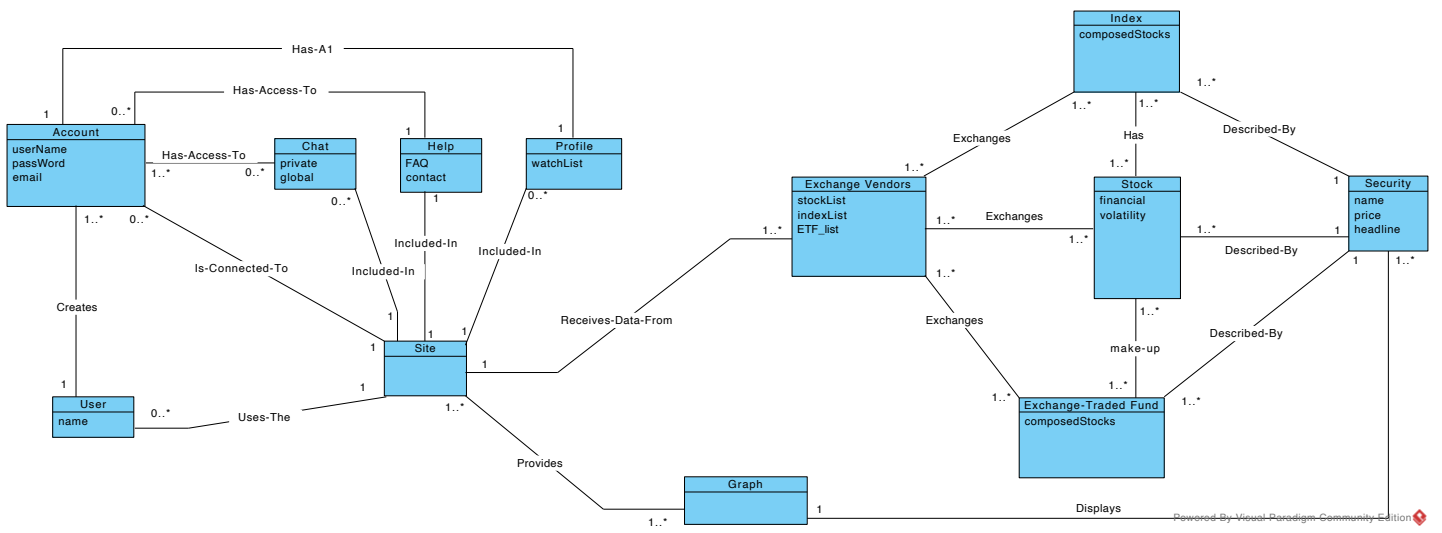# YoloTrader

The volatile stock analysis tool

Owen Murphy || Project Manager & Front End Developer
Prince Kalu || Software Designer & Backend Developer
Ty Wicks || Software Tester & Business Analyst

| ProjectVision |

YoloTrader is a dynamic website that houses an algorithm which extrapolates data from several market related variables to analyze stocks in order to predict short interest outcomes. Users will be able to create their accounts allowing them to add these desired stocks to their watchlist in order to follow them.
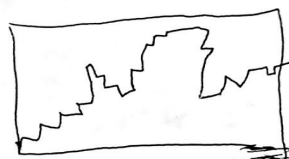
| Domain Model |
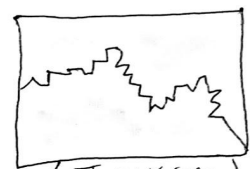
| **Wireframes** |

HOME | ABOUT | PROFILE | CONTACT

[TICKER]
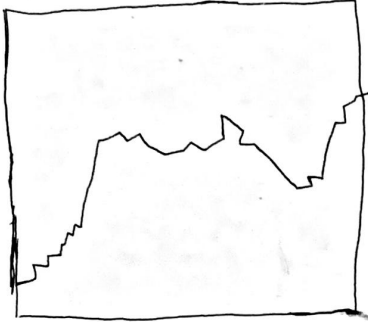
(TYPE CHAT HERE)

CONFIDENCE METER

KANGAROO
BULLISH
BEARISH
ODDS WINNING STRATEGY

| HOME | ABOUT | Profile | Contact | 🔍 |

Weclome to Software 1

Team

Owen

Kadu

Ty

| Use Cases |

Use Cases - Owen
- Contact for help
- Chat Thread
- Friend Lookup
- Delete Account
- Sign In

Use Cases - Prince
- Analyze Stock
- Lookup Stock
- Add to WatchList

Use Cases - Ty
- Create Account
- Delete Chat Thread
- Private Message

Use Cases - Owen

ID Contact for help
Scope Webpage contacting
Level user goal
Stakeholders and Interests
User - person on website and needs help understanding what to do
Staff - person responsible for taking requests from other people
Precondition: User is on the website
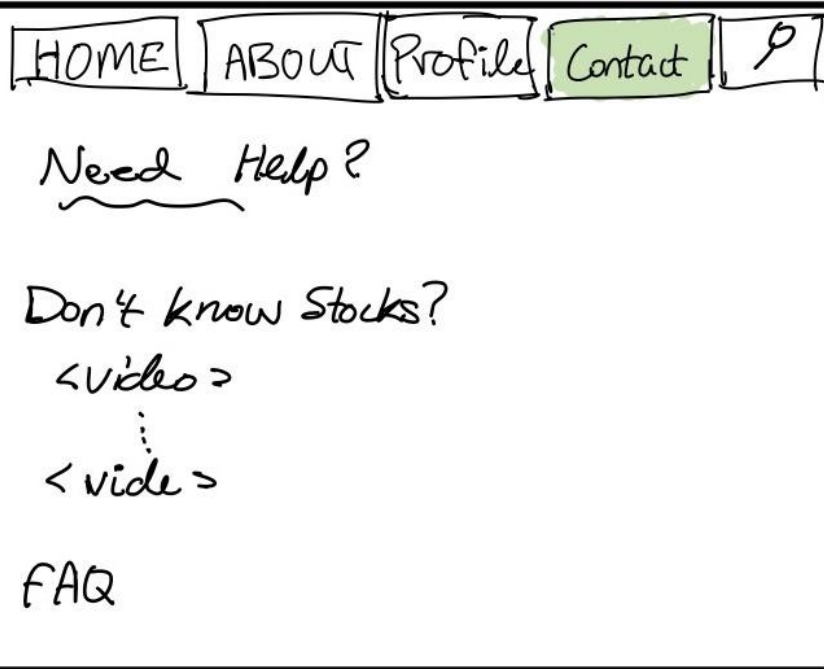**Postcondition:** User is contacted with a staff member

Main success scenario:
1. User navigates to the contact page
2. User finds the contact email address
3. The user clicks on the contact email address to initiate email
4. Their email application will start a new draft to the specified email address
5. The user types their message and sends
6. The user will be notified of a "Response Coming Soon!" subject email
7. The staff member receives the email and resopnds

Repeat steps 5 & 7 until the user is satisfied

8. The user and staff say their farewells

Extensions:
a* Anytime the website is not responding the user will check their internet connection and refresh the page
1a. If the user cannot find the "Contact" page
   a. The user will find a computer skilled person to help
2a. If the user cannot find the contact email address on the wepage
   a. The user will ask for a friend to assist
4a. If a new draft is not initiated on their OS
   a. The user will make sure they have a mailing application install on their OS
4b. If the email address does not show up in the "Send" box
   a. The user will copy the email address to the "Send" box
6a. The email auto responds letting user know email has been received
7a. If the user does not receive a response message
   a. The user can try contact a staff member directly, no guarantee of responding quickly

Use Cases - Owen

ID Chat Thread
Scope System Chat Thread
Level user goal
Stakeholders and Interests
User - person on website wants to talk about the top stocks
Staff - person responsible for ensuring civil chat threads pertaining to stocks
Precondition: User desires to discuss the top stocks
Postcondition: Chat Thread is initiated and the users talk about the stocks

Main success scenario:
1. User is on the homepage of the website and desires to talk about one of the top three stocks
2. The user clicks on the stock to view it, a url to another page
3. The webpage directs the user to the url of that stock
4. The system displays a chat box under the the stock
5. The user will click on the box and input their text
6. The user will click chat to send text to system
7. The system processes the text
8. The system displays the text below the chat box with all others

Repeat steps 5-8 up to five times to chat

Extensions:
a* Anytime the website is not responding the user will check their internet connection and refresh the page
2a. If the user cannot find the stock to click
   a. The user will find a computer skilled person to help
3a. If the url webpage is not found in the browswer
   a. The user will contact support
4a. If the user cannot find the chat box displayed under the stock
   a. Ther user will ask for a friend to help them locate it
6a. If the system does not respond
   a. the user will refresh the page
8a. If the text is not displayed
   a. The user will contact the contact email address and explain their situation

Use Case - Owen

ID Find Friends
Scope Locate friends' stock performance
Level user goal
Stakeholders and Interests
User - person on website wants to view other person stock performance
Staff - person responsible for maintaining the dynamic website
Precondition: User is on the website
Postcondition: The user find the specified person's stock performance

Main success scenario:
1. User gets on the website and want to see their friends page
2. User clicks on Find Friends
3. System prompts user to enter ID #
4. User enters the friends ID # and clicks "find"
5. System processes the number and
6. System requests server to ID associated with stock page
7. Systems receives the info and displays to the user's page

Extensions:
a* Anytime the website is not responding the user will check their internet connection and refresh the page
2a. If the user cannot find the find friends button to click
    a. The user will find a computer skilled person to help
3a. If the system does not prompt the user for the input
    a. The system will contact the help email address
4a. If the "find" button does not respond
    a. the user will try to refresh the page
5a. If the system does not indicate processing
    a. The user will refresh the page and try the process again
7a. If the system does not display the page
    a. The user will contact the help email address

Use Case - Owen

ID Delete Account
Scope User account is deleted
Level user goal
Stakeholders and Interests
User - a person registered under an account wants to delete their account
Staff - person responsible for maintaining the dynamic website
Precondition: User is logged onto the website
Postcondition: The user's account is deleted

Main success scenario:
1. User logs into account
2. User locates to the delete account button
3. User requests to delete account
4. System prompts the user to enter password
5. User inputs password
6. System validates password
7. Account is deleted

Extensions:
a* Anytime the website is not responding the user will check their internet connection and refresh the page
2a. If the user cannot find the delete account button to click
    a. The user will find a computer skilled person to help
4a. If a dialogue box does not appear for the user to enter password
    a. The user will check their browser compatibility
6a. If the password is invalid
    a. The system will ask for the password again

Use Case - Owen

ID Sign In
Scope User signs into account
Level user goal
Stakeholders and Interests
User - a person wants to login their account
Staff - person responsible for maintaining the dynamic website
Precondition: User is on the website
Postcondition: The user is logged into their account

Main success scenario:
1. User locates to the sign in page
2. System prompts user for username and password
3. User enters username and password
4. System validates credentials
5. User is logged into their account

Extensions:
a* Anytime the website is not responding the user will check their internet connection and refresh the page
1a. If the user cannot find the delete account button to click
    a. The user will find a computer skilled person to help
2a. If a dialogue box does not appear for the user to enter credentials
    a. The user will check their browser compatibility
4a. If credentials are invalid
    a. The system will ask the user to enter again

ID : Analyze Stock
Scope : Main Webpage function
Level : User goal
Stakeholders and Interests:
      User(Investor)- user wants to employ a winning strategy that will give them the greatest chances of profits, for the day.
Precondition: User is an investor, and wants another way to analyze volatile stocks.
Postcondition: The user is given the best strategy to apply, and the site waits for the outcome in order to improve upon its predictions.

Main success scenario: The Site Prediction is correct and the User went through with the best strategy and makes profit for the day.

Extensions: The user will be cautioned that this website is simply a tool, and is not sound investment advice, simply an odds calculator. The user will also be cautioned that the site will be wrong sometimes, and they should look to the confidence meter as a judge of the statistics presented.

ID : Add to watch list.
Scope : Main Webpage function
Level : User goal
Stakeholders and Interests:
      User(Investor)- user wants to add a stock to their watchlist.
Precondition: User is an investor, and wants to follow a specific stock.
Postcondition: The desired ticker(if found) is added to their watchlist.

Main success scenario: The desired ticker exists and is found, and was successfully added to the watchlist.

Extensions: If the desired ticker is not found then the closest tickers are shown, if any. The user will also be offered the option of emailing help, as well as placing a bug complaint if possible.

Use Cases - Prince

ID : Stock Look up
Scope : Main Webpage function
Level : User goal
Stakeholders and Interests:
      User(Investor)- user wants to look up a stock for a potential investment.
Precondition: User is an investor, and wants to find a stock.
Postcondition: The closest found ticker, along with related tickers are pulled up..

Main success scenario: The desired ticker exists and the equity graph as well as all related analytical data is presented to the user.

Extensions: If the desired ticker is not found then the closest tickers are shown, if any. The user will also be offered the option of emailing for help.

Use Cases - Ty

ID Create Account
Scope User creates an account
Level Connection Goal
Stakeholders and Interests
User - a person wants to create account to personalize a profile
Precondition:
User is a first-time visitor to site
Postcondition:
User's information is saved and will appear upon next visit to site

Main success scenario:
1. User enters site for first time
2. User clicks on create account
3. User enters username, email address and password
4. System logs username
5. System sends user email confirmation

Extensions:

a* If the user does not have a valid email address
2a. The user is allowed to access the websites pages and functions
   b. Upon re-entry, the user will be asked to accept cookies again and followed stocks will not be saved
   c. The user can create an email address at various sites
3a. If the user enters a taken username
   b. The system will ask the user to enter a new username

ID Delete Chat Thread
Scope Backend Maintenance
Level Maintenance
Stakeholders and Interests
User - Chat log will only contain relevant conversations for the day, preventing confusion on anecdotal advice.
Precondition:
Users have engaged with the chat.
The cutoff date and time for chat deletion is met.
Postcondition:
The current log of conversations have been erased.
The cutoff date and time are updated to the next day.

Main success scenario:
1. Users are notified ahead of end of day deletion
2. End of day cut off is met
3. Chat is disabled for 1 minute to ensure wipe is carried through
4. Chat is wiped
5. Chat is re-enabled for use

Extensions:
a* If the system fails to erase the chat at the selected time
2a. The system will make extra checks after the scheduled wipe to ensure a deletion was made
   a. If the check has not yet been made, a user can reach out via the help page
3a. If the chat does not re-enable itself
   c. The user will refresh their page
   d. If the chat is still disabled, the user will reach out to via the help page

Use Cases - Ty

ID Private message
Scope User Connection
Level User Goal
Stakeholders and Interests
User - User wants to interact with another user
Precondition: User is registered on website
Postcondition: The user interacts with another user privately

Main success scenario:
1. User looks up another user
2. Request to chat is sent to other user
3. User is notified when request is accepted
4. User enters their chat and presses enter to send
5. User is notified when the other user responds to their message

Extensions:
a* Anytime the chat page is not responding, the user will check their connection and refresh the page
2a. The user can't find the chat page
    a. The user will ask for help from a tech-savvy person they know
    b. The user will access the FAQ and help page for further guidance
3a. If the system does not find the entered username
    a. The system will ask for another username, and warn user about case sensitivity
4a. If the user can't submit a chat
    a. The user will refresh the page
    b. If a chat still can't be submitted, the user will seek reach via the help page
5a. If the second user declines message
    a. The user will be notified the user does not want to chat

| System Sequence Diagrams |

User

: System

1: addToWatchList(ticker)

1.1: ticker is on watchList
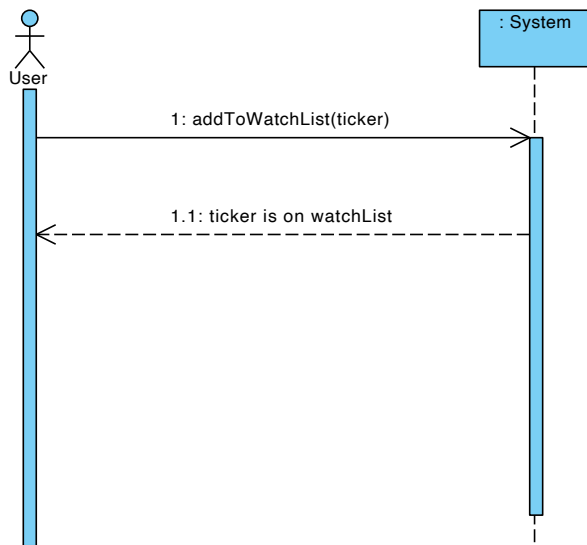
Operation: addToWatchList(ticker)
Cross Reference: Use Cases: Add Stock
Pre-conditions: User is on the stock
Post-conditions: User has successfully add the stock to their page

- stock is shown on profile (association formed)

User

: System

1: analyze(ticker)

1.1: bestStrategy, advisoryWarning, confidenceMeter
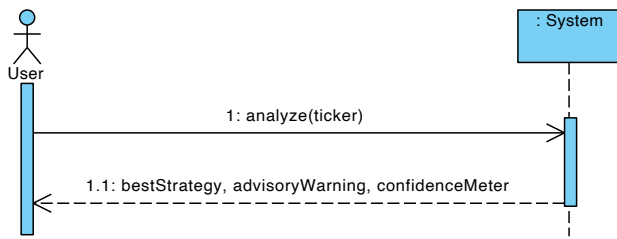
Operation: analyze(ticker)
Cross Reference: Use Cases: Analyze Stock
Pre-conditions: User is on stock page
Post-conditions: server analyzed stock and displayed results

● Best strategy is displayed
● Advisory warning displayed
● confidence meter displayed

Loop (Until deletion time is met)

1: deleteNotification(message)
1.1: incrementDeleteMessage

2: deletionTime
2.1: disableChat

3: deleteChat
3.1: re-enableChat
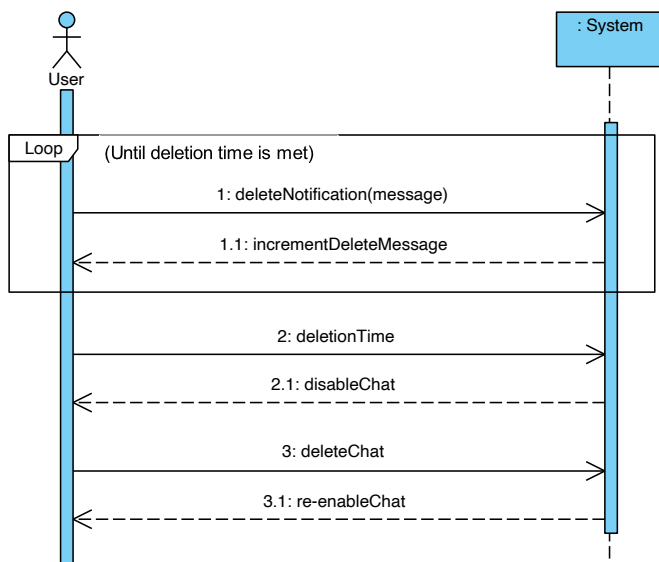
Operation: deleteNotification
Cross Reference: Use Cases: Registration
Pre-conditions: Current time nears deletion time
Post-conditions: Users are notified that chat deletion is nearing

● Users will be told how long until chat is deleted

Operation: deletionTime
Cross Reference: Use Cases: Registration
Pre-conditions: Users have been warned about chat deletion
Post-conditions: Users are locked out of chat

● Chat is locked from anyone accessing

Operation: deleteChat
Cross Reference: Use Cases: Registration
Pre-conditions: Chat has been locked
Post-conditions: Chat is erased and unlocked

● The day's chat is deleted
● Users are allowed back into chat

User

: System

1: clickStockName()

1.1: displayPage

loop

[Up to 3 times]

2: enterText(inputText)

3: clickChat()

3.1: displayChat(text)
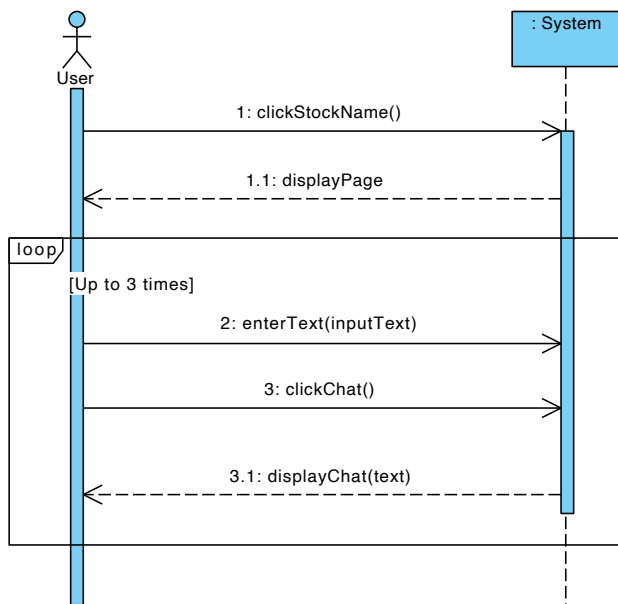
Operation: clickStockName
Cross Reference: Use Cases: chatThread
Pre-conditions: User is on the Home page
Post-conditions: User is on the desired stock's page

- User is redirected to the stock's page
- InputBox displayed on page

Operation: enterText(inputText)
Cross Reference: Use Cases: chatThred
Pre-conditions: User is on the desired Stock's page
Post-conditions: text is input into input box

- text is input into the text box

Operation: clickChat()
Cross Reference: Use Cases: ContactHelp
Pre-conditions: User has input text in inputBox on desired Stock's page
Post-conditions: Text is in

- Chat has been created (instance creation)
- chat is displayed on the chat thread (association formed)

User

1: clickContact()

1.1: displayPage

2: clickEmail()

2.1: initializeEmailDraft

: System
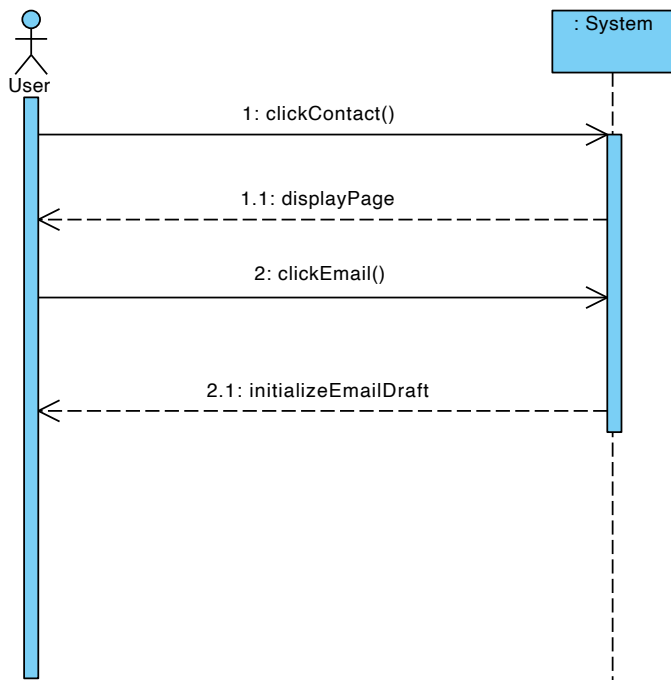
Operation: clickContact()
Cross Reference: Use Cases: ContactHelp
Pre-conditions: User is on the webpage
Post-conditions: User is on the Contact Page

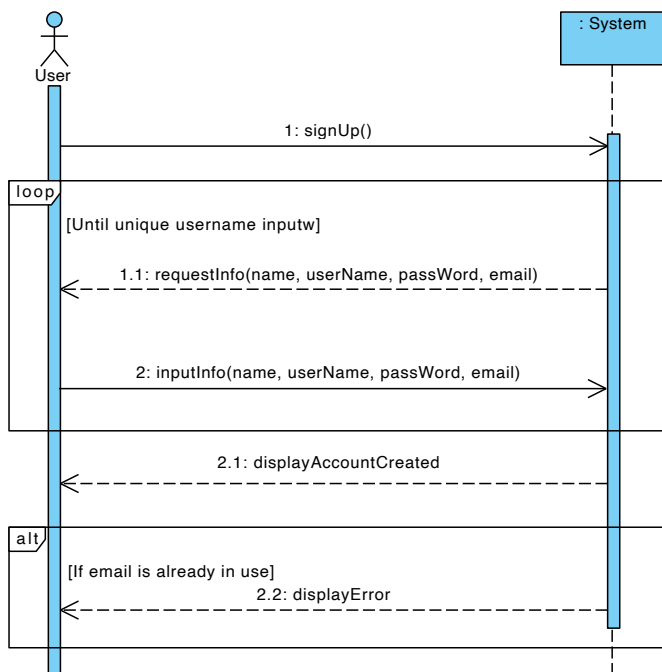- User gets redirected to the contact page

Operation: clickEmail()
Cross Reference: Use Cases: ContactHelp
Pre-conditions: User is on the Contact page
Post-conditions: Draft to email address is initiated

- Email draft is initiated (association formed)

```
        ○
       /|\          : System
       / \        ┌─────────┐
      User        └─────────┘
                       │
       │  1: signUp()  │
       ├──────────────▶│
  ┌loop┐               │
  │                    │
  │ [Until unique username inputw]
  │                    │
  │ 1.1: requestInfo(name, userName, passWord, email)
  │◀ ─ ─ ─ ─ ─ ─ ─ ─ ─ │
  │                    │
  │ 2: inputInfo(name, userName, passWord, email)
  ├──────────────────▶│
  └────────────────────┘
       │ 2.1: displayAccountCreated
       │◀ ─ ─ ─ ─ ─ ─ ─│
  ┌alt┐                │
  │ [If email is already in use]
  │         2.2: displayError
  │◀ ─ ─ ─ ─ ─ ─ ─ ─ ─ │
  └────────────────────┘
```

Operation: signUp()
Cross Reference: Use Cases: Account Creation
Pre-conditions: user is on webpage
Post-conditions: User is request to add info

● User is directed to input information

Operation: inputInfo(name, userName, passWord, email)
Cross Reference: Use Cases: Account Creation
Pre-conditions: User has requested to sign up
Post-conditions: User has an account with YoloTrader

● An account has been created for the user (instance created)

```
                                                                    : System
       User
                                 1: deleteAcccount()

                                1.1: requestPassword

                                 2: inputPassWord(pass)

                                2.1: validatesPassword

   alt
            [If invalid password]
                                 2.2: displayError
```
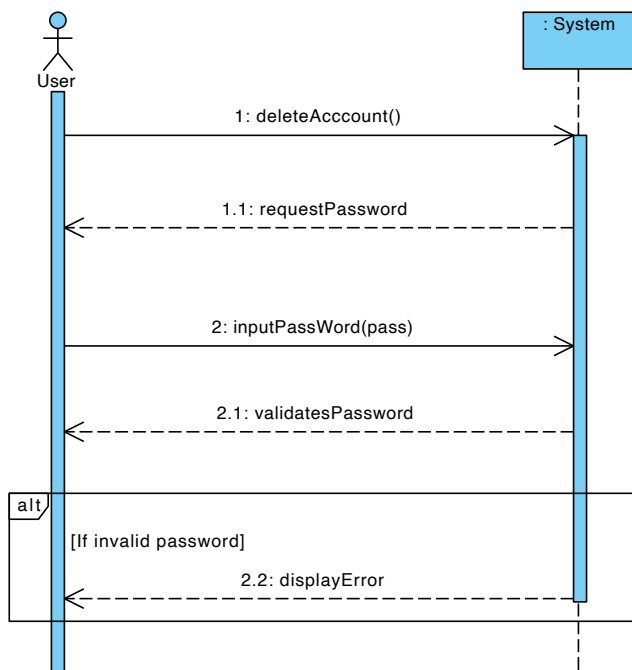
Operation: deleteAccount()
Cross Reference: Use Cases: Account Deletion
Pre-conditions: user has an account with YoloTrader
Post-conditions: System prompts user for password

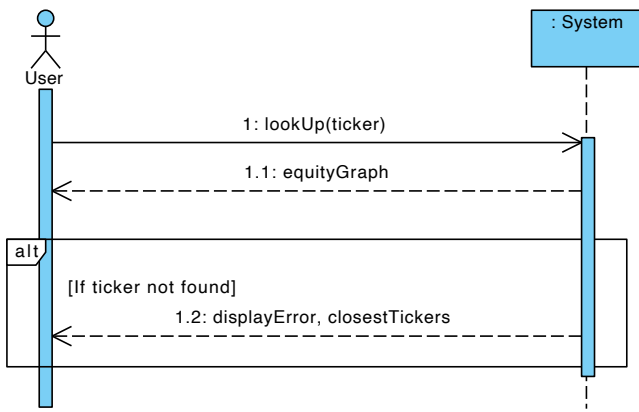● System prompts user to enter password to confirm

Operation: inputPassWord(pass)
Cross Reference: Use Cases: Account Deletion
Pre-conditions: user has requested to delete account
Post-conditions: System verified password and account is deleted

● Account deleted from database (instance deletion)

```
: Actor                              : System

         1: inputName(username)
    |──────────────────────────────>|
    |                                |
         1.1: displayUser
    |<- - - - - - - - - - - - - - - -|
    |                                |
  alt |                             |
    | [If username is NOT valid]    |
    |         1.2: displayError      |
    |<- - - - - - - - - - - - - - - -|
```
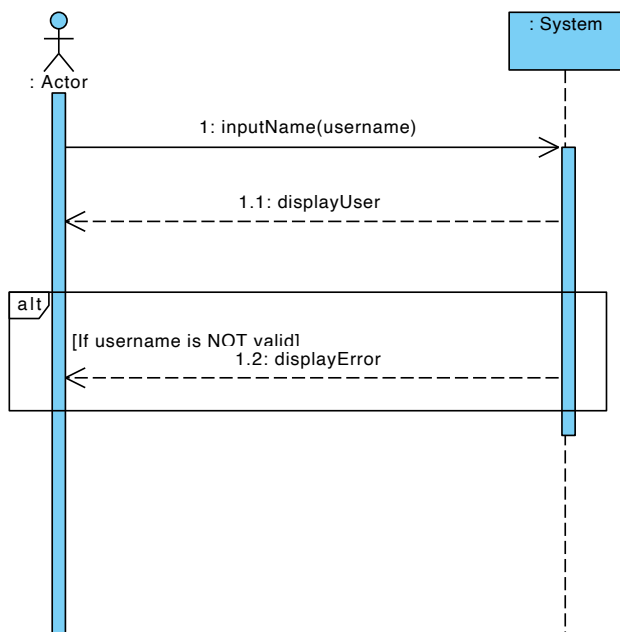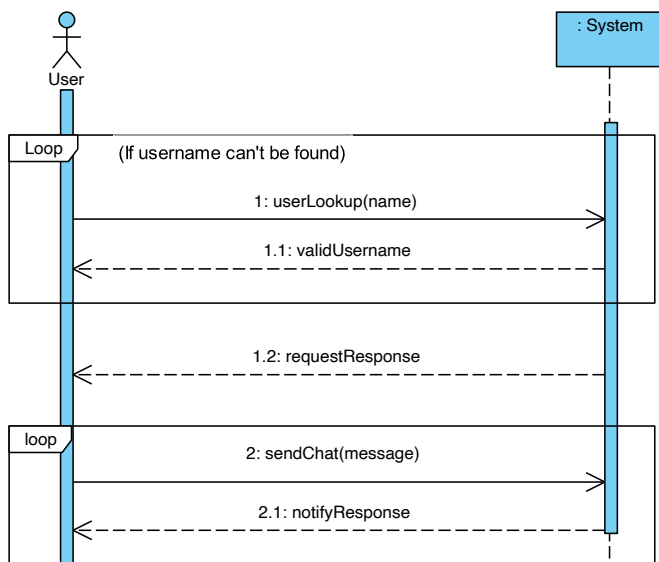
Operation: inputID(Id Number)
Cross Reference: Use Cases: FindFriend
Pre-conditions: User is on the webPage
Post-conditions: User finds friend

● Stock page associated with ID is displayed

User

: System

1: lookUp(ticker)

1.1: equityGraph

alt

[If ticker not found]

1.2: displayError, closestTickers

Operation: lookUp(ticker)
Cross Reference: Use Cases: Stock Lookup
Pre-conditions: User is on webpage
Post-conditions: User has found desired stock

● equity graph of stock is displayed

User

: System

Loop (If username can't be found)

1: userLookup(name)

1.1: validUsername

1.2: requestResponse

loop

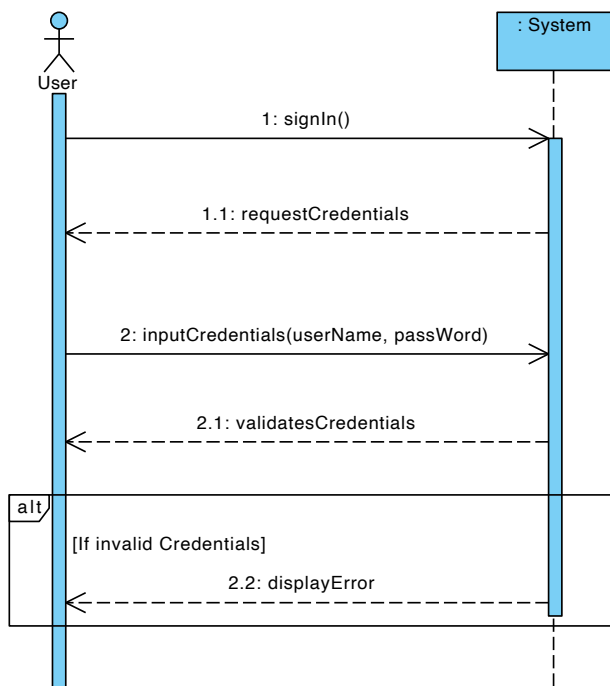2: sendChat(message)

2.1: notifyResponse

Operation: userLookup
Cross Reference: Use Cases: Private Message
Pre-conditions: User wants to message another user
Post-conditions: Chat request sent to other user

- If invalid username is entered, ask for another username
- Other user must accept request before chatting can begin

Operation: sendChat
Cross Reference: Use Cases: Private Message
Pre-conditions: User has been given access to message other user
Post-conditions: Other user responds to user

- User is notified when a message is received

User

: System

1: signIn()

1.1: requestCredentials

2: inputCredentials(userName, passWord)

2.1: validatesCredentials

alt

[If invalid Credentials]

2.2: displayError

Operation: signIn()
Cross Reference: Use Cases: Sign In
Pre-conditions: User has a registered account
Post-conditions: User is prompted for credentials

● System prompts user to enter credentials

Operation: inputCredentials(userName, passWord)
Cross Reference: Use Cases: Sign In
Pre-conditions: User has has been prompted for credentials
Post-conditions: User is logged in

● User is on account (association formed)

| Time Log |

TIMECARD

	OWEN MURPHY	
DATE	**HOURS**	**DESCRIPTION**
09-17	1	Created Schedule
09-19	4	Refined Schedule, contributed to Requirements list, designed Wireframes, updated web pages, setup business email
9-21	1	Created Use Case
9-23	0.5	Updated webpage by centering headers
9-26	1	Finished 3rd Use Case, created SSD's, System Operations, Operations Contract
9-28	2	Domain Model, Iteration I
9-29	6	Refining Domain Model, Use Case Diagram, SSD, requirements
9-29	4	Learning git commands, repositories, etc., schedule, use cases,

TOTAL: 19.5

TY WICKS		
DATE	**HOURS**	**DESCRIPTION**
9-19	1	Created Requirements Diagram
9-20	.5	Reviewed Wireframe, helped revise schedule
9-23	1	Created Use Cases
9-27	1.5	Created SSD's and Domain Model
9-28	3	Updated Domain Model, finalized use cases and created remaining SSD's

TOTAL: 6

PRINCE KALU		
DATE	**HOURS**	**DESCRIPTION**
9/20	3	Learning Backend Development & finding the best environment to create on.
9/21	1.5	Begun backend development.
9/21	2	Worked on sketches for iteration one & begun making a slideshow powerpoint.
9/26 - 9/27	7	Added detail to the front end of the home page, connected the front end to back end, researched API's, tested several API's for responsiveness and accuracy & learned Javascript.
9/28-9/29	3	Iteration one SSD's and Ticket System, helping with domain model, updating github, as well as finishing the slideshow.

TOTAL: 16.5

Tickets

Prince Kalu

1. Getting accurate price and other analytical data from market exchanges [Semi-Resolved]
   a. Getting accurate data from Binance(cryptocurrency) [Resolved]
   b. Getting accurate data from stock exchanges [Unresolved]
2. Creating a equity graph for displaying price data [Semi-Resolved]
   a. Detailed equity graph with plenty of options is up, however the price data is not as accurate as I would like.
3. Fetching relative news for each ticker [Unresolved]
4. Accounting for earnings for each ticker [Unresolved]
5. Connecting backend to front end [Semi-Resolved]
   a. Using Javascript to fetch data(JSON), I tested it and the console log is responding to the JSOn data, however I need to learn more about middleware.
6. Gathering/Creating the Position Costs Distribution, constructing a level 2 order book from the level 1 market data, Identifying market support and resistance [Unresolved]
7. Implementing my trading algorithm, based off technical analysis [Unresolved]
8. Setting up a basic machine learning framework [Unresolved]
9. Setting up and fine tuning a confidence meter based off the machine learning framework. [Unresolved]
10. Training & testing the machine [Unresolved]
11. Testing and fine tuning the accuracy of the machine's predictions [Unresolved]
12. Training & testing the machine post fine tuning [Unresolved]

| # Commits |

1 commit to github to host webpage

| Suggested Point Distribution |

Point distribution:

Out of 100 total points:
- Prince -> 33.3
- Owen -> 33.4
- Ty -> 33.3