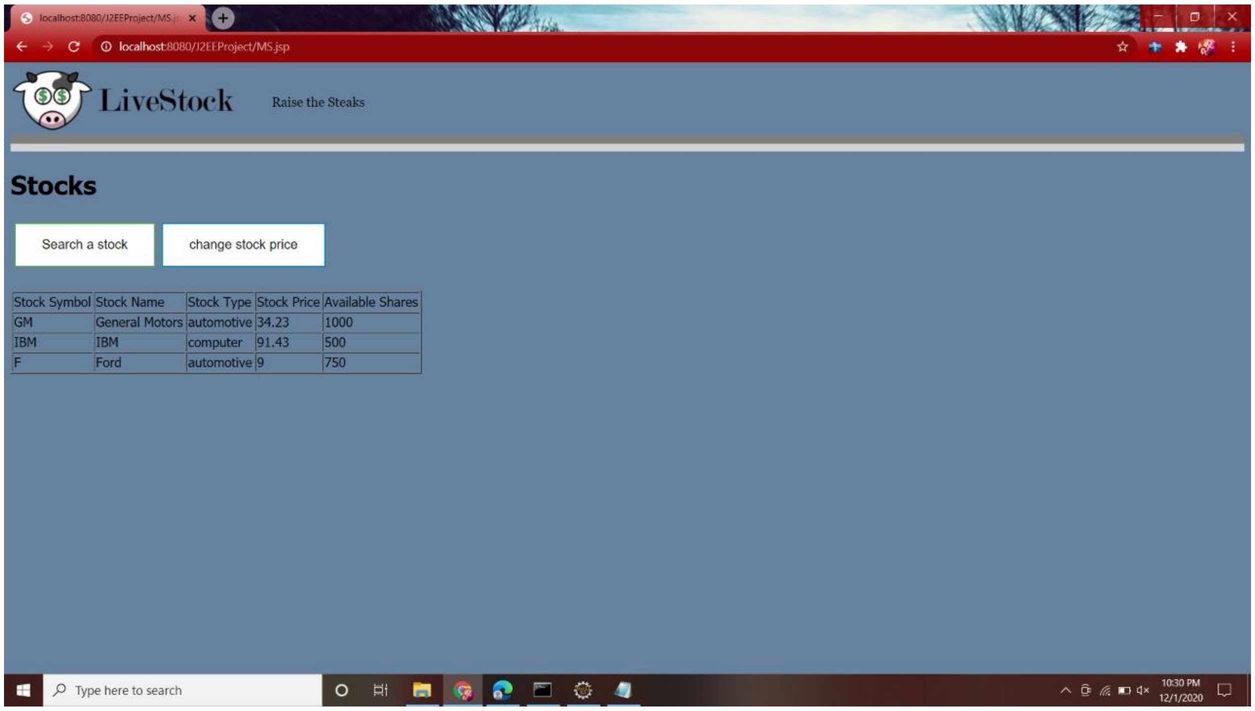
Successfully deleted


localhost:8080/J2EEProject/MO.jsp

LiveStock Raise the Steaks

Orders

Order Id	Stock Symbol	Order Type	Number of Shares	Cus Acc Num	Time	Price type	Stop Price	Stop diff	Cus share price	EMP ID	Recorded	Completed
1	F	Sell	30	1	2020-12-01 22:29:10.469356+05:30	Market	null	null	null	null	f	f
2	F	Sell	200	1	2020-12-01 22:29:10.498023+05:30	Trailing Stop	5	null	null	null	f	f
3	F	Buy	200	3	2020-12-01 22:29:10.501647+05:30	Market	null	null	null	null	f	f
4	IBM	Buy	100	1	2020-12-01 22:29:10.504475+05:30	Market	null	null	null	null	f	f
5	IBM	Sell	25	2	2020-12-01 22:29:10.510136+05:30	Market	null	null	null	null	f	f
6	GM	Buy	100	1	2020-12-01 22:29:10.514632+05:30	Market	null	null	null	null	f	f
7	GM	Sell	25	3	2020-12-01 22:29:10.518092+05:30	Trailing Stop	10	null	null	null	f	f
8	GM	Sell	25	1	2020-12-01 22:29:10.525034+05:30	Hidden Stop	10	null	null	null	f	f
9	GM	Sell	25	1	2020-12-01 22:29:10.528931+05:30	Trailing Stop	10	null	null	null	f	f



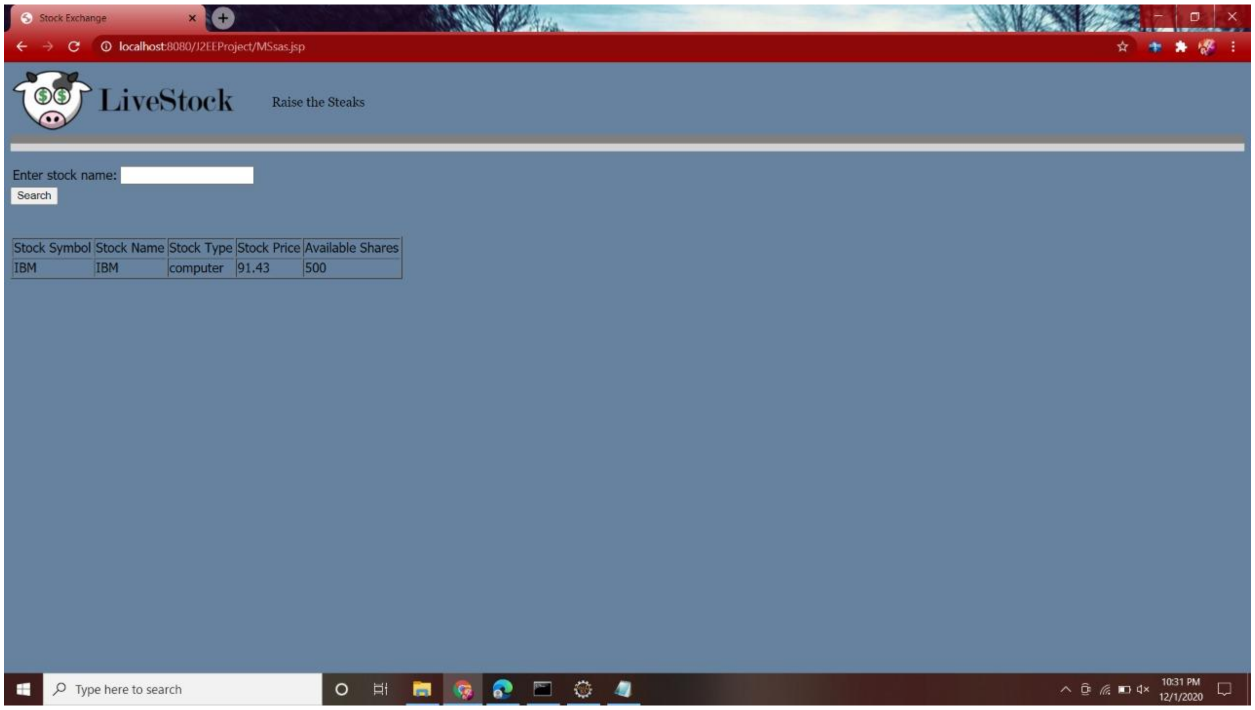
localhost:8080/J2EEProject/MS.jsp

LiveStock Raise the Steaks

Stocks

Search a stock | change stock price

Stock Symbol	Stock Name	Stock Type	Stock Price	Available Shares
GM	General Motors	automotive	34.23	1000
IBM	IBM	computer	91.43	500
F	Ford	automotive	9	750



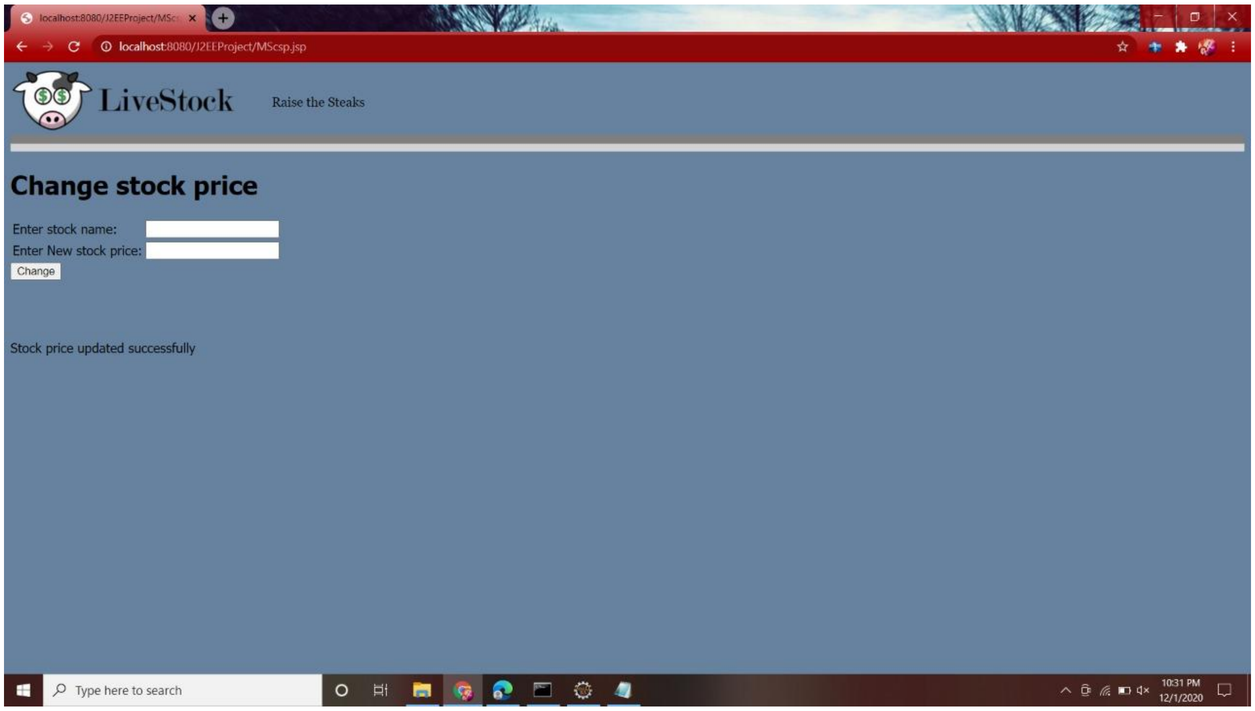
localhost:8080/J2EEProject/MSsas.jsp

LiveStock Raise the Steaks

Enter stock name:

Search

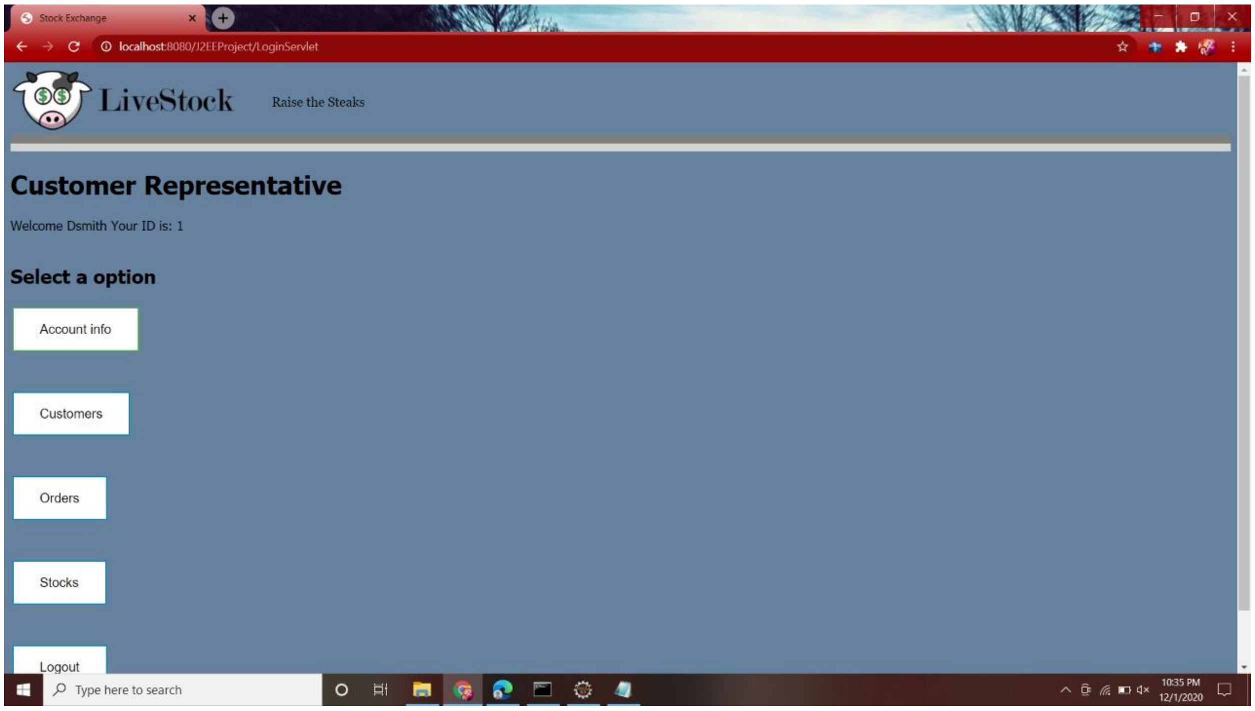
Stock Symbol	Stock Name	Stock Type	Stock Price	Available Shares
IBM	IBM	computer	91.43	500



Change stock price

Enter stock name:
Enter New stock price:
Change

Stock price updated successfully



Customer Representative

Welcome Dsmith Your ID is: 1

Select a option

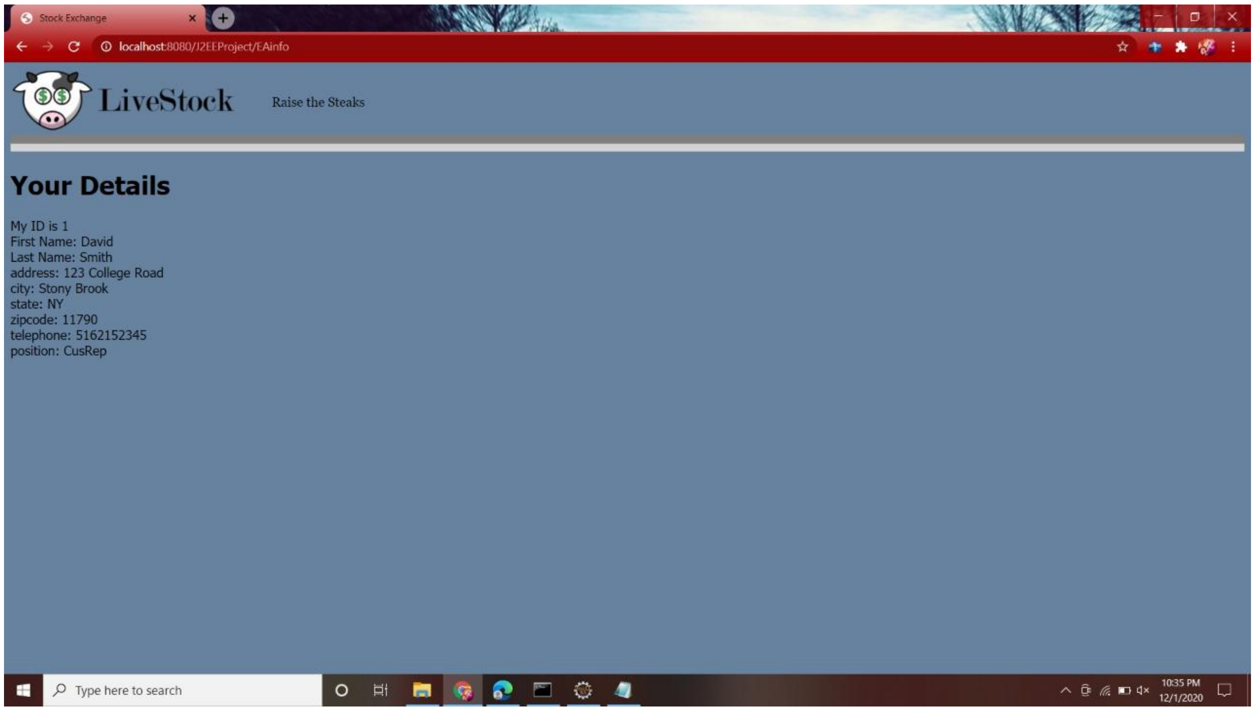
Account info

Customers

Orders

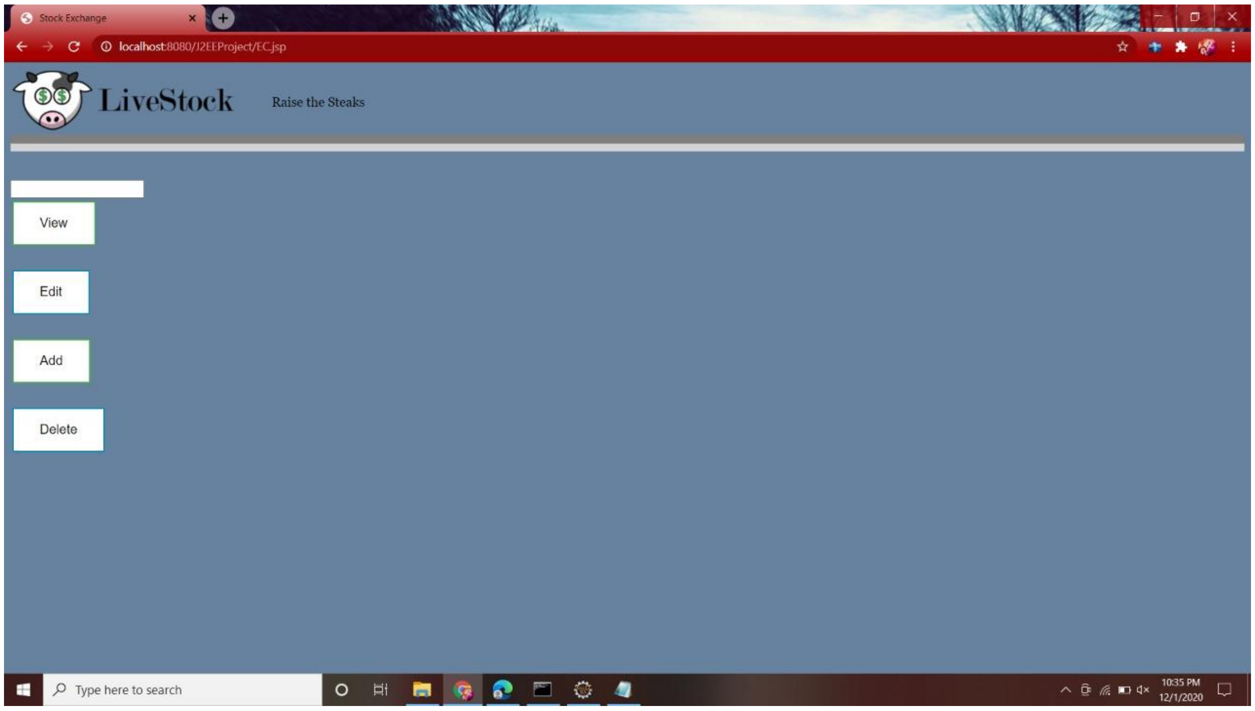
Stocks

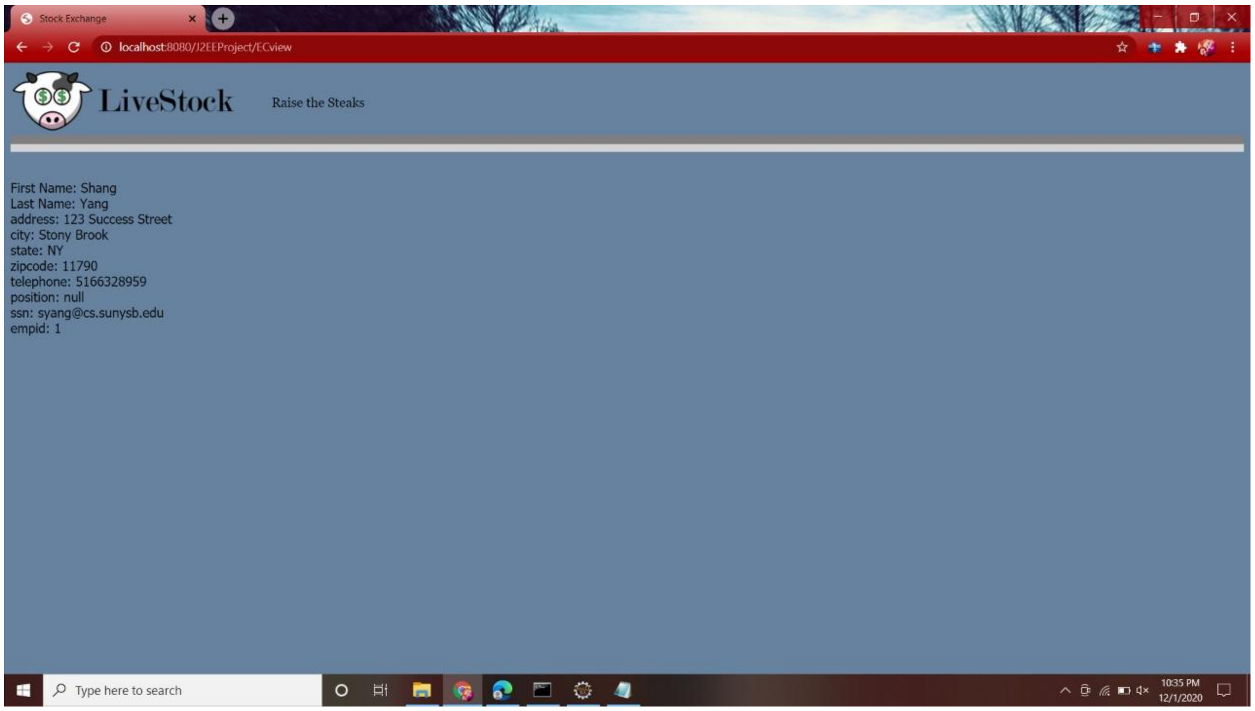
Logout



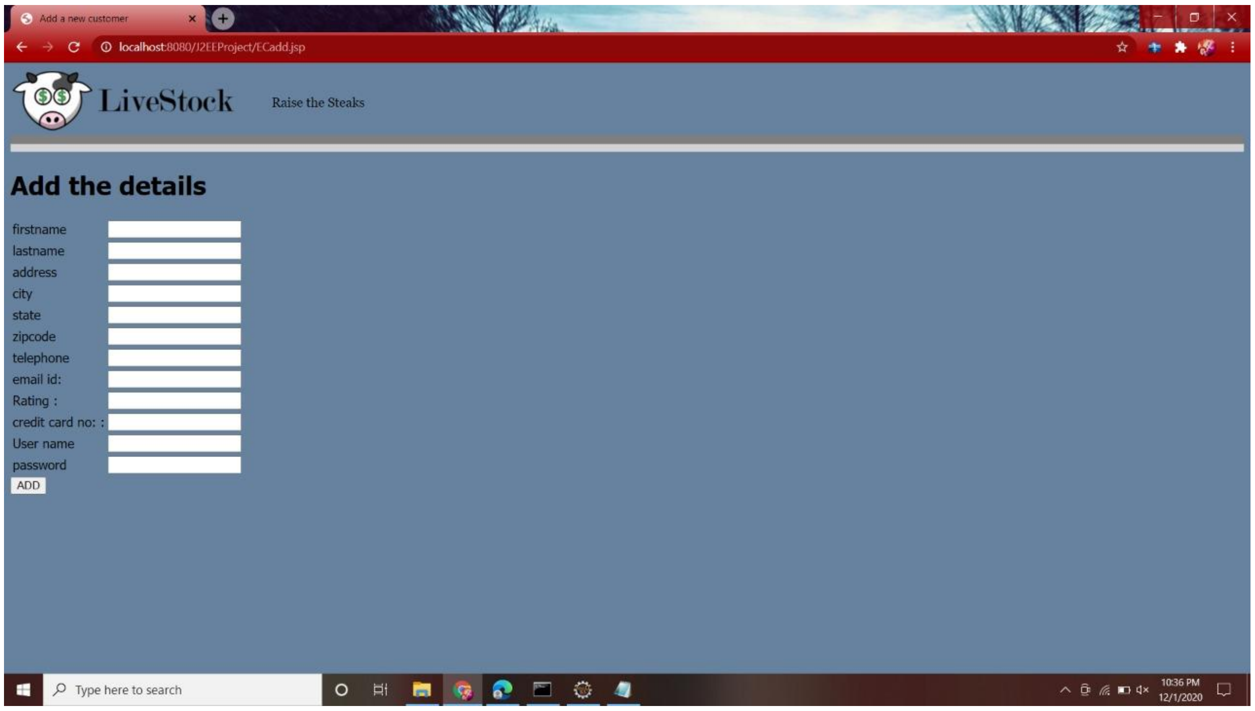
Your Details

My ID is 1
First Name: David
Last Name: Smith
address: 123 College Road
city: Stony Brook
state: NY
zipcode: 11790
telephone: 5162152345
position: CusRep



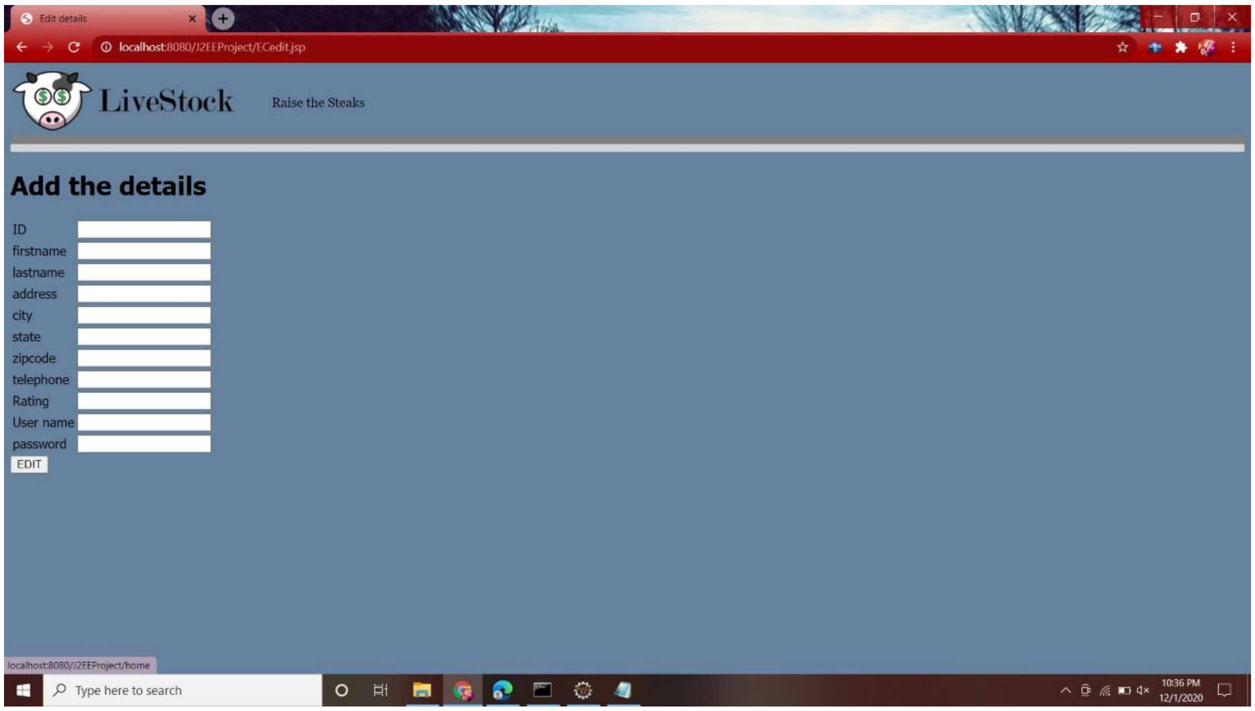


First Name: Shang
Last Name: Yang
address: 123 Success Street
city: Stony Brook
state: NY
zipcode: 11790
telephone: 5166328959
position: null
ssn: syang@cs.sunysb.edu
empid: 1



Add the details

firstname
lastname
address
city
state
zipcode
telephone
email id:
Rating :
credit card no: :
User name
password
ADD


Edit details

localhost:8080/J2EEProject/ECedit.jsp

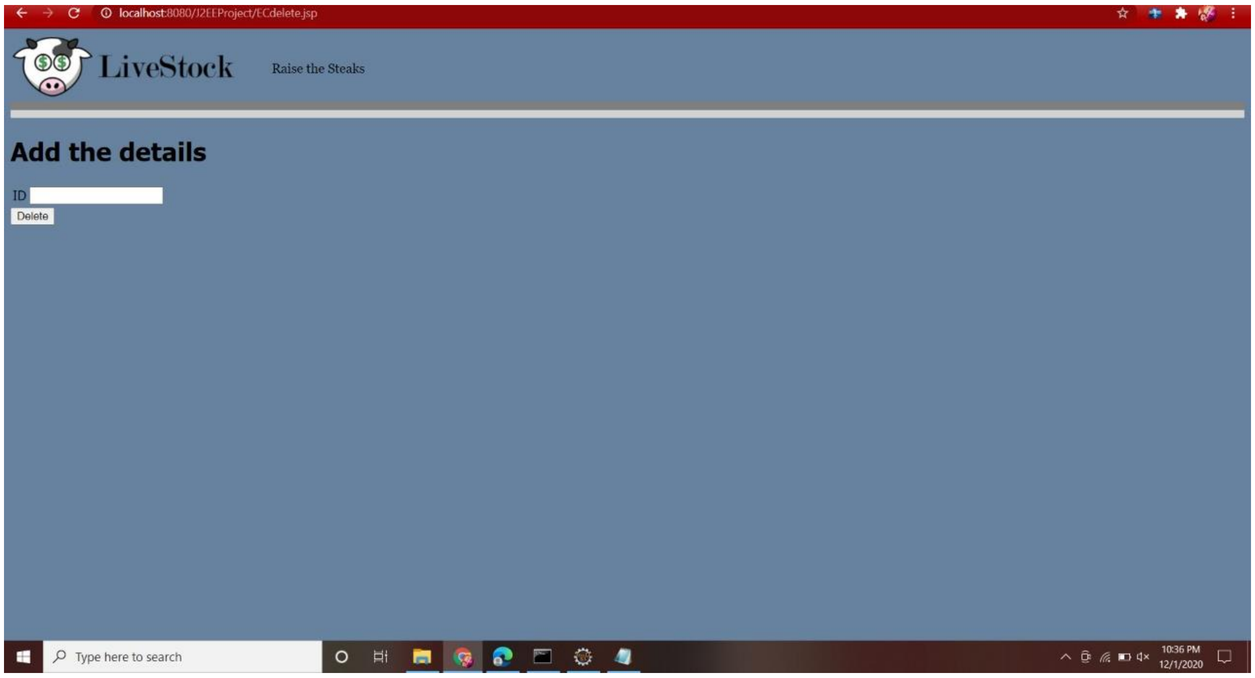
LiveStock Raise the Steaks

Add the details

ID	
firstname	
lastname	
address	
city	
state	
zipcode	
telephone	
Rating	
User name	
password	

EDIT

localhost:8080/J2EEProject/home

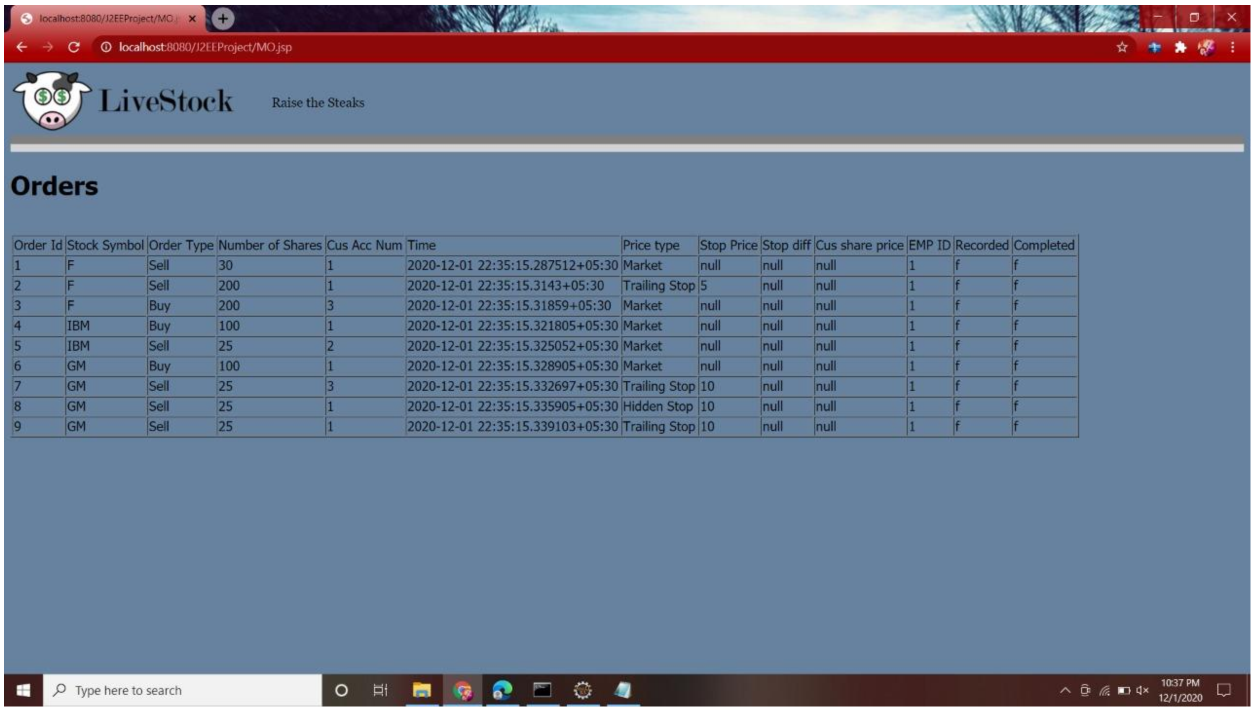

localhost:8080/J2EEProject/ECdelete.jsp

LiveStock Raise the Steaks

Add the details

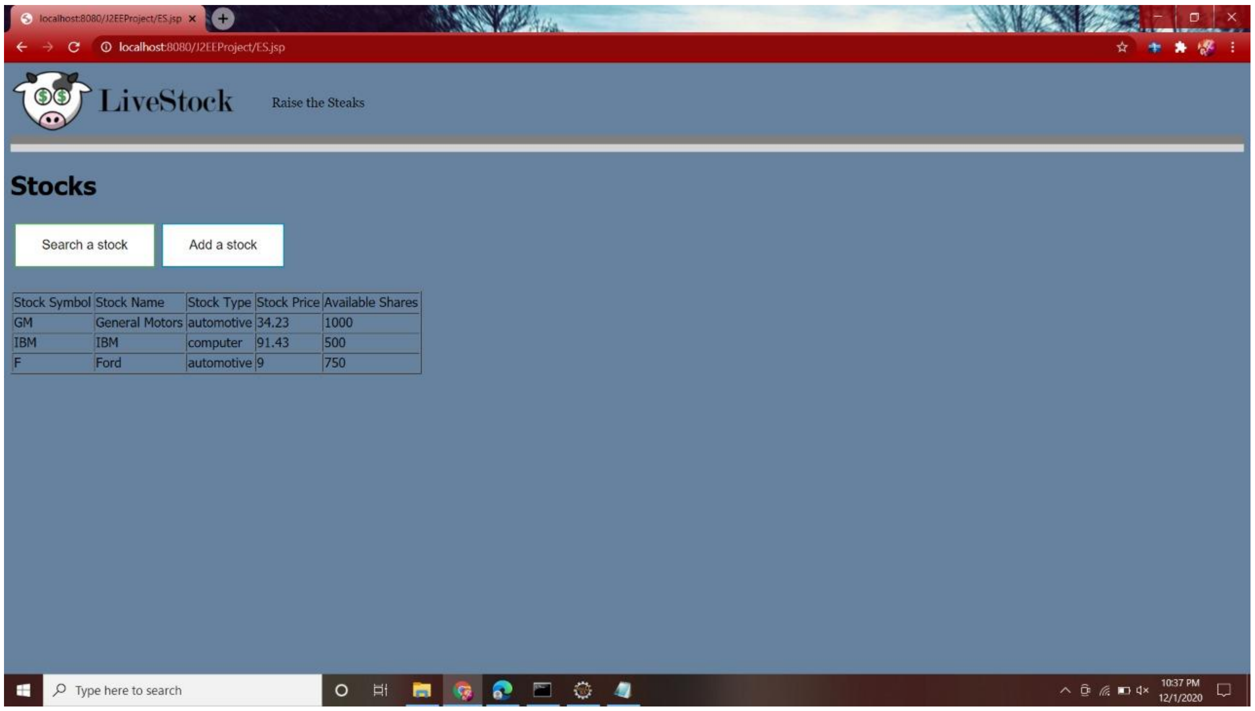
ID

Delete



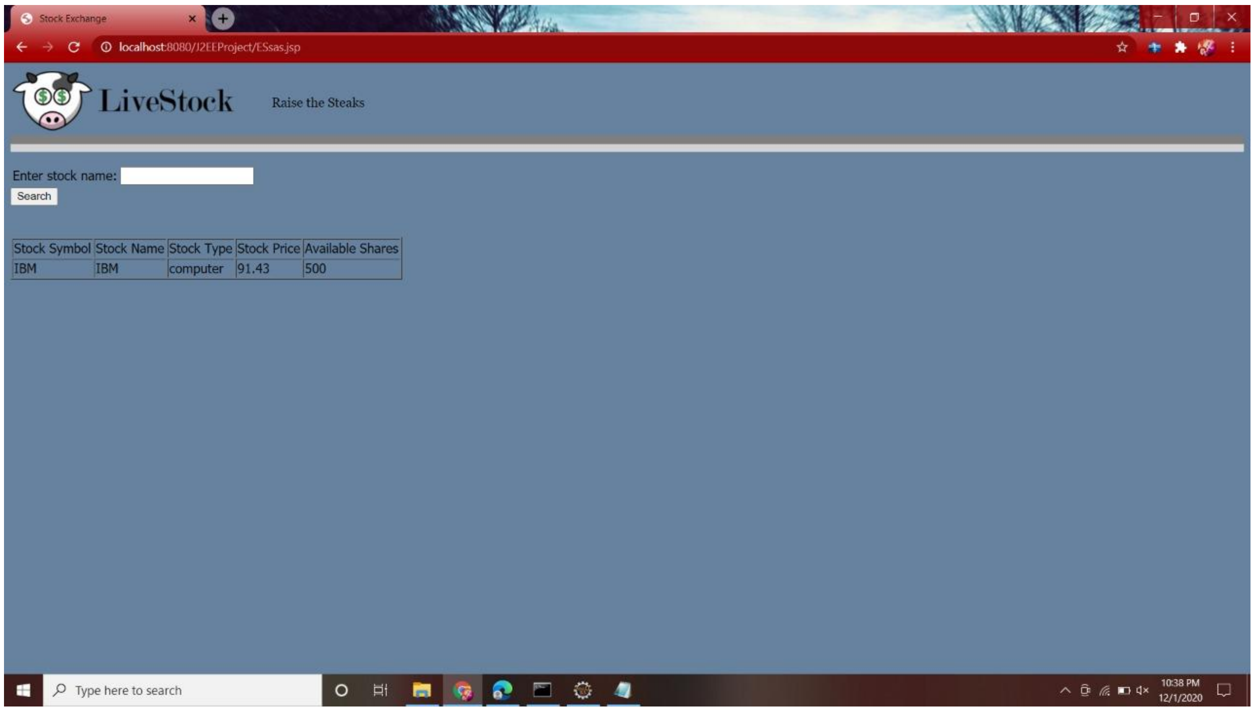
Orders

Order Id	Stock Symbol	Order Type	Number of Shares	Cus Acc Num	Time	Price type	Stop Price	Stop diff	Cus share price	EMP ID	Recorded	Completed
1	F	Sell	30	1	2020-12-01 22:35:15.287512+05:30	Market	null	null	null	1	f	f
2	F	Sell	200	1	2020-12-01 22:35:15.3143+05:30	Trailing Stop	5	null	null	1	f	f
3	F	Buy	200	3	2020-12-01 22:35:15.31859+05:30	Market	null	null	null	1	f	f
4	IBM	Buy	100	1	2020-12-01 22:35:15.321805+05:30	Market	null	null	null	1	f	f
5	IBM	Sell	25	2	2020-12-01 22:35:15.325052+05:30	Market	null	null	null	1	f	f
6	GM	Buy	100	1	2020-12-01 22:35:15.328905+05:30	Market	null	null	null	1	f	f
7	GM	Sell	25	3	2020-12-01 22:35:15.332697+05:30	Trailing Stop	10	null	null	1	f	f
8	GM	Sell	25	1	2020-12-01 22:35:15.335905+05:30	Hidden Stop	10	null	null	1	f	f
9	GM	Sell	25	1	2020-12-01 22:35:15.339103+05:30	Trailing Stop	10	null	null	1	f	f



Stocks

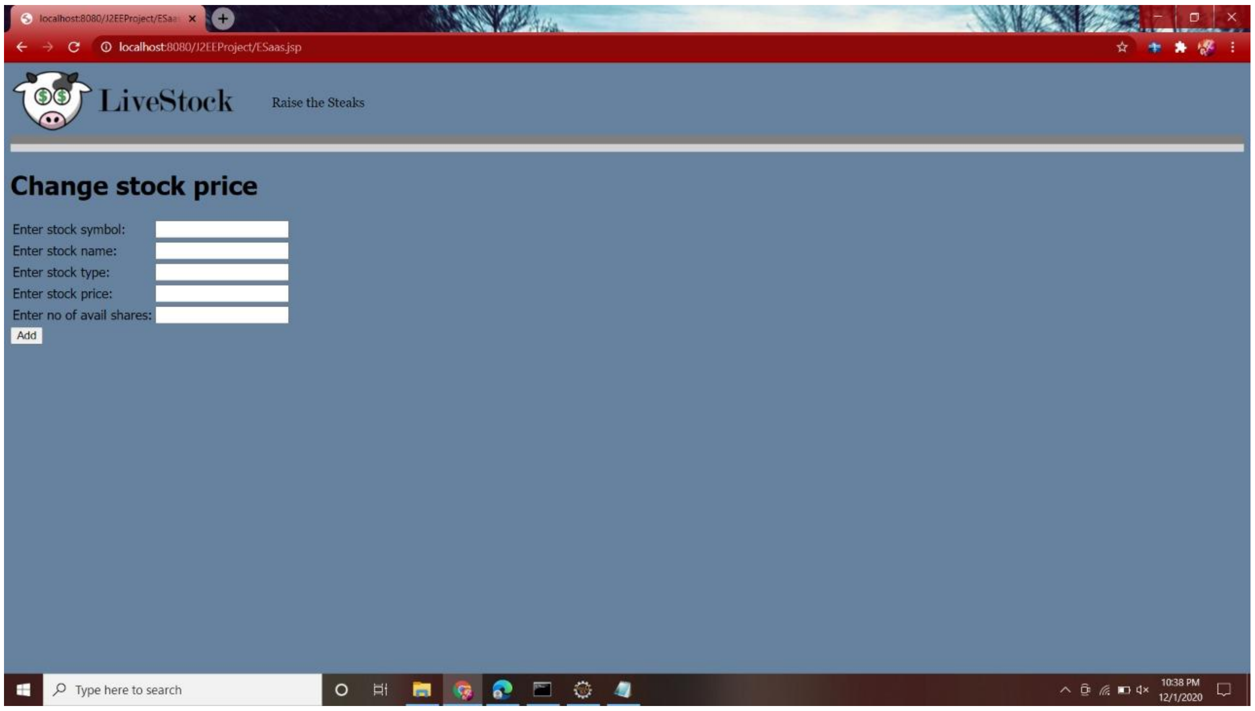
Stock Symbol	Stock Name	Stock Type	Stock Price	Available Shares
GM	General Motors	automotive	34.23	1000
IBM	IBM	computer	91.43	500
F	Ford	automotive	9	750



Enter stock name:

Search

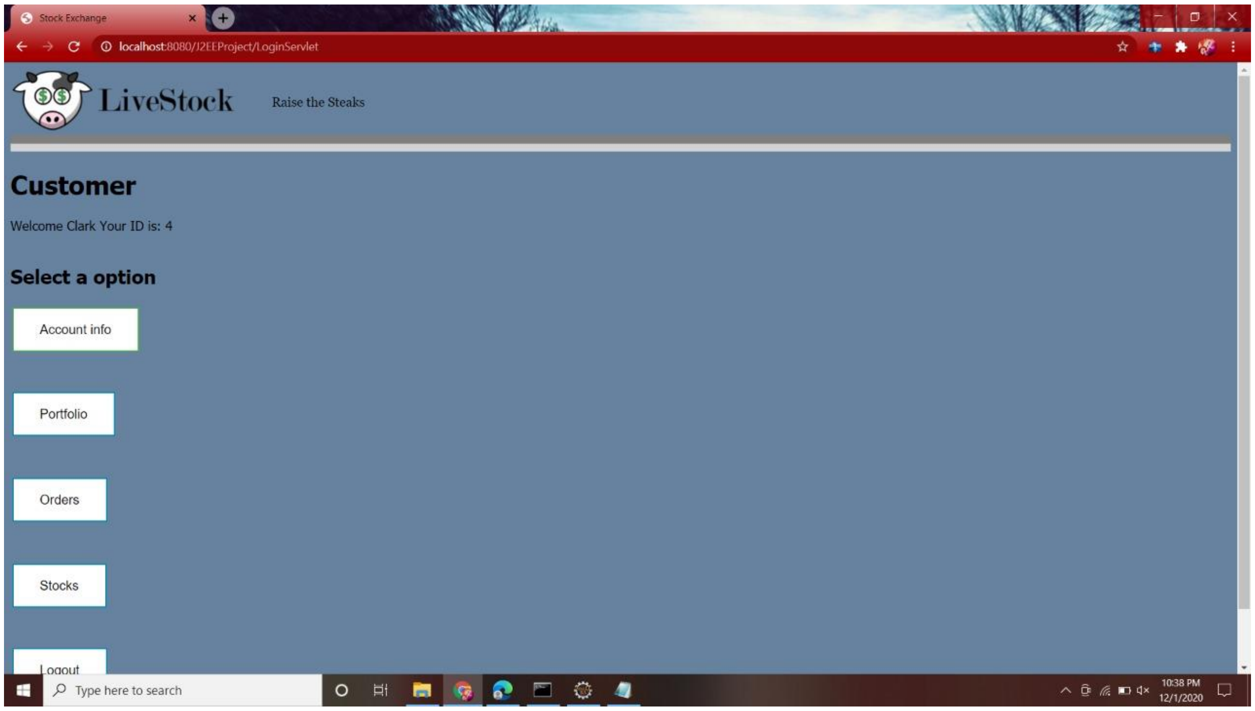
Stock Symbol	Stock Name	Stock Type	Stock Price	Available Shares
IBM	IBM	computer	91.43	500


localhost:8080/J2EEProject/ESas
localhost:8080/J2EEProject/ESaas.jsp
LiveStock
Raise the Steaks
Type here to search
10:38 PM
12/1/2020

Change stock price

Enter stock symbol:
Enter stock name:
Enter stock type:
Enter stock price:
Enter no of avail shares:

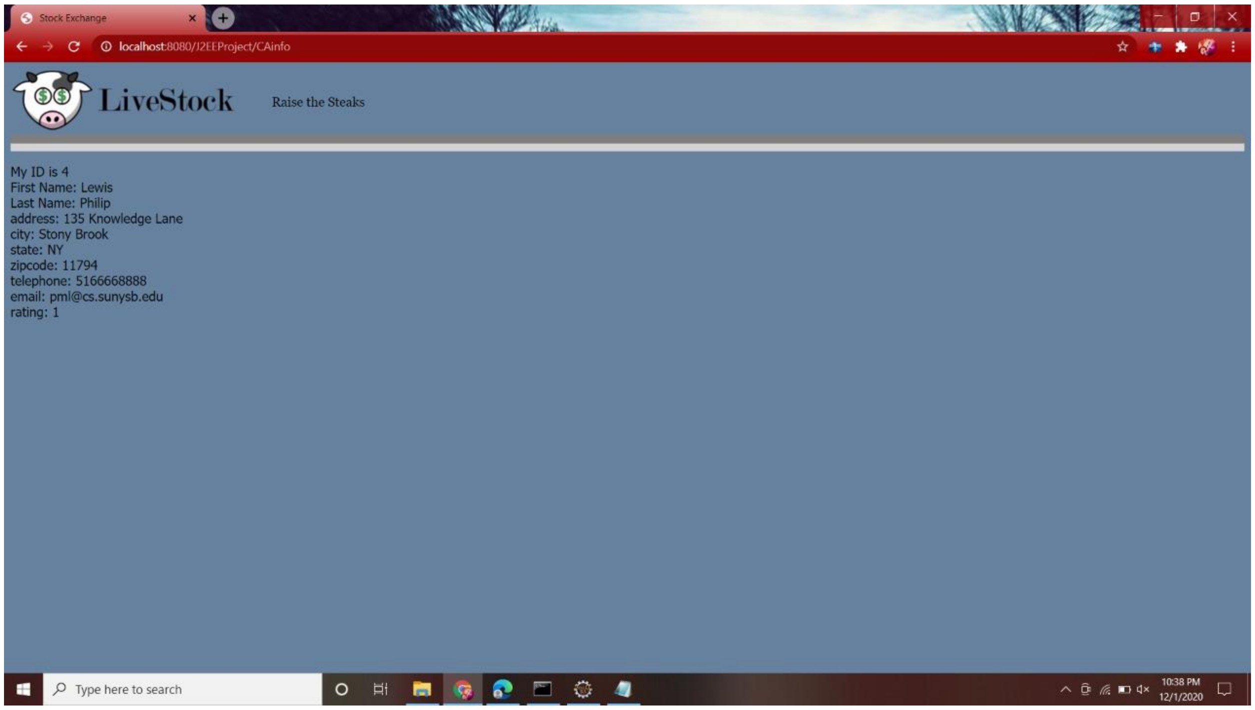
Add



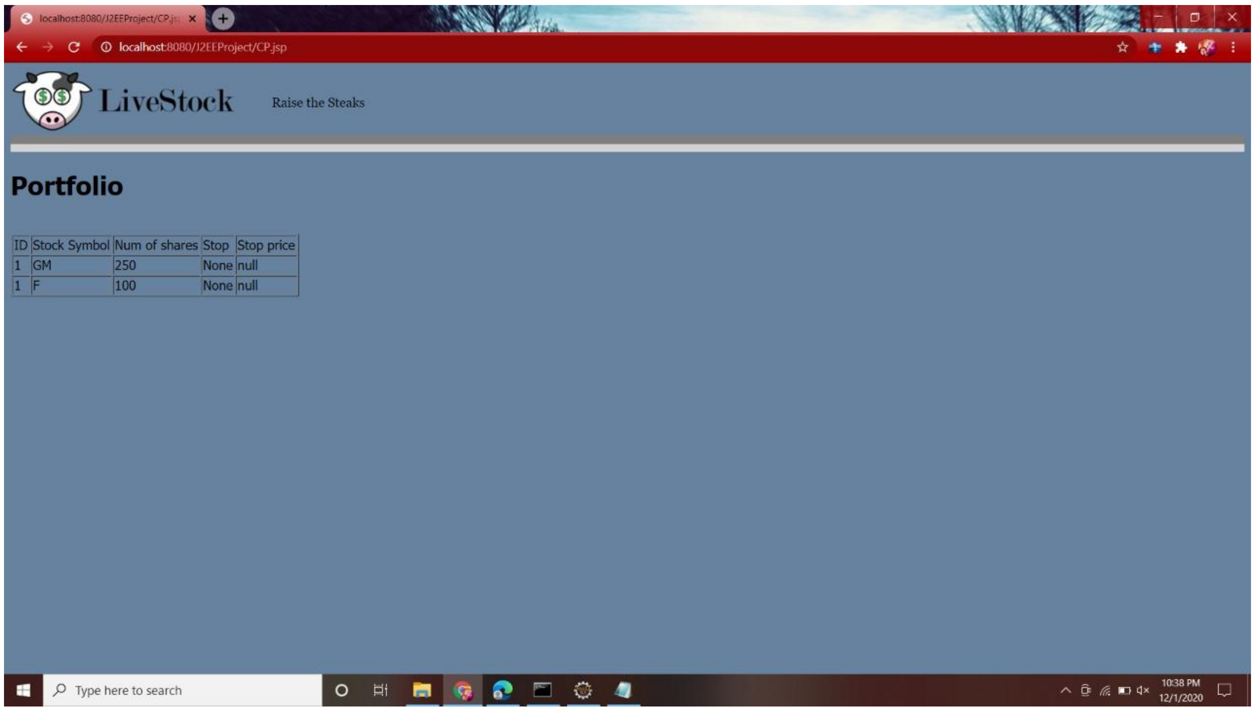
Customer

Welcome Clark Your ID is: 4

Select a option

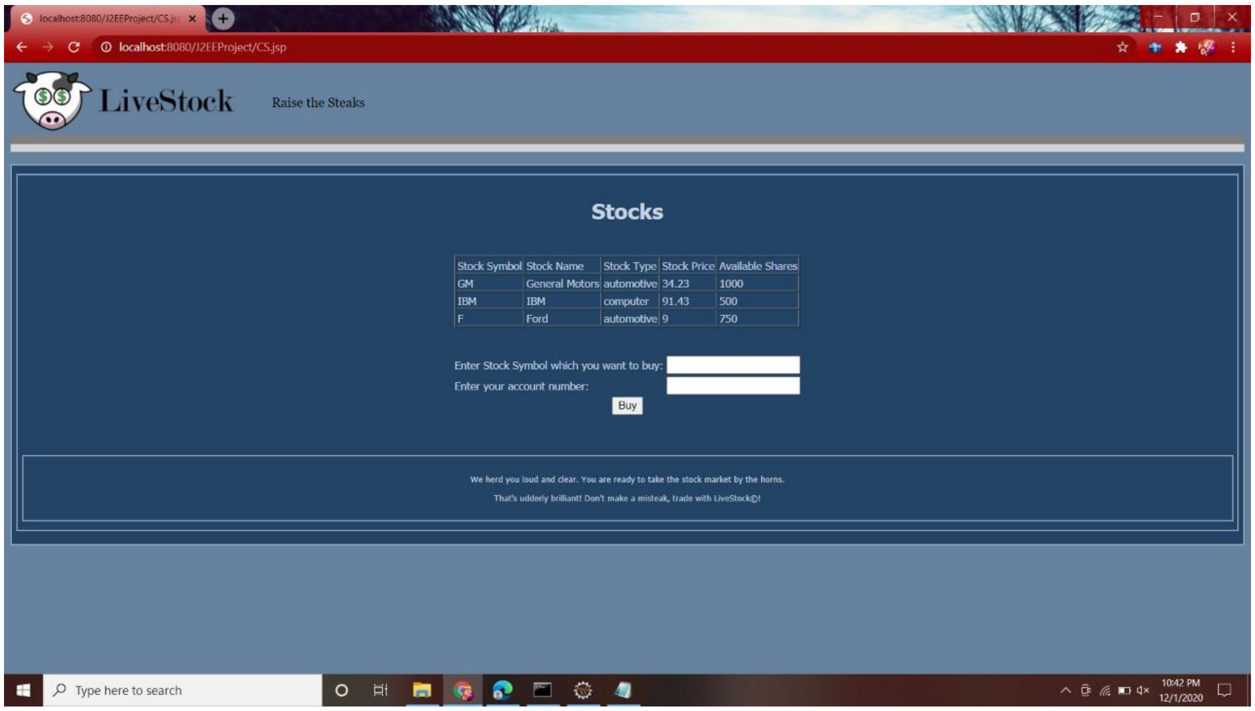


My ID is 4
First Name: Lewis
Last Name: Philip
address: 135 Knowledge Lane
city: Stony Brook
state: NY
zipcode: 11794
telephone: 5166668888
email: pml@cs.sunysb.edu
rating: 1



Portfolio

ID	Stock Symbol	Num of shares	Stop	Stop price
1	GM	250	None	null
1	F	100	None	null



Stocks

Stock Symbol	Stock Name	Stock Type	Stock Price	Available Shares
GM	General Motors	automotive	34.23	1000
IBM	IBM	computer	91.43	500
F	Ford	automotive	9	750

Enter Stock Symbol which you want to buy:
Enter your account number:

We herd you loud and clear. You are ready to take the stock market by the horns.

That's udderly brilliant! Don't make a misteak, trade with LiveStock©!

Deviations from design document

We added an option for customer representative where he can add stocks which can be purchased by customer. We added help options for every interface to make our stock trading system user friendly. Register as customer and employee was removed, because registering as customer should be done by customer representative and registering as customer representative should be done by manager, so we added those options there. We deviated from design document to improve our system which is not too much.

Work distribution among group members

Login interface and Customer interface done by Anubola Sai Abhinay. Manager interface done by Mohan Chandrakanth. Customer Representative interface done by Chilaka Avinash. And we each proofread and edited each other's writing.

Conclusion

In conclusion, a database is a far more efficient mechanism to store and organize data than spreadsheets, it allows for a centralized facility that can easily be modified and quickly shared among multiple users. Having a web based front end removes the requirement of users having to understand and use a database directly, and allows users to connect from anywhere with an internet connection and a basic web browser. It also allows the possibility of queries to obtain information for various surveys. Due to the number of customers are trading stocks, customer representatives are modifying customer data and modifying stocks and managers are modifying customer representative data and setting prices of stocks it is an ideal use for such a system.