

TRADICTION
CSC648 SOFTWARE ENGINEERING
SPRING 2020
TEAM 01

Milestone	Date
M4 Version V1	05/10/2020
M3 Version V1	04/23/2020
M2 Version V2	04/10/2020
M2 Version V1	03/26/2020
M1 Version V2	03/09/2020
M1 Version V1	03/05/2020

Team members:

SFSU email

Brian Lim – Team lead

blim6@mail.sfsu.edu

May Naing – Front-end Lead

mnaing1@mail.sfsu.edu

Sanjay Mirani – Back-end Lead

smirani1@mail.sfsu.edu

Dibs Bharati – GitHub Master

dbharati@mail.sfsu.edu

Zike Xian – Database master 1

zxian@mail.sfsu.edu

Benjamin Fletes – Database Master 2

bfletes1@mail.sfsu.edu

Table of Contents

Product Summary	4
Usability Test Plan	9
Test Objectives	9
Test Description	9
System Setup	9
Starting Point	9
Intended Users	9
URL of the system	9
What is Measured	9
Usability Task Description	10
Sentimental Twitter Analysis	10
Add Stocks into Watchlist (Registered Traders)	10
Filter Stock Information Alphabetically (A-Z) by Name	11
Search for Stock Information	11
Purchase Stocks (Registered Traders)	12
Usability Test Table	12
Questionnaire	13
QA Test Plan	14
Test objectives	14
HW and SW setup	14
Passwords shall be encrypted before storing in the database.	14
QA test plan	14
Users payment information shall be encrypted before storing in the database.	15
QA test plan	15
New registrations shall be audited by the administrator.	16
QA test plan	16
New registrations shall be approved by the administrator.	16
QA test plan	16
Application shall be able to retrieve information from the database and react in a timely manner.	17
QA test plan	17
Code Review	19
Coding Style	19
Peer Review	19
Applicant Verification	19
Twitter Analysis	20

Stock Database Retrieval	20
Stock Search Filtering	21
Purchasing Stocks	21
Data Encryption	22
Add to Watchlist	22
Accept/Reject Applicants	23
Self-check: Best Practices For Security	24
Major Assets Protected:	24
Processing of Encrypting Major Assets in DB:	24
Input Data Validation:	25
Email Address	25
Password	25
Self-check: Adherence To Original Non-functional Specs	27
Security	27
Audit	28
Performance	28
Capacity	29
Reliability	29
Recovery	29
Data Integrity	30
Compatibility	31
Conformation with Coding Standards	32
Look and Feel Standards	34
Scalability	35
Web Site Policies	36

Product Summary

Tradiction

UNREGISTERED USERS: -

1. Shall be able to create a new account as a trader.
 - 1.1. Shall be able to enter an email address of choice during registration.
 - 1.2. Shall be able to enter a password during registration.
 - 1.3. Shall have minimum password requirement of letters, numbers and special characters.
 - 1.4. Shall be able to enter the address during registration.
 - 1.5. Shall be able to enter contact number during registration.
 - 1.6. Shall be able to enter Social Security Number during registration.
 - 1.7. Shall be able to link a bank account during registration.
 - 1.8. Shall be able to upload bank statements during registration.
 - 1.9. Shall receive an Email verification upon registration.
2. Shall be able to create a new account as an expert.
 - 2.1. Shall be able to enter an email address of choice during registration.
 - 2.2. Shall be able to enter a password during registration.
 - 2.3. Shall have minimum password requirement of letters, numbers and special characters
 - 2.4. Shall be able to enter the address during registration.
 - 2.5. Shall be able to enter contact number during registration.
 - 2.6. Shall be able to enter Social Security Number during registration.
 - 2.7. Shall receive an Email verification upon registration.
 - 2.8. Shall be required to upload an expert certificate.
3. Shall be able to view Stock lists.
 - 3.1. Shall be able to sort the list alphabetically.
 - 3.2. Shall be able to search for a list of particular stocks.

- 3.2.1. Shall be able to enter text into a search field
 - 3.2.2. Shall be able to modify text in a search field.
- 4. Shall be able to view a particular stock in detail.
 - 4.1. Shall be able to click on the individual stock.
 - 4.2. Shall be redirected to a detailed stock page.
- 5. Shall be able to view news feeds.
 - 5.1. Shall be able to click on news articles.
 - 5.2. Shall be able to open another tab with an article.
- 6. Accounts shall be put on hold for verification from admin.

REGISTERED TRADERS: -

- 1. Shall be able to login.
 - 1.1. Shall be able to enter an email address.
 - 1.2. Shall be able to enter a password.
 - 1.3. Shall have minimum password requirement of letters, numbers and special characters.
 - 1.4. Shall be authenticated during login.
 - 1.5. Shall have the account blocked for a few hours, incase of multiple incorrect login attempts.
 - 1.6. Shall be prompted with a message if credentials are incorrect.
 - 1.7. Shall be able to recover password through mail by clicking on "Forgot Password".
- 2. Shall be able to edit personal information.
 - 2.1. Shall be able to change their password.
 - 2.2. Shall be able to update their phone number.
 - 2.3. Shall be able to update their address.
- 3. Shall be able to logout.
- 4. Shall be able to view news feeds.
 - 4.1. Shall be able to click on news articles.
 - 4.2. Shall be able to open another tab with an article.

5. Shall be able to view the Stock list.
 - 5.1. Shall be able to sort the list alphabetically.
 - 5.2. Shall be able to search for a list of particular stocks.
 - 5.2.1. Shall be able to enter text into a search field
 - 5.2.2. Shall be able to modify text in a search field.
 - 5.3. Shall be able to add stock to the watchlist.
6. Shall be able to view a particular stock in detail.
 - 6.1. Shall be able to click on the individual stock.
 - 6.2. Shall be redirected to a detailed stock page.
7. Shall be able to view the portfolio.
 - 7.1. Shall be able to show a summary of the portfolio for the day.
 - 7.1.1. Shall be able to show all stocks owned.
 - 7.1.2. Shall be able to show gains and losses from purchase price.
 - 7.2. Shall be able to sell stock.
 - 7.2.1. Shall be able to enter a quantity of stock to sell.
8. Shall be able to buy stocks by clicking on the buy button on the stock details page.
 - 8.1. Shall be able to enter quantities of stocks to buy.
9. Shall be able to view the transaction history.
 - 9.1. Shall be able to view the stock name of the stock bought.
 - 9.2. Shall be able to view the purchasing price of stock bought.
 - 9.3. Shall be able to view the quantity of stock bought.
 - 9.4. Shall be able to view the date of stock bought.
 - 9.5. Shall be able to view the time of stock bought.
10. Shall be able to add interested stocks to a watchlist.

REGISTERED EXPERTS: -

1. Shall be able to login.
 - 1.1. Shall be able to enter an email address.

- 1.2. Shall be able to enter a password.
 - 1.3. Shall be authenticated during login.
 - 1.4. Shall be prompted with a message if credentials are incorrect.
 - 1.5. Shall be able to recover password through mail by clicking on “Forgot Password”.
2. Shall be able to edit personal information.
 - 2.1. Shall be able to change their password.
 - 2.2. Shall be able to update their phone number.
 - 2.3. Shall be able to update their address.
3. Shall be able to logout.
4. Shall be able to view news feeds.
 - 4.1. Shall be able to click on news articles.
 - 4.2. Shall be able to open another tab with an article.
5. Shall be able to view the Stock list.
 - 5.1. Shall be able to sort the list alphabetically.
 - 5.2. Shall be able to search for a list of particular stocks.
 - 5.2.1. Shall be able to enter text into a search field
 - 5.2.2. Shall be able to modify text in a search field.
 - 5.2.3. Shall be able to click on the search button to view results.
 - 5.3. Shall be able to add stock to the watchlist.
6. Shall be able to view a particular stock in detail.
 - 6.1. Shall be able to click on the individual stock.
 - 6.2. Shall be redirected to a detailed stock page.
7. Shall be able to add interested stocks to a watchlist.

ADMIN: -

1. Shall be able to login.
 - 1.1. Shall be able to enter an email address.
 - 1.2. Shall be able to enter a password.
 - 1.3. Shall be authenticated during login.

2. Shall be able to logout.
3. Shall be able to view details entered by traders.
4. Shall be able to view details entered by experts.
5. Shall be able to put traders' accounts on hold for verification.
6. Shall be able to put experts' accounts on hold for verification.
7. Shall be able to accept trader accounts.
8. Shall be able to reject trader accounts.
9. Shall be able to accept expert accounts.
10. Shall be able to reject expert accounts.

Tradiction is a stock trading app that allows users the ability to browse recent stock-related news articles while also giving them the ability to buy, sell, and view stocks. Unlike other stock trading apps, Tradiction implements a proprietary sentimental analysis algorithm that analyzes twitter comments to keep users up to date on the latest trending market chatter. Tradiction also provides a feature allowing traders to communicate directly with stock experts, giving the trader the best market advice and information related to the stock possible. Tradiction can be found at <http://ec2-18-222-193-100.us-east-2.compute.amazonaws.com:8000>.

Usability Test Plan

Test Objectives

For our usability test plan, we decided to test five features: the sentimental twitter analysis, adding stocks into the watchlist, filtering stock information, searching for stock information, and purchasing stocks. First, we want to gauge the user's feedback on having a tool that provides the user with relevance of stocks being discussed. Second, we want to see how the user responds to adding stocks to the watchlist. Third, we want to see how intuitive it is for users to filter stock information. Fourth, we want to see how the user reacts to the iterative search functionality. Lastly, we want to see how the user feels about purchasing stocks.

Test Description

System Setup

The website is deployed on AWS on an EC2 instance. The database is on mysql and is also running on the same EC2 instance. The site is compatible with the last version of the Safari browser 11.1.2, the last version of Firefox version 70, and the last version of Chrome version 80.

Starting Point

The starting point for the website is the home page, where the user navigates to all other sections of the website using the navbar. From the home page, the user will also have the ability to create an account or log in.

Intended Users

The intended users are those interested in investing or those who would like to find out more information about the latest stock or investment related news.

URL of the system

The url to access the system is:

<http://ec2-18-222-193-100.us-east-2.compute.amazonaws.com:8000>

What is Measured

The usability test plan aims to measure user's satisfaction (ease of use) for each of the five features.

Usability Task Description

1. *Sentimental Twitter Analysis*

First, the user will have to navigate to the stock page. Next, the user will have to navigate to the detailed stock page from one of the stocks on the table. From there, the user will be able to view the

Task	Description
Task	View the sentimental twitter analysis in the detailed stock section.
Machine State	Homepage of Tradiction
Successful Completion Criteria	Successfully navigates to the detailed stock section and views the results of the sentimental twitter analysis.
Benchmark	≈ 1 minute

2. *Add Stocks into Watchlist (Registered Traders)*

Taking the assumption that the user already has an account, the user will first need to login. Then the user will navigate to the stocks page, where they can add stocks into the watchlist. They can then navigate to the watchlist to ensure that the stock was successfully added to the watchlist.

Task	Description
Task	Add a stock into the watchlist from the stock page and navigate to the watchlist page to ensure the stock was successfully added.

Machine State	Homepage of Tradiction
Successful Completion Criteria	User navigates to the watchlist page and confirms that the stock they added is present.
Benchmark	<= 5 minutes

3. *Filter Stock Information Alphabetically (A-Z) by Name*

The user will navigate to the stock page and filter the table by name by increasing order.

Task	Description
Task	Filter the table of stocks alphabetically (A-Z)
Machine State	Homepage of Tradiction
Successful Completion Criteria	Table of stocks successfully in order from (A-Z)
Benchmark	<= 2 minutes

4. *Search for Stock Information*

The user will navigate to the stock page and search for a stock using the search bar next to the table.

Task	Description
Task	Search for a specific stock using the search bar.
Machine State	Homepage of Tradiction

Successful Completion Criteria	Searching for the stock name on the search bar yields the proper stock information.
Benchmark	<= 2 minutes

5. *Purchase Stocks (Registered Traders)*

Taking the assumption that the user already has an account, the user will first need to login. Next, the user will have to go to the stock page and navigate to a stock in detail. From there, the user will purchase a stock using the buy button. The user will be directed to a buy page, where they can select a quantity of the stock that they would like to purchase. Once the purchase has been complete, they should go to the transaction history page to verify that the purchase was successful.

Task	Description
Task	Purchase any number of given stocks and verify that the stock has been successfully purchased by viewing the trading history section.
Machine State	Homepage of Tradiction
Successful Completion Criteria	Trading history section successfully shows stock purchased.
Benchmark	<= 5 minutes

Usability Test Table

Test/Use Case	% Completed	Errors	Comments
View sentimental twitter analysis	100%	None	Easy to navigate to sentimental twitter analysis.

Add stocks into watchlist	100%	None	Easy to add stocks to the watchlist.
Filter stock information alphabetically, (A-Z)	100%	None	Users had no issues with this functionality.
Search for stock information	100%	None	Users had no issues with this functionality.
Purchase stocks	100%	None	Some users struggled with purchasing stocks due to the amount of navigating that needed to take place.

Questionnaire

Please mark only one x for each question.

Question	Strongly Disagree	Disagree	Neutral	Agree	Strongly Agree
1. I found the system unnecessarily complex.			x		
2. I found the system very cumbersome to use.		x			
3. I thought the system was easy to use.				x	

QA Test Plan

Test objectives

First, we will be testing to ensure that confidential user information (passwords and user payment information) is properly being encrypted before being stored in the database. Next, we will be testing to see if an administrator can audit and approve new registrations. Lastly, we will test to ensure that the application can retrieve information from the database in a timely manner.

HW and SW setup

AWS EC2 free tier eligible running Ubuntu and mysql. Django is the web framework to ensure that the user can visit the website. Mysqlworkbench will be used to access the database locally. Users will test on Chrome and Firefox. Link to test is:

<http://ec2-18-222-193-100.us-east-2.compute.amazonaws.com:8000/index>.

1. Passwords shall be encrypted before storing in the database.

QA test plan

Test #	Title	Description	Test Input	Expected Corrected Output	Test Results
1	Password field hidden	When user tries entering text in password field, it should display as ****.	Test123.	*****	Pass
2	Password complexity not met	When the user does not fulfill the password complexity requirement, prevent the user from continuing and prompt the user that the password provided was incorrect.	Test	Message in red prompting the user that the password did not meet complexity requirements. Reject the password provided.	Pass
3	Login successful	When the user	User	Redirect to the	Pass

		enters login information for an account that exists, allow the user to login.	enters correct login information.	home page. Will display the user name on top right.	
--	--	-------------------------------------------------------------------------------	-----------------------------------	-----------------------------------------------------	--

2. Users payment information shall be encrypted before storing in the database.

QA test plan

Test #	Title	Description	Test Input	Expected Corrected Output	Test Results
1	Create a trader account.	User creates a trader account.	Fill out all forms correctly.	Prompt notifying the user that the account creation is under review by an administrator.	Pass
2	Failure to provide SSN when creating a trader account.	User leaves the SSN field blank when creating a trader account.	Fills out form correctly except leaving SSN field blank.	Prompt notifying the user that the account cannot be created because one or more required fields were left blank.	Pass
3	Failure to provide Routing Number when creating a trader account.	User leaves the routing number field blank when creating a trader account.	Fills out form correctly except leaving routing number field blank.	Prompt notifying the user that the account cannot be created because one or more required fields were left blank.	Pass

3. New registrations shall be audited by the administrator.

QA test plan

Test #	Title	Description	Test Input	Expected Corrected Output	Test Results
1	Email confirmation for registered user	When a registered account has been created, an automated email will let the user know that their account is under review.	Enter all fields in registration correctly and click Register.	Popup letting the user know that their account is pending approval. Redirect to the home page.	Fail
2	Unapproved account login	User tries logging in before the account has been approved	Enter correct information while account is under approval.	An error message for the user letting them know that their account is still under review.	Pass
3	Send help message to administrator	An unregistered user can navigate to the help page and leave an email for the administrator.	Enter proper email and message in the contact us section.	A message letting the user know that the message has successfully been sent.	Fail

4. New registrations shall be approved by the administrator.

QA test plan

Test #	Title	Description	Test Input	Expected Corrected Output	Test Results
1	Email confirmation for account approval.	When a registered account has been approved, an automated response	None	The user will receive an email letting them know that their account	Pass

		will be sent by the administrator.		has been approved.	
2	Registered user can login	Once the account has been approved, the user can login successfully.	User enters correct login information.	The user will be redirected from the login page to the home page. The user's name will be displayed on the top right.	Pass
3	Unapproved account login	When the user tries logging in before the account has been approved, there will be an error message for the user letting them know that their account is still under review.	Enter correct information while account is under approval.	An error message for the user letting them know that their account is still under review.	Pass

5. Application shall be able to retrieve information from the database and react in a timely manner.

QA test plan

Test #	Title	Description	Test Input	Expected Corrected Output	Test Results
1	Visit home page	Load the home page	Enter the URL: http://ec2-18-222-193-100.us-east-2.compute.amazonaws.com:8000/prototypeapp/index into the browser search	Page loads successfully in ≥ 3 seconds.	Pass

			bar.		
2	Login successful	From the login page, enter correct credentials.	User enters correct login information.	Page loads successfully in ≥ 3 seconds. Redirect to the home page.	Pass
3	Visit news page	From the home page, click on the news tab.	Click on the news tab.	Page loads successfully in ≥ 3 seconds.	Pass

Code Review


Coding Style


- All code will be properly indented using tabs.
- All code will have understandable variable and function names.
- All algorithms that are complex will have comments embedded for easier understandability.
- All front-end code will adhere to the styling guidelines provided in the styling chart.

Peer Review

Below are the email chains for specific sections of code:

Applicant Verification

CSC 648 - Code Review (Applicant Verification) 


 **Brian Lim** <blim292@gmail.com>
to miranisjanjay417, dibs@harati, benjamin.fletes, may.naing97, xianzke ~


8:03 PM (1 hour ago) ☆ ↶ ⋮

Hi Everyone,


This is the code that Sanjay wrote for the administrator to verify applicants. Please take a look at it and give him your feedback.

Thanks,
Brian


 verification.txt

 **dibs@harati**
The function definitions make sense and the error handling in this sections seems to be done well.


8:48 PM (1 hour ago) ☆

 **Benjamin Fletes**
> <verification.txt> It is good just the naming like verifyExpert should be verifyExpert just to make it easy for people to read and for me when two words in wh

8:57 PM (1 hour ago) ☆

 **Kevin Syn**
The code is easy to read and understandable I think it is no senior issue here.

9:27 PM (34 minutes ago) ☆

 **Brian Lim** <blim292@gmail.com>
to Kevin, Benjamin, miranisjanjay417, dibs@harati, May ~

9:53 PM (7 minutes ago) ☆ ↶ ⋮

These functions are simple as is and do not require any changes or comments.

xxx

Twitter Analysis

CSC 648 - Code Review (Twitter Analysis)

Brian Lim

<blim292@gmail.com>

to dibabharati, miranisanjay417, xianzike, benjamin fletes, maynaing97

Hi Everyone,

This is the code that Sanjay wrote for the sentimental twitter analysis on the detailed stock page. Please take a look at it and give him your feedback.

Thanks,
Brian

twitteranalysis.txt

8:01 PM (1 hour ago)

☆ ↻ ⋮

dibs bharati

The code looks good. The "if statement" is easy to understand and seems to follow the principles of procedural programming utilizing the computer resources eff.

8:45 PM (1 hour ago) ☆

Benjamin Fletes

<benjaminfletes@gmail.com>

to miranisanjay417, dibabharati, benjamin fletes, maynaing97, xianzike

> <twitteranalysis.txt> From the twitter it is good but the naming got me confused and realized it was negative or positive best it names it positive and negati

8:53 PM (1 hour ago) ☆

Kevin Syn

<ksyn@gmail.com>

to miranisanjay417, dibabharati, benjamin fletes, maynaing97, xianzike

I am not easy to read on this part since it has not commented on the analysis and variables. But it still looks good to me for now.

9:49 PM (10 minutes ago) ☆

Brian Lim

<blim292@gmail.com>

to miranisanjay417, dibabharati, benjamin fletes, maynaing97, xianzike

I would def. try to include comments for this. Reading through it, comments would def. help with understanding what is happening.

9:55 PM (4 minutes ago) ☆

May Naing

<maynaing97@gmail.com>

to me, miranisanjay417, dibabharati, benjamin fletes, maynaing97, xianzike

I think since you're using an api here, comments would really help understand what's going on. Other than that, everything looks good.

9:59 PM (0 minutes ago) ☆ ↻ ⋮

Stock Database Retrieval

CSC 648 - Code Review (Stock Database Retrieval)

Brian Lim

<blim292@gmail.com>

to miranisanjay417, dibabharati, benjamin fletes, maynaing97, xianzike

Hi Everyone,

This is the code that Sanjay wrote to retrieve information from the database in a timely manner. Please take a look at it and give him your feedback.

Thanks,
Brian

stockdatabaseretr...

8:00 PM (1 hour ago)

☆ ↻ ⋮

dibs bharati

Please add few lines of comment whenever necessary. Beside that everything looks good.

8:40 PM (1 hour ago) ☆

Benjamin Fletes

<benjaminfletes@gmail.com>

to miranisanjay417, dibabharati, benjamin fletes, maynaing97, xianzike

> <stockdatabaseretrival.txt> There could a small explanation for symbols on what they are taking and I is not a good naming convention maybe row would be bet

8:48 PM (1 hour ago) ☆

Kevin Syn

<ksyn@gmail.com>

to miranisanjay417, dibabharati, benjamin fletes, maynaing97, xianzike

No need to change at me for this part.

9:40 PM (16 minutes ago) ☆

Brian Lim

<blim292@gmail.com>

to miranisanjay417, dibabharati, benjamin fletes, maynaing97, xianzike

The code was easy to understand, probably don't need to change anything, but feel free to add some comments if you want.

9:44 PM (12 minutes ago) ☆

May Naing

<maynaing97@gmail.com>

to me, miranisanjay417, dibabharati, benjamin fletes, maynaing97, xianzike

Everything looks good here. Nothing needs to be changed.

9:45 PM (12 minutes ago) ☆ ↻ ⋮

Great, thanks!

Thanks!

Looks good to me.

Stock Search Filtering

CSC 648 - Code Review (Stock Search Filtering)

7:58 PM (2 hours ago)

Brian Lim

<bim292@gmail.com>

to mirisanjay417, dibsbharati, benjamin.fletes, maynaing97, xianzke

Hi Everyone,

This is the code that Sanjay wrote to filter the stocks table using the search bar. Please take a look at it and give him your feedback.

Thanks,
Brian

search.txt

dibs bharati

The code looks beautiful. The alignment and spacing are perfect.

8:38 PM (1 hour ago)

Benjamin Fletes

> <search.txt> It looks good. Not sure if there is a mapping but there is no mapping then use indexing. But my question to this is what are they are searching?

8:44 PM (1 hour ago)

May Naing

I don't think there's anything that needs changing here. Looks good to me.

9:29 PM (30 minutes ago)

Kevin Syn

Looks good to me at this part.

9:31 PM (28 minutes ago)

Brian Lim

<bim292@gmail.com>

to Kevin, May, mirisanjay417, dibsbharati, Benjamin

The code is concise. I would probably look to add some comment above the script to know exactly what the script does.

9:35 PM (24 minutes ago)

Purchasing Stocks

CSC 648 - Code Review (Purchasing Stocks)

7:56 PM (2 hours ago)

Brian Lim

<bim292@gmail.com>

to dibsbharati, mirisanjay417, benjamin.fletes, maynaing97, xianzke

Hi Everyone,

This is the code that Sanjay wrote for the registered trader to buy stocks. Please take a look at it and give him your feedback.

Thanks,
Brian

purchase.txt

dibs bharati

The code looks nice and clean. Few more lines of comment would be helpful.

8:37 PM (1 hour ago)

Benjamin Fletes

> <purchase.txt> The code is good I do not see anything that is a problem. I do not see anything that brings to my concern. The only suggestion would be the var

8:39 PM (1 hour ago)

May Naing

This is a pretty large chunk of code. I think comments could be added here as well to explain what's going on.

9:26 PM (34 minutes ago)

Kevin Syn

It looks good for the purchase and understandable. Since it has lot of variables, more comments about the details should be better.

9:30 PM (29 minutes ago)

Brian Lim

<bim292@gmail.com>

to Kevin, May, dibsbharati, mirisanjay417, Benjamin

This function has a lot of content. However, variable names are easy to understand. I would probably look to add some comments within each section so that when you refer to your code in the future, it'll be easier to digest and understand.

9:40 PM (20 minutes ago)

21

Data Encryption

CSC 648 - Code Review (Data Encryption)

Brian Lim

<bilm292@gmail.com>

to ddbbharati, miranisanjay417, benjamin.fletes, maynaing97, xianzike

Hi Everyone,

This is the code that Sanjay wrote to encrypt data before storing it in the database. Please take a look at it and give him your feedback.

Thanks,

Brian

dataencryption.txt

7:54 PM (2 hours ago)

☆

Benjamin Fletes

> <dataencryption.txt> It is good and there is nothing to worry.

8:31 PM (1 hour ago)

☆

dibs bharati

The code is well written. The general names for the various request methods have followed. All the loop are efficiently executed.

8:53 PM (1 hour ago)

☆

May Naing

Naming conventions here are a little shorter, making it slightly harder to understand. Also, not sure if this part is important but I think you're missing the p

8:56 PM (1 hour ago)

☆

Kevin Syn

pretty good on this adding database code but I can't see if the code has some point need to worry about.

9:36 PM (22 minutes ago)

☆

Brian Lim

<bilm292@gmail.com>

to Kevin, May, ddbbharati, miranisanjay417, Benjamin

Looks good, I would look to put some comments to better show what this code does:

```
exrouting = "\t\t" + AESCipher(key).encrypt(routingno) + "\t\t"
eaccount = "\t\t" + AESCipher(key).encrypt(accono) + "\t\t"
essn = "\t\t" + AESCipher(key).encrypt(ssnno) + "\t\t"
```

...

9:52 PM (7 minutes ago)

☆

Add to Watchlist

CSC 648 - Code Review (Add to Watchlist)

Brian Lim

<bilm292@gmail.com>

to miranisanjay417, ddbbharati, benjamin.fletes, maynaing97, xianzike

Hi Everyone,

This is the code that Sanjay wrote for a trader to add stocks to the watchlist. Please take a look at it and give him your feedback.

Thanks,

Brian

addtowatchlist.txt

7:52 PM (2 hours ago)

☆

Benjamin Fletes

> <addtowatchlist.txt> I do see it as good and there is no problem.

8:29 PM (1 hour ago)

☆

dibs bharati

The function addtowatchlist is done well with careful executions of the "if else" statement with precise coding.

8:50 PM (1 hour ago)

☆

May Naing

Everything looks good here. Code is pretty straight forward. I don't think there's anything that needs fixing other than adding comments.

9:15 PM (44 minutes ago)

☆

Brian Lim

Nothing needs to be changed. The code is readable and easy to understand.

9:25 PM (34 minutes ago)

☆

Kevin Syn

to me, May, miranisanjay417, ddbbharati, Benjamin

This one is easy to read. Looks good to me.

...

9:32 PM (27 minutes ago)

☆

Accept/Reject Applicants

CSC 648 - Code Review (Accept/Reject Applications) 



Brian Lim <blim292@gmail.com>

To: mirensanjay417, dibsbharati, benjamin.fietes, xiancike, may.naing97

Hi Everyone,

This is the code that Sanjay wrote for the administrator to accept or reject applicants. Please take a look at it and give him your feedback.

Thanks,
Brian



acceptrejectappli...



Benjamin Fietes

> <acceptrejectapplications.txt> Based on looking at this code, I believe it is good and the only thing is reduce some of the lines in my case. Some of the line

8:27 PM (1 hour ago)



May Naing

I think everything looks good. Naming conventions are clear and concise, making it easier to understand what each function does. I think you could add short com

8:43 PM (1 hour ago)



dibs bharati

I believe this program, which accepts or rejects user application is good enough. The error handling makes sense. Maybe a suitable printout message when the use

8:59 PM (59 minutes ago)



Kevin Syn

This part looks pretty clear to me. If more comments here will be easier to read.

9:38 PM (19 minutes ago)



Brian Lim <blim292@gmail.com>

To: Kevin, May, mirensanjay417, dibsbharati, Benjamin

For the accepttrader function, I would probably add a comment for what this does:

```
e = rows[0][0]
```

Otherwise, code is understandable and does not require comments.

xxx

9:49 PM (9 minutes ago)



 Reply

 Reply all

 Forward

Self-check: Best Practices For Security

Major Assets Protected:

- Trader Registration Information
 - SSN
 - Bank Statement
 - Routing Number
 - Account Number
 - Password
- Expert Registration Information
 - SSN
 - Password
- Portfolio
 - Purchasing Stocks
 - Selling Stocks

Processing of Encrypting Major Assets in DB:

To encrypt the data, we used an API called AESCipher. By providing a 16-bit key, we are able to encrypt the data using the AESCipher object and an encrypt method. Once the strings have been encrypted, they are uploaded into the database where they can later be retrieved and decrypted using the same key.

Encrypted Trader Table:

id	firstname	lastname	address	city	state	phoneno	ssno	bankst	routingno	accountno	lid	status
5	Sanjay	Mirani	2505 33rd Avenue	San Francisco	California	4154259203	kdFRmVd+XS+LJVBVDASPu0eVP9wYmeqY...	media/SanjayMirani_bankstatement.jpg	EUWd0dFJex8ZNYy8XLWJs+LE0gkL3QZrR2...	2ENJAUN0i2YE7Am+TpsQyy499I5gt15Pb...	31	Accepted

Encrypted Login Table:

lid	username	password	role
30	admin	admin	admin
31	miranisanjay417@gmail.com	JP/lgWpWAlq0+7r6Kvm/wUKbpYyKAYJLV2S72...	trader
NULL	NULL	NULL	NULL

Input Data Validation:

The email and password validations were done using the javascript functionality. The minimum amount of requirements were set for both of them.

- *Email Address*

1. The input email can't be empty. If it is empty, suggestive text will appear in red color.
2. If the email has numbers, letters, '@' symbol and the name of the site issuing the email, it is valid. In this case, a text will appear showing the approval of input.

The code is presented below:

```
21 <script>
22
23     function validate()
24     {
25         var Email = document.myform.email.value;
26         var password1 = document.myform.cpwd1.value;
27         var password2 = document.myform.cpwd2.value;
28         var regularExpression = /^(?=.*\d)(?=.*[a-zA-Z])[a-zA-Z0-9]{7,}$/;
29         var status=false;
30         if (Email === "" )
31         {
32             document.getElementById("mail").innerHTML = "Enter the correct email address";
33             document.getElementById("mail").style.color = "Red";
34             status=false;
35         }
36         else if(Email.match(/^[A-Za-z0-9]+@(gmail.com|yahoo.com|yahoo.in|gtu.edu.in|mail.sfsu.edu)$/))
37         {
38             document.getElementById("mail").innerHTML = "Ok input";
39             document.getElementById("mail").style.color = "green";
40             status=true;
41         }
42     }
43
44
45     if (!regularExpression.test(password1)) {
46         document.getElementById("psw1").innerHTML = "The password doesnt fulfill the requirement";
47         document.getElementById("psw1").style.color = "Red";
48     }
49     status=false;
50 }
```

- *Password*

1. The input password must have a lowercase letter, an uppercase letter, a number and should be at least seven characters long. Suggestive text will appear in red color if the requirement is not met.
2. The user will be asked to input their password twice. If the passwords don't match, suggestive text will appear in red color.
3. When all the conditions are met, a text will appear in green color saying the requirements are fulfilled.

The code is presented below:

```

44
45     if (!regularExpression.test(password1)) {
46         document.getElementById("psw1").innerHTML = "The password doesnt fulfill the requirement";
47         document.getElementById("psw1").style.color = "Red";
48
49         status=false;
50     }
51     else if (password1 !== password2) {
52         document.getElementById("psw1").innerHTML = "The password doesn't match";
53         document.getElementById("psw1").style.color = "Red";
54         document.getElementById("psw2").innerHTML = "The password doesnt match";
55         document.getElementById("psw2").style.color = "red";
56         status=false;
57     }
58     else
59     {
60         document.getElementById("psw1").innerHTML = "Password requirements satisfied";
61         document.getElementById("psw1").style.color = "green";
62         document.getElementById("psw2").innerHTML = "Password requirements satisfied";
63         document.getElementById("psw2").style.color = "green";
64
65     }
66
67     return status;
68 }
69
70 </script>

```

Self-check: Adherence To Original Non-functional Specs

Security

Non-Functional Requirement	Status	Additional Information
Login shall be required to make purchases.	DONE	
User's shall verify their emails when registering an account.	DONE	
User's shall be able to set a display name different from their email.	ON TRACK	
User's emails shall not be displayed by default.	ON TRACK	
Passwords shall be encrypted before storing in the database.	DONE	
User's session timeout limit shall be decided by the administrator.	DONE	
User's session shall only be ended by code design.	DONE	
Content uploaded by users shall be audited by the administrator.	DONE	
User's payment information shall be encrypted.	DONE	
This site shall not accept any third-party cookies.	ON TRACK	

Audit

Non-Functional Requirement	Status	Additional Information
New registrations shall be audited by the administrator.	DONE	
New registrations shall be approved by the administrator.	DONE	
Users shall not be able to login to administrator accounts.	DONE	

Performance

Non-Functional Requirement	Status	Additional Information
The site loading time shall be less than 2 seconds for all screens.	ON TRACK	
Application shall be able to retrieve information from the database and react in a timely manner.	DONE	
The site shall handle requests asynchronously following a REST format.	ON TRACK	

Capacity

Non-Functional Requirement	Status	Additional Information
The total data storage for the site shall not exceed 80% of the server's capacity for this site.	DONE	
The website shall be capable of handling at least 50 users.	DONE	
The website shall be scalable, so that new features can be added easily.	DONE	

Reliability

Non-Functional Requirement	Status	Additional Information
Downtime for maintenance shall be less than 3 hours per month.	ON TRACK	
Downtime for maintenance shall not affect the site's main functionality.	ON TRACK	
In all cases, users shall be informed of downtime for maintenance, either via an announcement on the main page, or e-mail.	ON TRACK	

Recovery

Non-Functional Requirement	Status	Additional Information
In case of a total site failure, the whole site shall be shut down for revision.	DONE	
If the site is broken, the mean time to recovery shall not exceed one day.	ON TRACK	
User data is the most valuable aspect and priority will be placed on recovering such data in case of total failure.	ON TRACK	

Data Integrity

Non-Functional Requirement	Status	Additional Information
Database tables shall be backed up weekly.	ON TRACK	
Administrator shall be able to execute a recovery if needed.	DONE	
Image sizes shall be restricted to at most 1 megabyte.	ON TRACK	
Images shall be uploaded in jpg, jpeg, or png formats.	DONE	
Images will be saved on Amazon's s3 storage server.	ISSUE	After further discussion we decided not to use s3 as it would overcomplicate our

		deployment strategy and so we decided on storing images in the project repository itself.
URLs to image will be stored on the database.	DONE	

Compatibility

Non-Functional Requirement	Status	Additional Information
The site shall be compatible with the last version of the Safari browser version 11.1.2	DONE	
The site shall be compatible with the last version of the Firefox browser version 70	DONE	
The site shall be compatible with the last version of the Chrome browser version 80	DONE	
Third party applications shall not be able to modify any content that may affect the site compatibility.	DONE	
Content should be able to be ignored by most popular ad-block services.	DONE	
The site shall be able to account for any other compatibility issues created as a result of browser updates in the future.	ON TRACK	

The site should be compatible to escalate to new databases.	ON TRACK	
--------------------------------------------------------------------	-----------------	--

Conformation with Coding Standards

Non-Functional Requirement	Status	Additional Information
Architecture and design standards shall meet all the requirements listed under the High-level system architecture and technologies used section of this document.	DONE	
Design pattern is to be strictly enforced with all aspects of the site.	DONE	
Appropriate documentation must be created for all code that is individually written for future maintenance.	ON TRACK	
Production code shall not have any log or output to the console.	ON TRACK	
All errors must not halt the web application without appropriate error handling.	ISSUE	We will look to find as many bugs as possible to account for them, but there will always be the possibility that there will be a bug that takes down the

		page (Will not kill the web server).
Only working code that meets all code standards shall be submitted to the main branch of the project repository.	ON TRACK	
Code shall be thoroughly tested and debugged before being considered working code.	ON TRACK	
All internal errors and exceptions encountered when writing or modifying code shall be stored in a log.	ISSUE	Did not implement this as in the development process. Internal errors and exceptions encountered were discussed as a team using slack and zoom.
Any error that can affect the site's functionality shall be reported to the user.	ON TRACK	
Errors shall be handled in a way that does not affect site functionality.	ON TRACK	
The whole production cycle of the site shall be finished at least one week before the delivery date.	ON TRACK	
The site shall be tested and debugged as a whole product	ON TRACK	

at least one week before the delivery date.		
The site shall not be launched without all priority one features finished and working.	ON TRACK	
All major changes to the application shall be discussed by the team and communicated to the class CTO.	ON TRACK	

Look and Feel Standards

Non-Functional Requirement	Status	Additional Information
The application and its layouts shall look professional.	ON TRACK	
The site shall be simple, so that it is usable to a wide range of users, and all previously mentioned parties.	DONE	
Targeted users will be the main priority for ensuring usability and readability.	DONE	
Elements on screen shall meet the compatibility standards of all supported browsers.	DONE	

Elements on screen shall meet the compatibility standards of all supported browsers on mobile devices.	ISSUE	Not enough time until due date to get mobile compatible code.
Elements on screen shall be aesthetically pleasing.	ON TRACK	
The site shall be able to work correctly without mouse interaction.	ISSUE	A mouse will be required to navigate through the website.
The site shall be able to work correctly without keyboard interaction.	DONE	
Elements in screen shall be resized automatically without user interaction when being loaded in all the different platforms supported by the site.	ISSUE	Not enough time to get bootstrap working effectively for all web pages.
Application's user interface shall make it easy for users to find what they are looking for.	ON TRACK	

Scalability

Non-Functional Requirement	Status	Additional Information
The CPU instance and storage capacity shall be updated to be able to handle a large number of users if needed.	DONE	

Web Site Policies

Non-Functional Requirement	Status	Additional Information
A link to the policies of this site shall be always visible in all its pages to be accessible by all the parties.	ON TRACK	
The site will not store any payment information.	DONE	
User's shall agree to the application's privacy policy before using the product.	DONE	