Report 1
for
"Titan Trading"

Report 1: Part 3

Group #7

Steven Adler
Kristene Aguinaldo
Timothy Liu
Nicholas Lurski
Avanish Mishra
Safa Shaikh
Brooks Tawil
Kristian Wu
David Zhang

supervised by
Dr. Ivan Marsic

February 18, 2018

Contents

Chapter 1

Customer Statement of Requirements

1.1 Problem Statement

Dear Group 7, Here at the Rutgers Investments Co., our employees trade on the stock market everyday and have allowed our company to survive in the competitive field. With our current success, we are looking to train the next generation of traders and finance experts. We believe that the best way to learn about trading is by participating yourself. With experience, traders will know the right stocks to choose that can yield an average of above 10With Titan Trading, investors, traders, and students of all experience levels will have a chance to interact with the stock market in a fun, competitive manner without using their actual finances. We want our game to teach the novice trader the basics of investments and give them a platform to try things out before committing to investing in real life. For the more experienced day-trader, our game will give them a chance to try out some new investment strategies, or compete against their colleagues without risking their real assets. We can bring together traders of all levels in a no risk environment where they can really learn about investment trading. The market includes a wide range of assets from commodities, bonds, currency, etc. each of which follow different trends and trading practices. On top of the diverse investment options, there are also the tools, which include market limits, buy/sell orders, dividend pay comparisons, and graphs and data visualization tools, that investors use in their work on the market floor. Even for an experienced trader, the stock market is a complex and ever growing challenge, so being able to practice and learn using real data and tools, without risking personal loss, is what we want our game to provide. So far, what were asking you to do has already been done by a lot of different companies, and we want to take our product to the next level in order to give our users a more engaging way to practice trading. First, we want this program to be like a game where you compete against other people, and depending on your standings, you can earn rewards. By including the competition aspect, we hope to incentivize players to make strategic and profitable investments, instead of making risky and unrealistic trades. Furthermore, we want to have a universal league of traders that includes every user of our product. This, along with the competitive nature of the game, will provide for a more realistic trading environment for the players participating in our game. We also want to allow users to create private leagues so they can compete with people their friends or colleagues. This will allow people to participate in competitions with people of their own skill level rather than all the other people in the world. We hope that this encourages beginners to begin learning about trading with their friends, and eventually participate in real trading. The second thing we want our product to include is to incorporate cryptocurrency trading. Today, cryptocurrencies have taken the spotlight with their exponential growth rates and the news cycle dominated by cryptocurrency analysis once Bitcoin peaked at 19,500 dollars per coin. Bitcoin is just one of the many cryptocurrencies available to the public, albeit one of the most well known ones. Cryptocurrencies still rather new to our society, but anyone can invest in them and certain coins have shown 20,000 times return for the investor. Now that the cost for one coin is so high, finding a good entry point to investing is difficult so we want to give our users the ability to practice trading cryptocurrency before deciding whether or not they want to make the investment in real life. Although the whole premise of this project is just a game, there must be definite rules to guarantee fair play. In terms of leagues, every user registered on the website will enter a universal league. In this league, each user enters the stock market (without cryptocurrency) with the same buying power. To ensure that other users are not unjustly punished for playing in market slumps, this universal league will operate in seasons. At a season start, any currently registered user on the site will automatically be registered into the universal league for that season, and anyone that registers during a current season of the universal league will not be able to participate until the next season starts. The rules for the universal league should also be made to mirror the real-world as closely as possible. Other private leagues can be made by users, and these leagues will have preferences set by the league creator to dictate the rules for that game. Rules for one league may seem cheap or unfair to other players, so allowing private, customizable leagues is important to ensure that everyone plays their own fair game. These rules can include: the exclusion of certain stocks, the time needed to conduct transactions, interest, dividends, loans, and more. This product needs to solidify competitiveness to ensure game longevity. For all leagues, leaderboards will be included to rank players. Each players profile will record their league rankings as a way to show off his or her hard work. A universal leaderboard will also be included that factors in all seasons of the universal league as a way for users to compare each other across the website. Rewards for placing well within notable leagues will also be included, such as site announcements and extra features. Special leagues will also be provided, which only invite users that have placed extraordinarily well in the universal league. This will give a desire for users to differentiate themselves from the competition by providing a more intense league. Although the primary motive to play the game is for the competition between users, the product should also cater to beginners to ease the learning curve. After a user has registered on the site and is currently playing in a league, the website will have several pages that serve as the game manual and help guide. These pages will display videos that explain the stock market, how to buy and sell stocks, and other general stock market essentials. Leagues other than the universal league will also be provided specifically to beginners to meet and play with other eager, newer users. The overall aim of this game is to both educate users about the stock market and to provide a competitive environment for enthusiasts to demonstrate their market skills, while not being as cutthroat as the real world market. The advantage over current stock market games will be the ability to trade cryptocurrencies, a more refined competitive aspect, and a beginner-friendly environment.

Sincerely, The Rutgers Investments Co.

1.2 Glossary of Terms

League – A collection of aspiring investors participating in an instance of a stock market simulation. Each league will come with its own pre-determined rule set and goals. A winner will be determined at the end of a season.

- **Universal** – Every investor will be automatically slated into this league. It will involve all active users on the program.
- **Private** – League will be created and maintained by a League Administrator.

League Manager – The creator of a private league. He/she will be responsible for inviting participants and designating the ruleset including victory conditions and timespan. Users are only admins for the league he/she has created.
Cryptocurrency – A form of virtual currency that uses cryptography to secure transactions, regulate the generation of units of currency, and verify transfer of assets. It is susceptible to volatile prices.
Investor – One who puts money to use through purchases with the expectation of profits.
Assets – Property owned by company or person, in our case, the user. It is regarded as having value.
Commodity – A basic good such as a food, metal, or agricultural product that investors can buy and sell.
Bonds – A fixed monetary investment that an investor loans for a defined period of time at a specific interest rate.
Buy Order – Tool in the game used to purchase stock or cryptocurrency. It will automatically handle acquisition of assets. Buying power will update to reflect this transaction.
Sell Orders – Tool in the game used to sell stock or cryptocurrency. It will automatically handle the sales of assets. Buying power will update to reflect transaction.
Limit Orders – Tool in the game used to set a maximum purchase price or minimum sale price for assets. If these conditions are not met, the transaction will not occur.
Dividends – A distribution of a companys earnings to a class of its shareholders. It can be given as cash payments, stock shares, or property.
Buying Power – Amount of free spending money available for use for an investor.
Market Slumps – A period of time where share prices and trading volume are reduced due to declining stock prices.
Season – A division of the year marked by changes in weather and daylight hours. This will be used as the length of time for the Universal League. Approximately three months.

Chapter 2

System Requirements

2.1 User Stories

Identifier	User Story	Point Value
ST-1	As a user, I can create an account by registering my email address with the website in order to save my information and play history.	2
ST-2	As a user, I can login to my account through a login page so I can verify my credentials via my username and password.	1
ST-3	As a user, I can view an About page that details information about the team who developed Titan Trading to learn more about the motivation behind this product.	2
ST-4	As a user, I can view an Instruction/FAQ page that details different features of the application in order to receive guidance on how to play the game.	2
ST-5	As a user, I will automatically be entered into a universal league - league in which all users compete - with a fixed starting balance so I can immediately start gameplay and make transactions.	4
ST-6	As a user, I can join a league either by creating a new league and becoming a league manager or by being invited to a league by another league manager, with the exception of the universal league.	4
ST-7	As a user, I can create a private league with up to 20 members and automatically be designated as a league manager upon league creation in order to manage league details and rules.	3

ST-8	As a league manager, I should be able to set rules for my league, including a starting balance, and league name, in order to manage gameplay among a smaller group.	3
ST-9	As a league manager, I should be able to invite other users by email address or username to start a competition with additional players in my league.	2
ST-10	As a league manager, I would like to create private cryptocurrency leagues that only allow trading of cryptocurrency in order to isolate the volatile market of cryptocurrency from the regular market. This can be designated as a setting during league creation.	4
ST-11	As a league manager, I can set an end date for each of my privately owned leagues in order to define the end of the game. The player with the most money wins.	1
ST-12	As a league manager, I can post or send announcements to the members of my league.	3
ST-13	As a user, I can access a Dashboard page that allows me to see the leagues I am competing in, my rank and balance in each league, and my statistics so I can keep track of my progress.	3
ST-14	As a user, I can view a page with current stock market trends so I can make an educated decision about which stock to buy.	3
ST-15	As a user, I can change my personal information and avatar to personalize what my profile looks like to other users and how it is displayed on leaderboards.	3
ST-16	As a user, I can view a leaderboard for each of my leagues that ranks the top 10 people with the most money in the league, and it must update once a day at midnight.	4
ST-17	As a user, I should receive notifications for selected stocks to know when they have exceeded or dropped past a threshold value set by me.	5
ST-18	As a user, I can access a transaction page in order to buy new stocks and sell stocks that I own.	9
ST-19	As a user, I can search companies either by their full name or acronym so I do not have to spend extra time searching for a company.	2

ST-20	As a user, I can receive accomplishment trophies once I reach specific milestones and achievements within the universal league so I am motivated to continue playing.	4
ST-21	As a visiting user, I should be able to view the home page, as well as view the statistics for the universal league (even if I do not have an account) in order to view the competition the game provides.	2
ST-22	As a site administrator, I can change universal league settings and view and delete inactive leagues to remove unwanted data.	3
ST-23	As a site administrator, I can access data about site statistics, including the number of users, active and non-active leagues, and transaction details in order to display accurate statistics on my website.	3
ST-24	As a user, I can view a glossary page that will tell me the key terms needed to start investing.	1
ST-25	As a user, I can view an investing guide page that will tell me how the stock market and cryptocurrency work and how to invest properly.	1

2.2 Nonfunctional Requirements

2.2.1 Functional

In order to be able to accommodate for a growing user base we will be creating our web application around a Postgresql database, which is known in the software industry for its ease of scale. Additionally we will be creating private leagues that up to 20 people can join with their friends in order to make the game more personal for users. We will also be using password hashing and salting in order to protect user credentials.

2.2.2 Usability

We will be using Django, which is a Python web-application platform in order to create our websites Model View Controller. Along with this we will be using Bootstrap to create a good looking website that will be smooth and easy to navigate and use.

2.2.3 Reliability

We will be prototyping our website on a heroku instance, and deploy it to Amazon Web Services (AWS) for maximum uptime. All transactions will be performed and verified server side and not client side so that someone could not cheat and manipulate the trades going on. We will also keep backups of user data in case of a server failure.

2.2.4 Performance

Since we are using Postgresql, we have the ability to scale up and provide for a large number of users on the fly with great ease. Additionally, python makes efficient web-server applications so we will not have to worry all that much about demands on the server.

2.2.5 Supportability

Python provides a lot of debug tools, and provides a lot of information when things go wrong which will benefit us greatly when debugging our program. We will also be able to provide users with error messages when things do go wrong so that they know that their trade has not gone thru or various other errors that could happen.

2.3 On-Screen Appearance Requirements

On-screen appearance requirements detail requirements related to the user interface experience and are written from an objective perspective. These requirements have been separated into two categories: all site requirements and page-to-page requirements. All site requirements are those that apply to every single page across the website domain, while page-to-page requirements are those specific to different web pages.

We have brainstormed the formats for 5 different types of web pages: the homepage, dashboard page, individual league page, buy page, and sell page. We also intend to include a custom 404 not-found page, FAQ page, glossary page, and instruction page, whose requirements are also listed below.

2.3.1 All Site Requirements

Identifier	Requirement	Points
OSR-1	Every page will have navigational tabs at the top of the screen to access the home, about (dropdown with about us, FAQ page, instruction page, glossary), and dashboard, as well as a drop down menu to access their profile settings or log out.	1
OSR-2	Every page should contain a stock-ticker footer showing growing and declining stocks.	1
OSR-3	Every page will have the logo of Titan Trading in the top left corner.	1

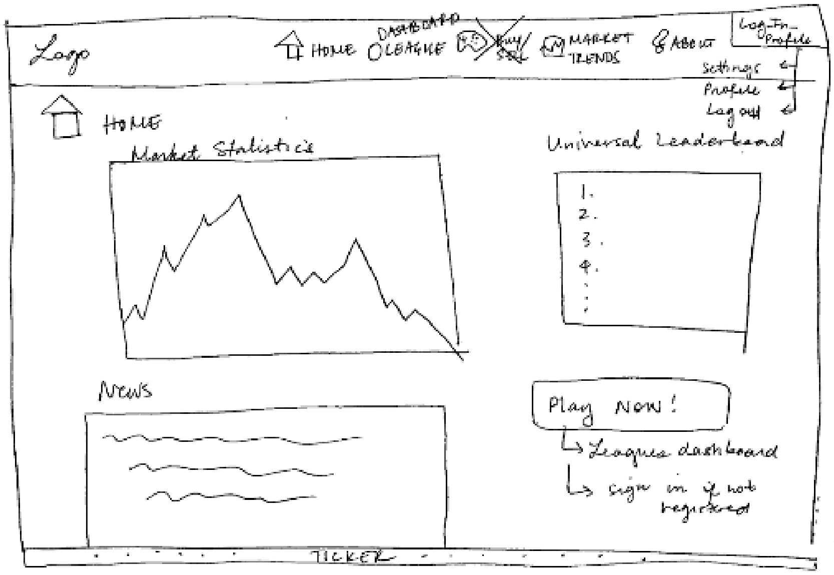
2.3.2 Page to Page Requirements

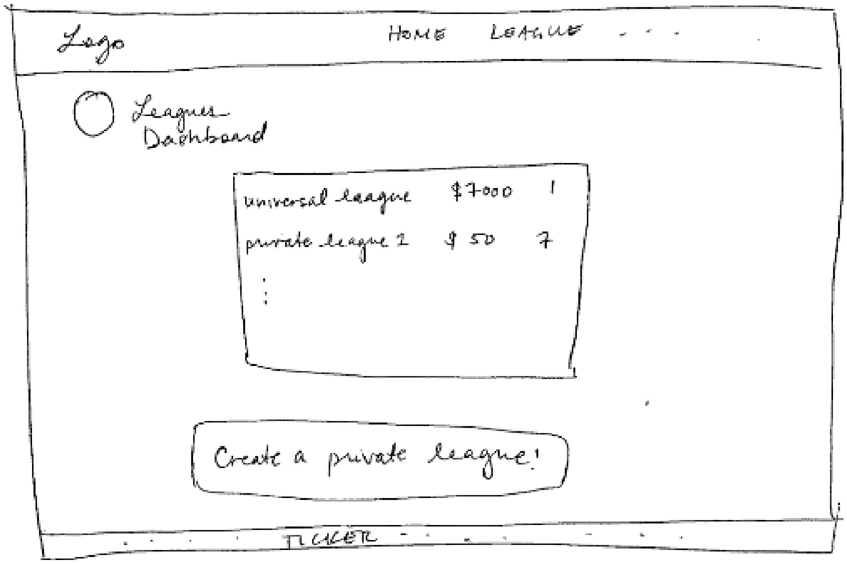
Identifier	Requirement	Points
OSR-4	A custom 404 not found page will be displayed to a user when they try to access a URL/URI that does not exist or is not designed for them to be accessing.	1
OSR-5	The homepage should include current market trends and articles, as well as the leaderboard for the universal league.	3
OSR-6	The dashboard page should contain a list of leagues the user is competing in and a button to create a new private league.	2

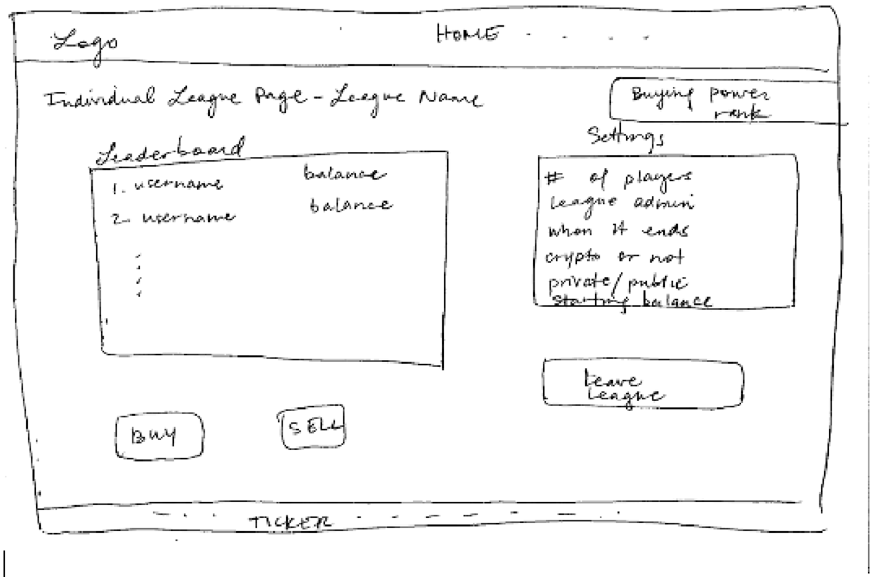
OSR-7	Every individual league page should present information about the user's current buying power, total worth, and rank within the league.	2
OSR-8	Every individual league page should present buttons to buy or sell stocks.	1
OSR-9	Every individual league page should present a leaderboard of the top ranking players within the league and a list of league settings, eg: league end date, starting balance, league administrator, etc.	3
OSR-10	The buy page should present information about the user's current buying power, total worth, and rank within the league.	2
OSR-11	The buy page should contain a widget that will return market statistics about the stock the user searched for.	2
OSR-12	The buy page should present a form that allows the user to submit buy orders.	2
OSR-13	The sell page should present information about the user's current buying power, total worth, and rank within the league.	2
OSR-14	The sell page should have a portfolio of all the stocks the players owns within a specific league and should have a button to submit sell orders.	4
OSR-15	The About page should include the mission statement behind TITAN TRADING.	4
OSR-16	The FAQ page should present answers to common questions such as, ”What is the Universal League? What is a private League? I'm new to investing, is there a place I can go to learn more about stocks? What is cryptocurrency and how is it involved in our game?”	3
OSR-17	The glossary page will have a list of trading terms and definitions that will be sorted alphabetically.	2
OSR-1	The instruction page should have screenshots of how to buy/sell stocks,create/delete private leagues, buy/sell cryptocurrency within the game.	3.5
OSR-19	Each user should have a profile page to display notifications, trophies, and username.	2

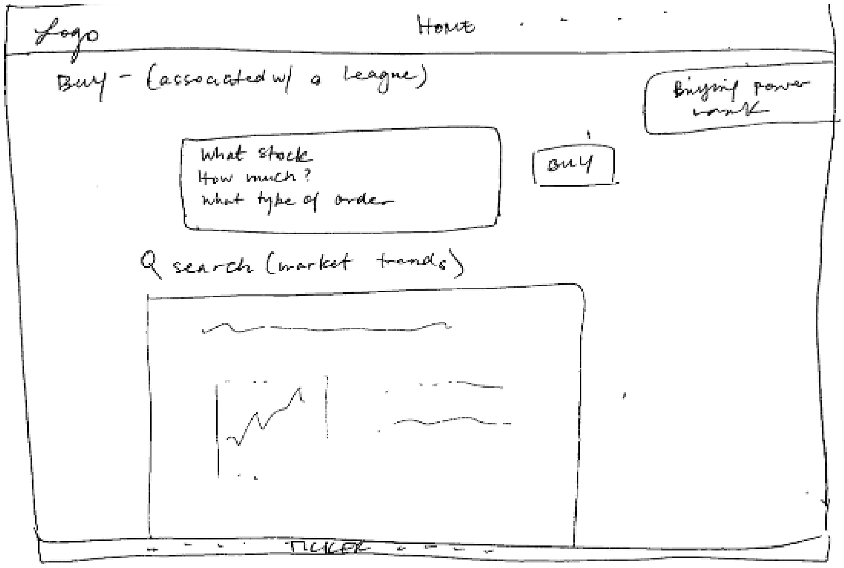
2.3.3 Sketches of User Webpages

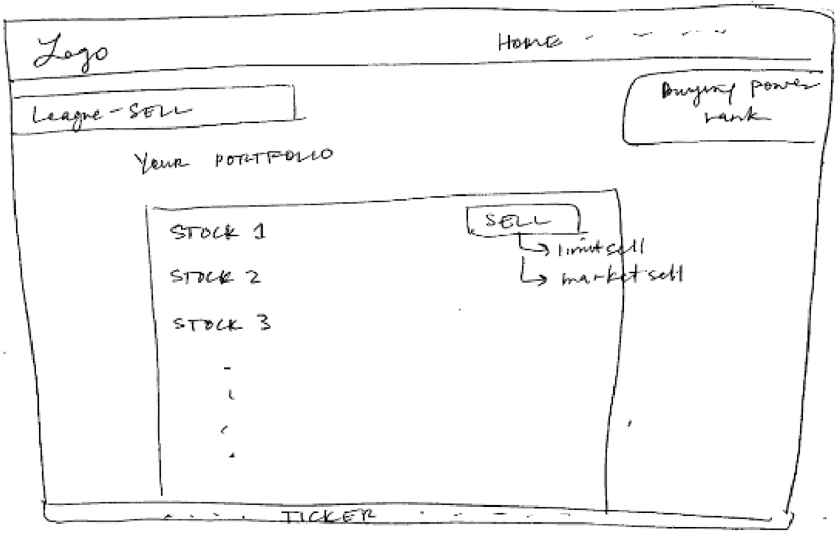
Brainstormed sketches of Titan Trading webpages are shown below. They include: the homepage, the leagues dashboard page, an individual league page, buy page, and sell page.


Logo
HOME
DASHBOARD
LEAGUE
~~BUY/ SELL~~
MARKET TRENDS
ABOUT
Log-In Profile
Settings
Profile
Log off
HOME
Market Statistics
Universal Leaderboard
1.
2.
3.
4.
News
Play Now!
Leagues dashboard
sign in if not registered
TICKER


Logo
HOME
LEAGUE
Leagues Dashboard
universal league
$7000
1
private league 1
$ 50
7
Create a private league!
TICKER

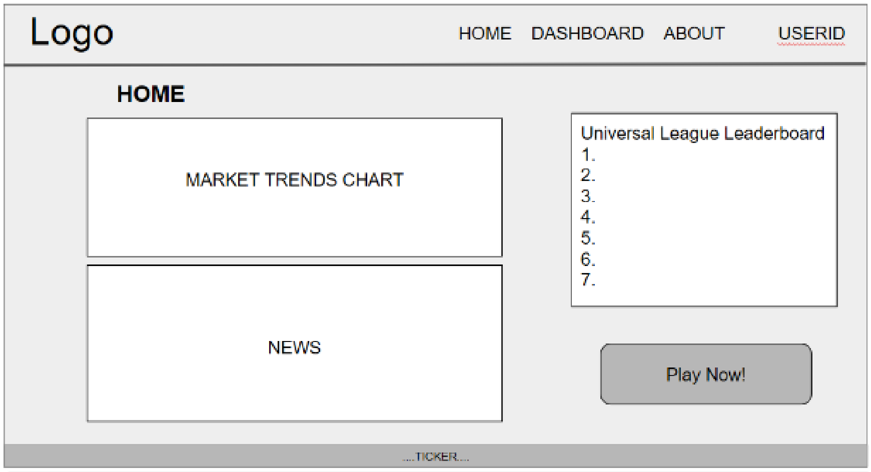

Logo
HOME
Individual League Page - League Name
Buying power
rank
Leaderboard
1. username
balance
2. username
balance
Settings
# of players
League admin
when it ends
crypto or not
private/public
starting balance
Leave League
BUY
SELL
TICKER


Logo
Home
Buy - (associated w/ a League)
Buying power
rank
What stock
How much?
what type of order
BUY
Q search (market trends)
TICKER

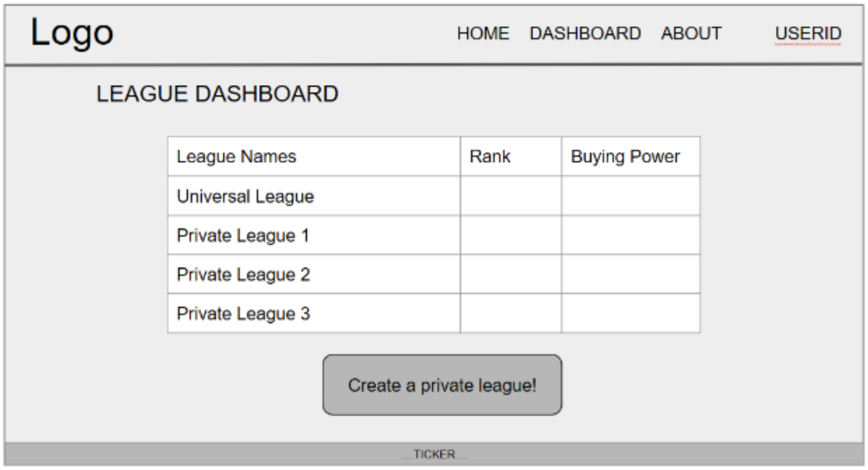

Logo
Home
League - SELL
Buying power
rank
YOUR PORTFOLIO
STOCK 1
STOCK 2
STOCK 3
SELL
limit sell
market sell
TICKER

2.3.4 First Draft UI Electronic Representation

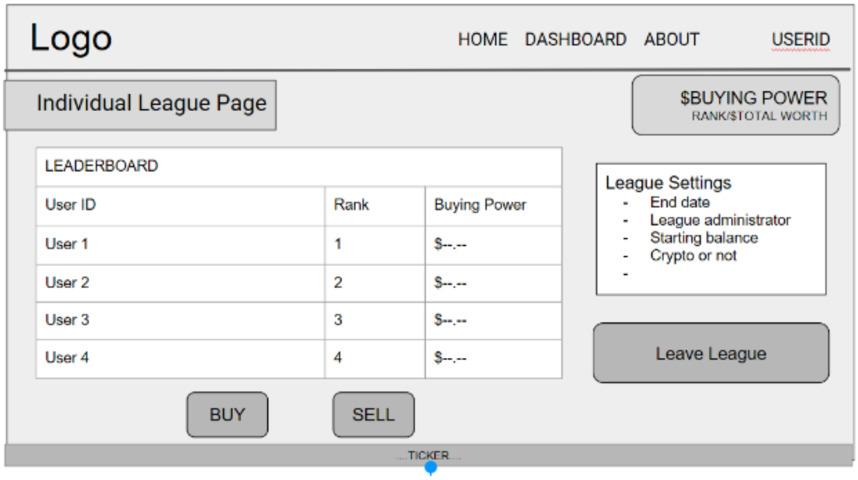
After finalizing a first draft of the UI layout, we made an electronic representation.



Universal League Leaderboard
1.
2.
3.
4.
5.
6.
7.

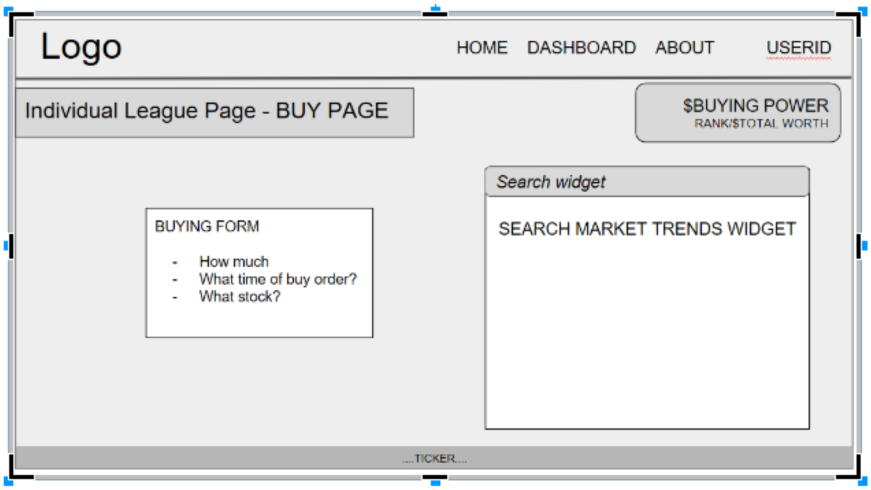


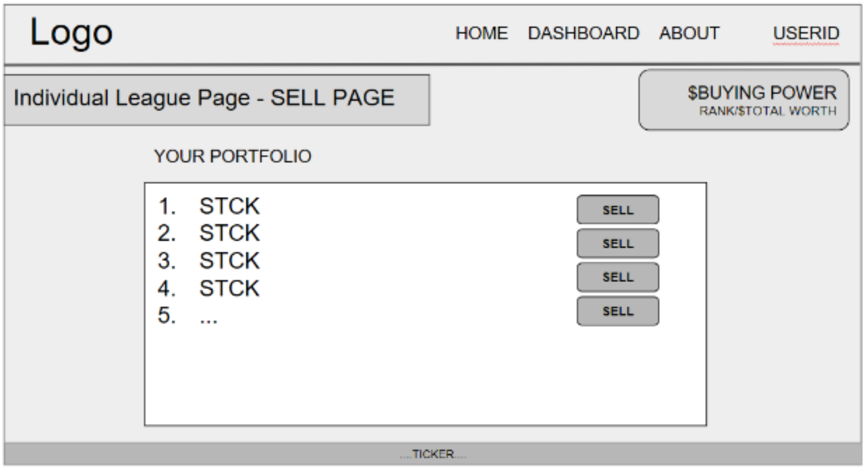
League Names	Rank	Buying Power
Universal League		
Private League 1		
Private League 2		
Private League 3		



LEADERBOARD		
User ID	Rank	Buying Power
User 1	1	$--.--
User 2	2	$--.--
User 3	3	$--.--
User 4	4	$--.--

League Settings
- End date
- League administrator
- Starting balance
- Crypto or not
-





2.4 Acceptance Tests

Acceptance tests that the customer will run to check that the system meets the requirements are as follows. Note, however, that these test cases provide only a coarse description of how a requirement will be tested. Acceptance tests will follow a naming convention of AT-X.Y where X is the number of the requirement that is being tested, and Y is the number for the given test. Following the requirements table the following acceptance tests have been formulated:

Acceptance Tests for ST-1

AT-1.1 – Register into the system using a valid email address and fill out required information. (Pass: New account information is registered in database)

Acceptance Tests for ST-2

AT-2.1 – Using a pre-existing account, log in using a correct email and password.(Pass: User information is able to be pulled based on the logged in user)

AT-2.2 - Attempt to log in with an incorrect email and password. (Pass: The system rejects an incorrect combination of email and/or password)

Acceptance Tests for ST-3

AT-3.1 - Have a test user navigate to the about page and attempt to give a summary of the product based on information on the about page. Match this user description to some standard description given by the development team. (Pass: The test user description matches closely to the desired developer description)

Acceptance Tests for ST-4

AT-4.1 - Have a test user, unfamiliar with the product, navigate to the FAQ and attempt to understand and answer their questions using the given FAQ. (Pass: No questions that are obvious are left unanswered and not on the FAQ)

Acceptance Tests for ST-5

AT-5.1 - A new user is automatically entered into the universal league, and their statistics are viewable on a global leaderboard. (Pass: User A is able to see the standing of User B on the global leaderboard)

AT-5.2 - A given logged in user is able to check their own standing in the universal league. (Pass: User A can navigate to the global leaderboards and see their standing)

Acceptance Tests for ST-6

AT-6.1 - A league administrator can invite others to join the league over email.

AT-6.2 - A league administrator can invite others to join the league using an existing username.

AT-6.3 - Non pre-registered users can sign up for the website and join a league they were invited to over email.

AT-6.4- Pre-registered users can sign up for the website and join a league they were invited to over email or through their existing username. (Pass cases all involve seeing a user either register and be shown asa player in the league, or an already registered user is put into the league.)

Acceptance Tests for ST-7

AT-7.1 - Ensure that any user can create their own league, becoming the league administrator. (Pass: the user creating the league is placed asd a player and is given league administrative permissions.)

AT-7.2 - League should be limited to the 20 player limit.

Acceptance Tests for ST-8

AT-8.1 - Preferences and settings for a given league are set upon creation. (Pass: The league admin can set the settings of the league upon its creation.)

AT-8.2 - Certain preferences for a given league are changed after the creation of the league. (Pass: Settings not critical to be set at the creation can be adjusted.)

Acceptance Tests for ST-9

See AT-6.3 and AT-6.5

Acceptance Tests for ST-10

AT-10.1 - Ensure that when creating a league, the administrator is able to specify the use of cryptocurrencies in the league. (Pass: The setting is marked and the league will use cryptocurrencies instead of stock information)

Acceptance Tests for ST-11

See AT-8.1 and AT-8.2 fro preference setting. This is simply another preference to set and ensure that it is active.

Acceptance Tests for ST-12

AT-12.1 - League administrator can write up their own message or use a predefined message for certain actions.

AT-12.2 - Announcements can be sent by the league administrator to the players in the league (Pass: Users receive notification and message from the admin)

Acceptance Tests for ST-13

AT-13.1 - A user can navigate from any point on the website to their leagues page.

AT-13.2 - Ensure that a user can clearly see the leagues that they are a part of.

AT-13.3 - Ensure that a user get statistics and information regarding each league.

(Pass: For each of these the pass is straightforward and involves proper displaying of the information for a given user by the system and the ability for the user to navigate the UI.)

Acceptance Tests for ST-14

AT-14.1 - Statistics about individual stocks can be navigated to.
AT-14.2 - Market trends can be navigated to and viewed using graphs and data visualization.
AT-14.3 - A user can navigate the UI and easily reach the market trends page.

Acceptance Tests for ST-15

AT-15.1 - From the user account page, view the current profile picture and allow for selection of a new profile picture.
AT-15.2 - User should also be allowed to change their display name.
AT-15.3 - Other personal information on the account page should be changeable. (Pass: Successful change of any of the information on the account)

Acceptance Tests for ST-16

AT-16.1 - A specific leagues leaderboard can be navigated to and is displayed to the user.
AT-16.2 - The timing system updates the current standings at midnight of each trading day, or at the end of the trading day. (Pass: League A is given new standings and updates automatically based on the time of day)

Acceptance Tests for ST-17

AT-17.1 - Notification of stock price drops below/above a certain level, automatically set. (Pass: Uses holding the stock or interested in a stock are notified when an pre-set threshold is met.)

Acceptance Tests for ST-18

AT-18.1 - Users can navigate to the transactions page.
AT-18.2 - Users can buy stock from the transactions page.
AT-18.3 - Users can sell stock from the transactions page.

Acceptance Tests for ST-19

AT-19.1 - Search functionality supports stock tickers and full company names. (Pass: The system can successfully return the company page based on a predetermined list of companies of varying popularity)

Acceptance Tests for ST-20

AT-20.1 - A users achievements can be navigated to and viewed.
AT-20.2 - Notifications for getting an achievement are properly distributed and displayed.
AT-20.3 - Achievement progress is properly tracked and displayed.

Acceptance Tests for ST-21

AT-21.1 - A user that is not logged in is served the home page.
AT-21.2 - The home page displays the global leaderboard, which a non-logged user can still view. (Pass: A non-logged in user is served no user specific data and is instead given the standard homepage.)

Acceptance Tests for ST-22

AT-22.1 - Site administrators have ultimate ability to view and delete leagues (Pass: A site administrator account deletes a league that they are not a part of)

Acceptance Tests for ST-23

AT-23.1 - Visualization about the backend active users, number of leagues, stock trades and transaction details can be accessed by any site admin. (Pass: An admin is able to view a transaction in any given league on multiple occasions. A site admin can view the data for any randomly given league.)

Acceptance Tests for ST-24

AT-24.1 - Glossary page can be navigated to from anywhere on the site.

Acceptance Tests for ST-25

AT-25.1 - Investing guide page can be navigated to from anywhere on the site.

AT-25.2 - The investing guide provides a good getting started point and offers plenty of resources for a novice trader to learn and grow. (Pass: Give a novice test user access to the guide and perform a user test. The guide should provide the novice a starting point where they would feel comfortable jumping into using the product.)

Chapter 3

Functional Requirements Specification

3.1 Stakeholders

The main stakeholders of Titan Trading are novice traders who want to learn more about the stock market and gain trading experience. Through this application, they will be able to participate in the stock exchange without using real currency. Through their support and use of our site, we will be able to better simulate the stock market and create a more genuine experience. In addition to those using Titan Trading for learning, there are those who can use this application as a teaching tool. Teachers or people teaching others about the stock market can use Titan Trading to expose others to the stock market and how trading works. Furthermore, Rutgers Investments Co. is a stakeholder of Titan Trading. Since they are the ones commissioning the creation of this application, the success of Titan Trading directly affects the investment they made in the program. On a similar note, we, the developers of this application, are stakeholders of Titan Trading since we are invested in the creation of the program. The success or failure of Titan Trading affects our reputation as programmers, and whether or not the application is popular is a reflection to our abilities to create a good UI and program.

3.2 Actors and Goals

3.2.1 Actors

Primary Actors

For every use case there are direct actors who are involved in the use case/user story

Site Administrator– Owns the website and can edit the user interface and universal league settings. Is a league manager for the universal league.

User – Someone who has an account and can become a player by joining leagues.

League Manager – A user who owns a private league and can adjust any settings for the league.
Player – A user who is a member of a league.
Universal Player - A user is by default a player in the universal league.
Private League- A player profile created for a private league and has a player ID unique to this league.

Guest - Someone who visits the website without creating an account. Can view our homepage containing market statistics.

Secondary Actors

The secondary actors participates in the use case but does not initiate it. Subtypes include supporting actors and offstage actors. They help achieve the goal or must know about the outcome.

Finance API - Service to request current market data and stock prices. Includes information about stock time series data, cryptocurrencies, stock technical indicators, and sector performances. We plan to use the Alpha Vantage API.
Database - Stores all information related to app usage, including user data, leagues, league statistics (leaderboard, player rank), and user-league associations.
Browser - Provides a user interface for users to interact with app HTML pages.
Widgets - Website modules that display relevant data, including news, stock information, and market trends.

3.3 Use Cases

Our application has two primary modes of access - guest and user. Guests can view the homepage, About page, and FAQ page. To use the full functionality of our application, guests sign up through a login page. Once signed up, users are automatically entered into a universal league, managed by the site administrator. Users can also create their own private leagues and invite other users to private leagues.

Potential use cases:
* indicates that the use case is important and should be described in a Fully-Dressed manner.

1. *Create a league (User)
2. Sign up (Guest)
3. Login (User)
4. Logout (User)
5. *Invite user to League (Manager)
6. *View statistics (Guest/User)
7. *Buy stock (Player)
8. *Sell stock (Player)
9. *View Dashboard and Leaderboards (User)
10. Accept/decline league invitation (User)
11. Search stock (Player)

12. *View FAQ/Instructions (User)
13. Unlock Trophy (User)

3.3.1 Casual Descriptions

***Note: ST-23 is related to all use cases because it is a general requirement for the system administrator.

UC1 - Create a league (ST-6,7,8,10,11,22)

A user can create a new private league, and select it as a cryptocurrency league or regular stock league, set the starting balance, and set an end date for the league. The end date shall dictate who has the most money from all players in the league, declaring the winner.

UC2 - Sign Up (ST-1,5)

A visitor can create a new account by clicking the Sign Up button on the homepage. To create an account, the visitor registers with their email address and chooses a username and password. After account creation, the now user will automatically be entered in a universal league.

UC3 - Login (ST-1,3)

After account creation, a visitor can log into an account through a login page by entering in a valid username and respective password. The system will update the website to reflect the visitor is now a user.

UC4 - Logout (ST-1)

The user can log out of an account by clicking the Logout button on the website. The system will redirect the user to the homepage and reflect that the user is now a visitor.

UC5 - Invite User to League (ST-6,7,9)

A league manager can invite users to a league by clicking Add Users on their league page. Then, the league manager will enter in the desired users emails. The system will email invites to these users, which the users can accept via hyperlink. If the email is not registered, the user can create an account directly from the invite.

UC6 - View Statistics (ST-14,21)

A user or guest can view market statistics by clicking the Home button. The system will pull the current market trends from the Finance API and redirect the user to the homepage which will display these trends.

UC7 - Buy Stock (ST-5,18)

A user can buy stocks on any league by clicking the buy button for the league. The system will redirect the user to a buy form, which the user will fill out with their desired stock name and quantity to purchase. The system will return the order total from the Finance API, and the user can submit the purchase given they have enough capital. The system will reflect the transaction in the database.

UC8 - Sell Stock (ST-5,18)

A user can sell stocks on any league by clicking the sell button for the league. The system will load a page displaying the users league portfolio. The user can then click the sell button from any stock in the portfolio and fill out the form to complete the selling process. The system will reflect the transaction in the database.

UC9 - View Dashboard and Leaderboards (ST-5,12,13,15,16,17,20,21)

A user can view the dashboard by clicking the Dashboard button, which lets the system redirect them to the dashboard page. From there, the user can select a league to view the league page, which includes the leaderboard.

UC10 - Accept/Decline (ST-6, ST-9, OSR-19)

A user can accept or decline any pending league invites from league managers on the users respective profile page. To accept or decline, click the respective button and the system will reflect the decision.

UC11 - Search Stock (ST-18, ST-19, OSR-11)

On the buy page, a user can search for a stock through the search widget. The system will match the users search against the names and symbols of stocks from the Finance API, and it will either display matching stocks or a failed search error message.

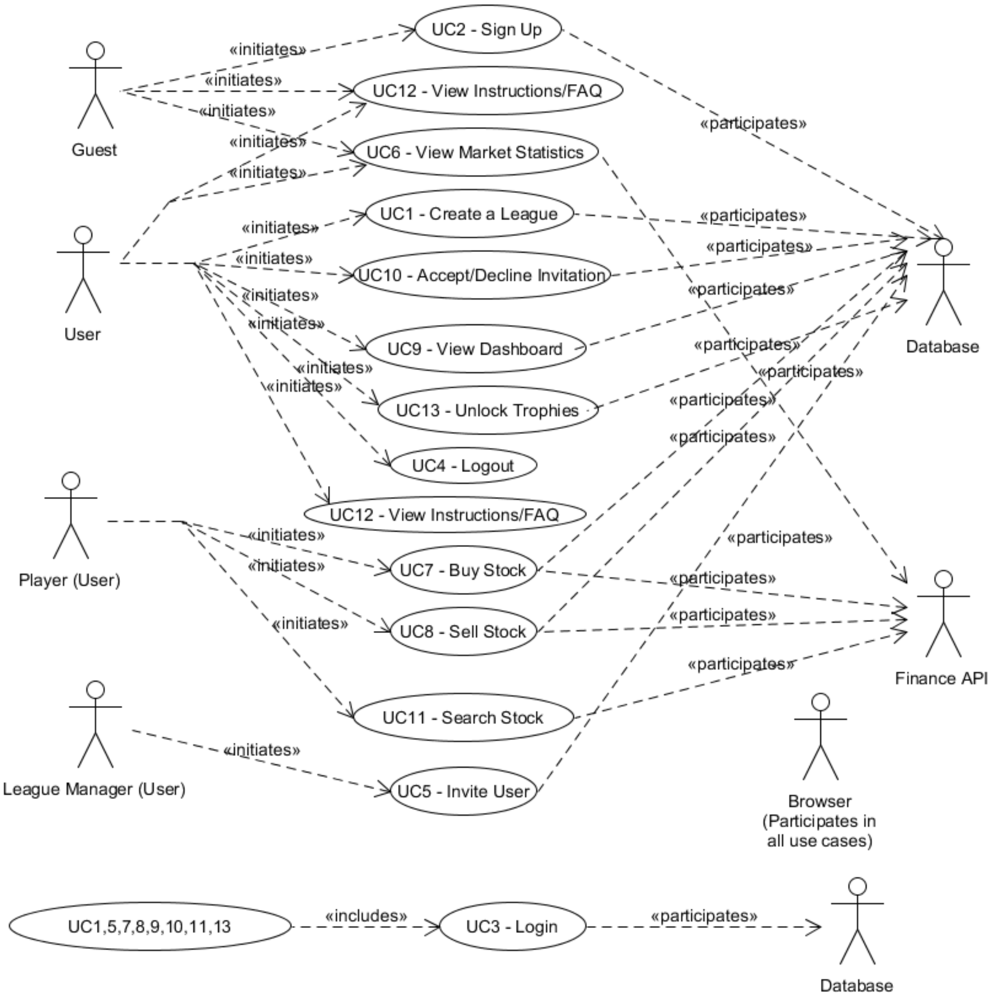
UC12 - View Instructions/FAQ (ST-3,4,24,25, OSR-15, OSR-16, OSR-18)

A user can hover over the About button, which will show buttons for the instruction and FAQ pages. The user can then click the chosen page to be redirected to that page.

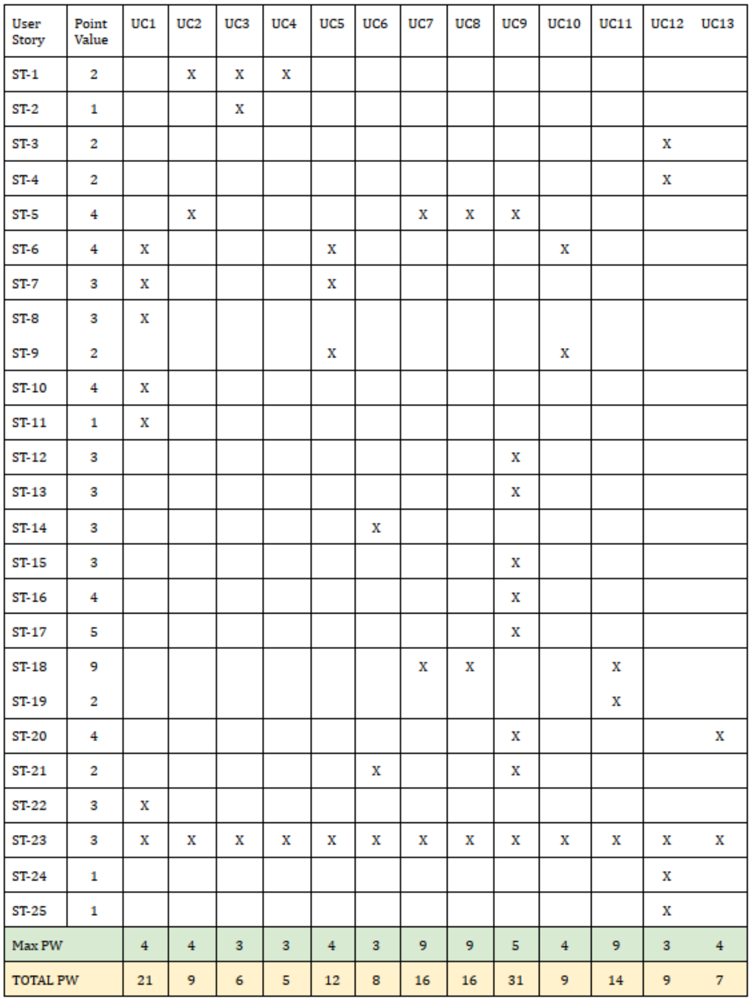
UC13 - Unlock Trophy (ST-20)

A user will receive trophies after reaching milestones in the universal league or overall performance (e.g. doubling money in a league.).

3.3.2 Use Case Diagram



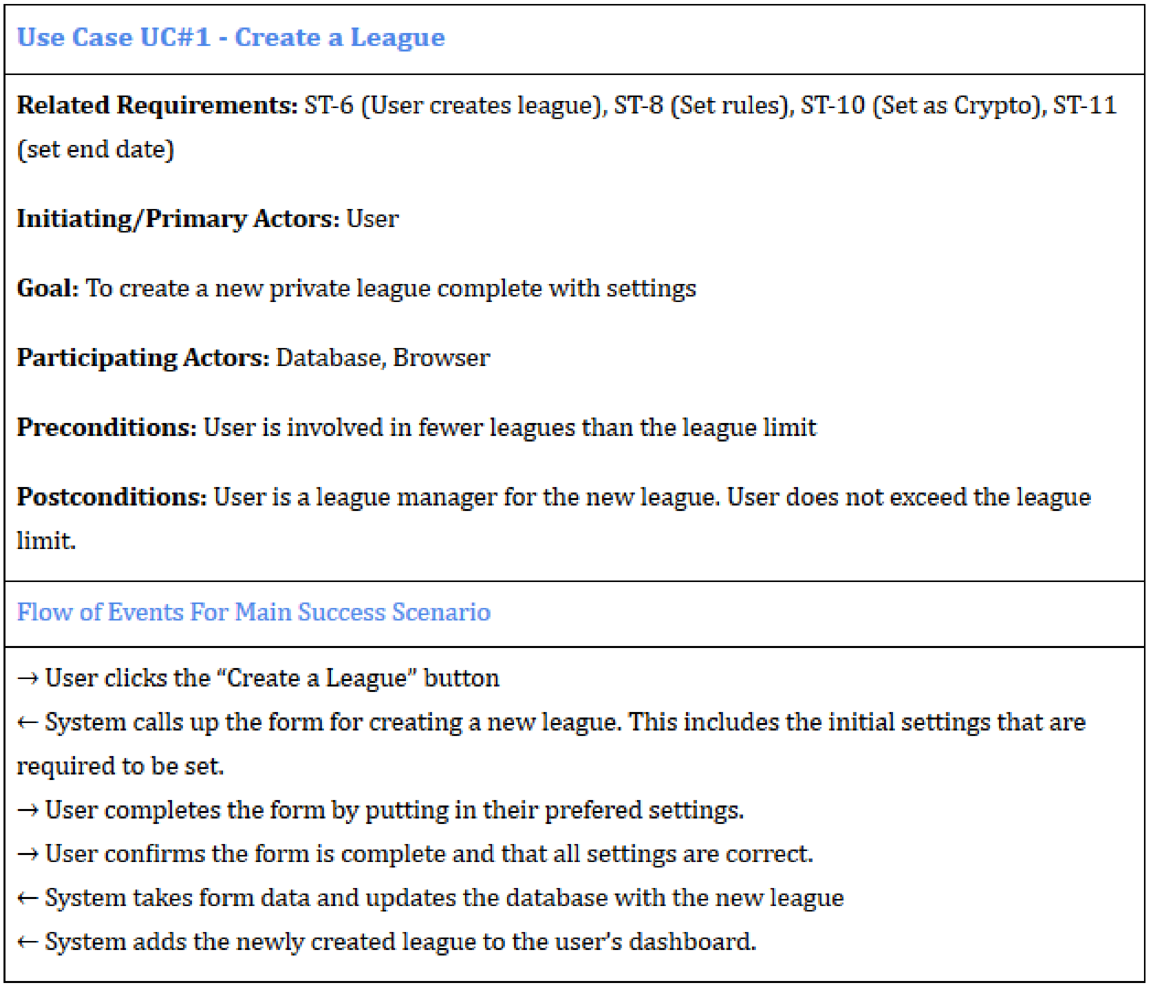
3.3.3 Traceability Matrix



User Story	Point Value	UC1	UC2	UC3	UC4	UC5	UC6	UC7	UC8	UC9	UC10	UC11	UC12	UC13
ST-1	2		X	X	X									
ST-2	1			X										
ST-3	2												X	
ST-4	2												X	
ST-5	4		X					X	X	X				
ST-6	4	X				X					X			
ST-7	3	X				X								
ST-8	3	X												
ST-9	2					X					X			
ST-10	4	X												
ST-11	1	X												
ST-12	3									X				
ST-13	3									X				
ST-14	3						X							
ST-15	3									X				
ST-16	4									X				
ST-17	5									X				
ST-18	9							X	X			X		
ST-19	2											X		
ST-20	4									X				X
ST-21	2						X			X				
ST-22	3	X												
ST-23	3	X	X	X	X	X	X	X	X	X	X	X	X	X
ST-24	1												X	
ST-25	1												X	
Max PW		4	4	3	3	4	3	9	9	5	4	9	3	4
TOTAL PW		21	9	6	5	12	8	16	16	31	9	14	9	7

3.3.4 Fully Dressed Use Cases

Use Case 1



Use Case UC#1 - Create a League

Related Requirements: ST-6 (User creates league), ST-8 (Set rules), ST-10 (Set as Crypto), ST-11 (set end date)

Initiating/Primary Actors: User

Goal: To create a new private league complete with settings

Participating Actors: Database, Browser

Preconditions: User is involved in fewer leagues than the league limit

Postconditions: User is a league manager for the new league. User does not exceed the league limit.

Flow of Events For Main Success Scenario

→ User clicks the "Create a League" button
← System calls up the form for creating a new league. This includes the initial settings that are required to be set.
→ User completes the form by putting in their prefered settings.
→ User confirms the form is complete and that all settings are correct.
← System takes form data and updates the database with the new league
← System adds the newly created league to the user's dashboard.

UC#4

Use Case UC#4 - Logout

Related Requirements: ST-2(User login), ST-21(Viewing the homepage as a visitor)

Initiating/Primary Actors: User

Goal: To logout of the user's account and allow access for another account to login

Participating Actors: Browser

Preconditions: User is already logged on to their account on the website

Postconditions: User successfully logged out and returned to the visitor homepage

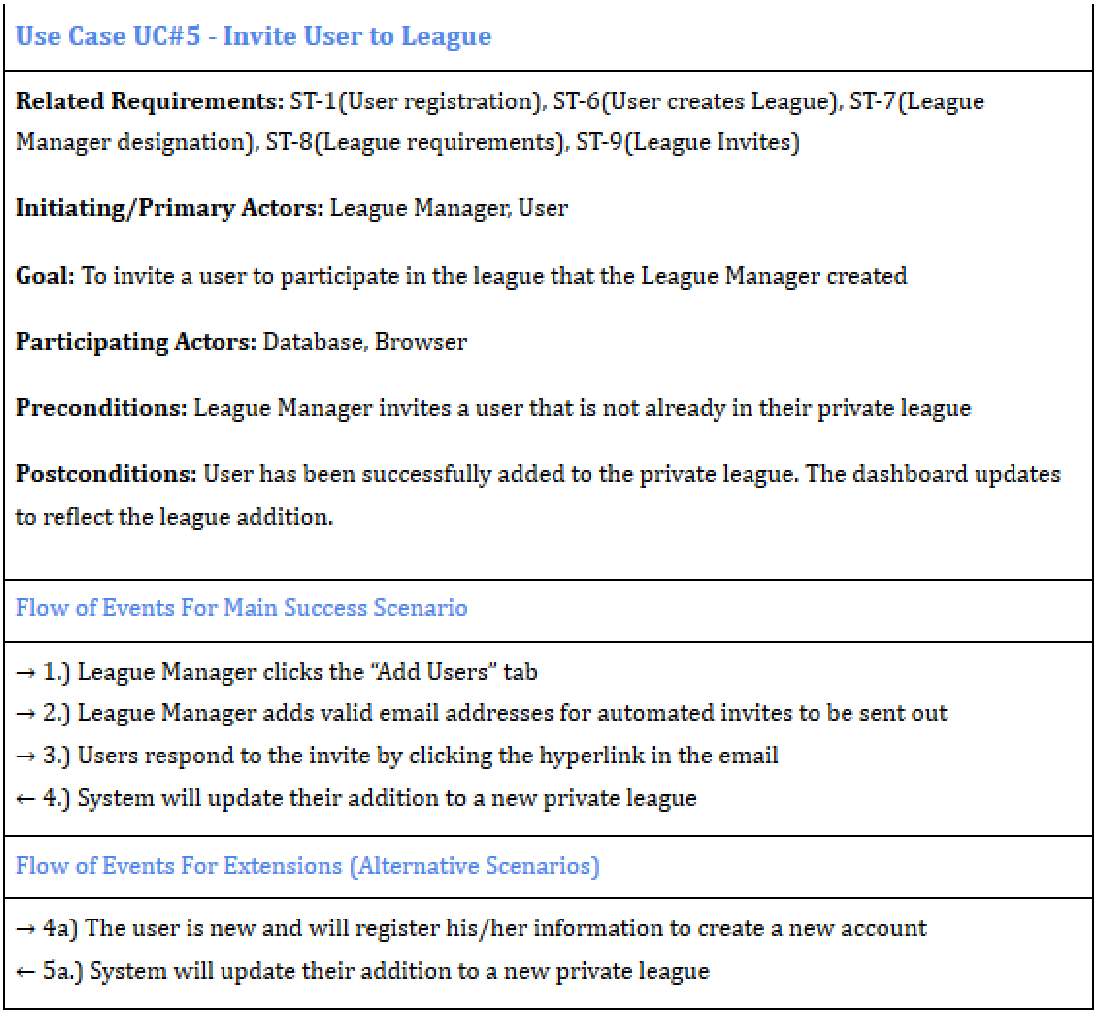
Flow of Events For Main Success Scenario

→ 1.) User clicks the "Logout" button

← 2.) System updates to reflect a visitor instead of a user

→ 3.)User is redirected to the visitor homepage

UC#5



Use Case UC#5 - Invite User to League
Related Requirements: ST-1(User registration), ST-6(User creates League), ST-7(League Manager designation), ST-8(League requirements), ST-9(League Invites) **Initiating/Primary Actors:** League Manager, User **Goal:** To invite a user to participate in the league that the League Manager created **Participating Actors:** Database, Browser **Preconditions:** League Manager invites a user that is not already in their private league **Postconditions:** User has been successfully added to the private league. The dashboard updates to reflect the league addition.
Flow of Events For Main Success Scenario
→ 1.) League Manager clicks the "Add Users" tab → 2.) League Manager adds valid email addresses for automated invites to be sent out → 3.) Users respond to the invite by clicking the hyperlink in the email ← 4.) System will update their addition to a new private league
Flow of Events For Extensions (Alternative Scenarios)
→ 4a) The user is new and will register his/her information to create a new account ← 5a.) System will update their addition to a new private league

UC#6

Use Case UC#6 - View Market Statistics
Related Requirements: ST-14 (View market statistics page), OSR-5 (Homepage) **Initiating/Primary Actors:** User, Guest **Goal:** To view the market statistics page **Participating Actors:** Finance API, Browser **Preconditions:** User/Guest is on any page of the website. **Postconditions:** User/Guest is on the market statistics page (homepage).
Flow of Events For Main Success Scenario
→ User clicks the “Home" button ← System pulls the current stock trends from the Finance API. ← System redirects user to the visitor homepage.

UC#7

Use Case UC#7 - Buying a Stock

Related Requirements: ST-18(Accessing a Transaction Page and performing a transaction)

Initiating/Primary Actors: User

Goal: To buy a stock for a league.

Participating Actors: Database, Browser, Finance API

Preconditions: User is on the League Dashboard page.

Postconditions: User has bought the desired stock and the Database reflects the purchase.

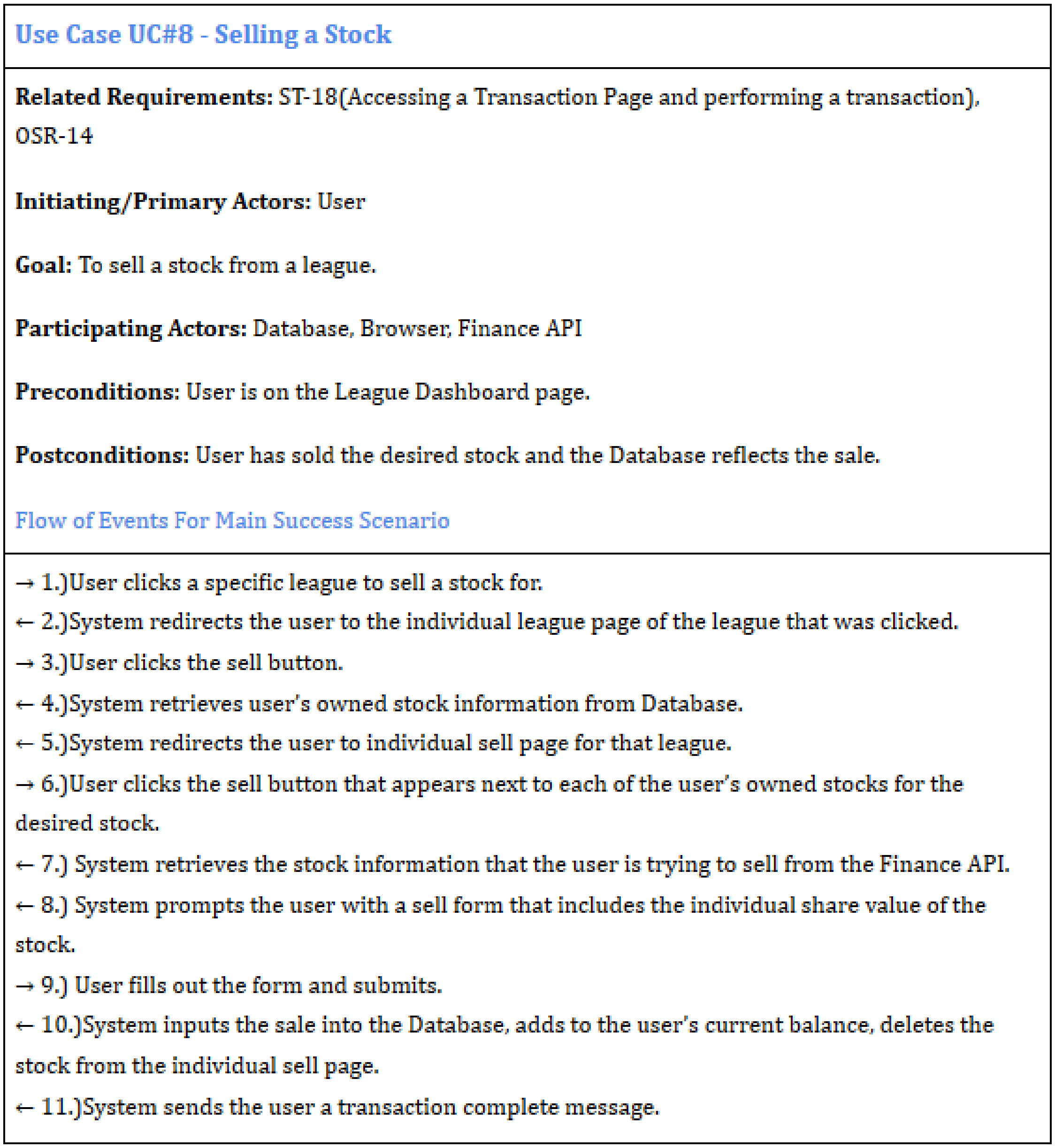
Flow of Events For Main Success Scenario

→ 1.)User clicks the specific league to buy a stock for.
← 2.)System redirects the user to the individual league page of the league that was clicked.
→ 3.)User clicks the buy button.
← 4.)System redirects the user to individual buy page for that league and renders a buy form..
→ 5.)User enters the stock and amount of shares desired for purchase.
← 6.) System pulls from the Finance API to display the price of each share and total price of the amount of desired shares.
→ 7.)User submits the information needed to buy a stock and can afford the stock.
← 8.)System inputs the purchase into the Database,subtracts from the user's current balance.
← 9.)System sends the user a transaction complete message.

Flow of Events For Extensions (Alternative Scenarios)

→ 7a.)User submits the information needed to buy a stock, but does not have enough money to afford the stock.
← 8a.)System sends the user a transaction failed message.

UC#8



Use Case UC#8 - Selling a Stock

Related Requirements: ST-18(Accessing a Transaction Page and performing a transaction), OSR-14

Initiating/Primary Actors: User

Goal: To sell a stock from a league.

Participating Actors: Database, Browser, Finance API

Preconditions: User is on the League Dashboard page.

Postconditions: User has sold the desired stock and the Database reflects the sale.

Flow of Events For Main Success Scenario

→ 1.)User clicks a specific league to sell a stock for.
← 2.)System redirects the user to the individual league page of the league that was clicked.
→ 3.)User clicks the sell button.
← 4.)System retrieves user's owned stock information from Database.
← 5.)System redirects the user to individual sell page for that league.
→ 6.)User clicks the sell button that appears next to each of the user's owned stocks for the desired stock.
← 7.) System retrieves the stock information that the user is trying to sell from the Finance API.
← 8.) System prompts the user with a sell form that includes the individual share value of the stock.
→ 9.) User fills out the form and submits.
← 10.)System inputs the sale into the Database, adds to the user's current balance, deletes the stock from the individual sell page.
← 11.)System sends the user a transaction complete message.

UC#9

Use Case UC#9 - Viewing Dashboard and League Leaderboards

Related Requirements: ST-13 (accessing league page and statistics), ST-16 (access specific private league leaderboards), OSR-7, OSR-9

Initiating/Primary Actors: User

Goal: To find league ranking information

Participating Actors: Database, Browser

Preconditions: User is on any page of the website.

Postconditions: User is successfully presented information on current league rankings.

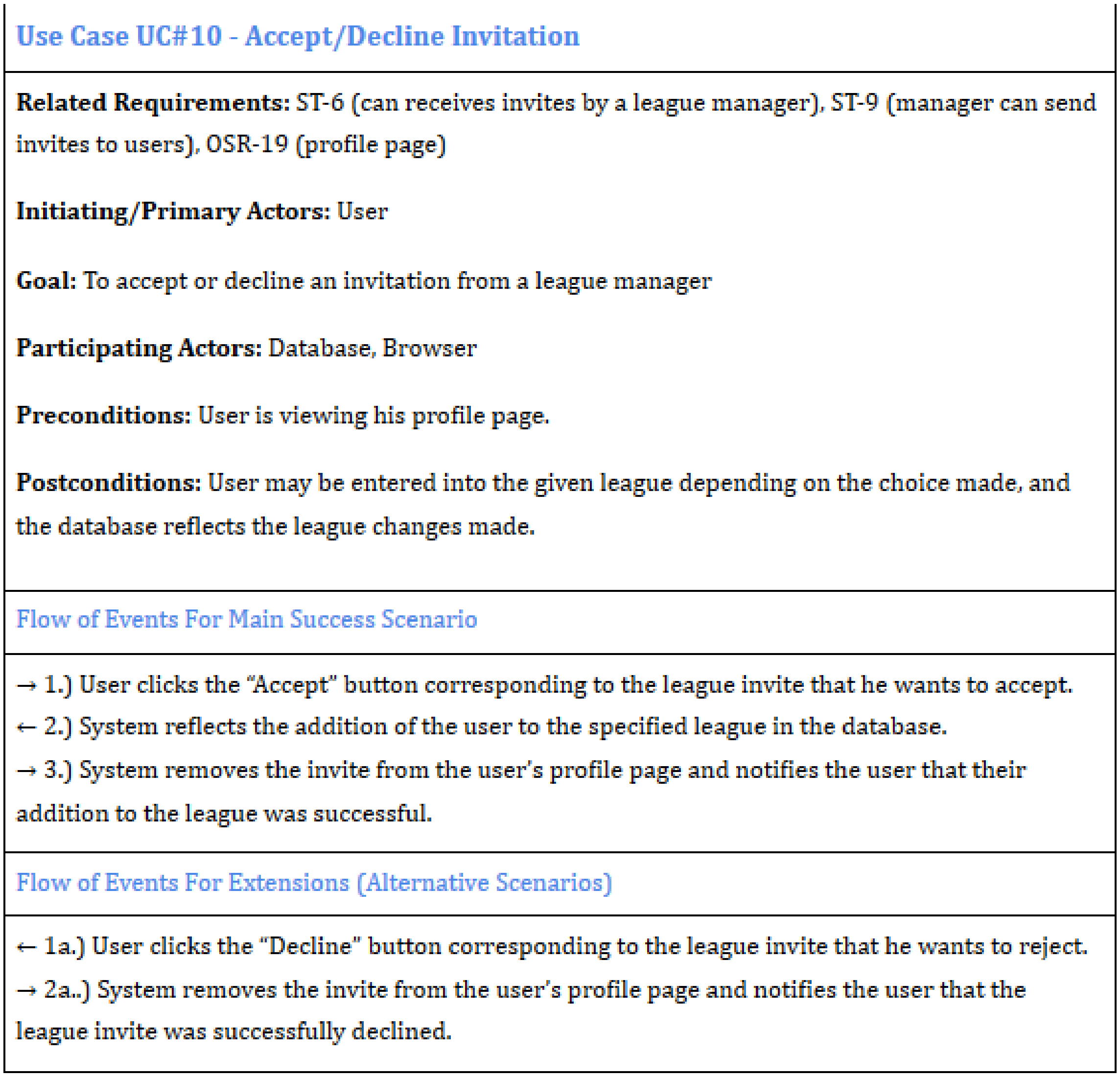
Flow of Events For Main Success Scenario

→ 1.) User clicks the "Dashboard" button.
← 2.) System redirects the user to the Dashboard page.
← 3.) System pulls information from the Database about the user's rank and buying power in each league.
→ 4.) User clicks the universal league.
← 5.) System redirects the user to the universal league individual page.
← 6.) System pulls leadership information from the Database about the universal league.

Flow of Events For Extensions (Alternative Scenarios)

→ 4a.) User clicks a private league.
← 5a.) System redirects the user to an individual private league page.
← 6a.) System pulls leadership information from the Database about the private league.

UC#10



Use Case UC#10 - Accept/Decline Invitation

Related Requirements: ST-6 (can receives invites by a league manager), ST-9 (manager can send invites to users), OSR-19 (profile page)

Initiating/Primary Actors: User

Goal: To accept or decline an invitation from a league manager

Participating Actors: Database, Browser

Preconditions: User is viewing his profile page.

Postconditions: User may be entered into the given league depending on the choice made, and the database reflects the league changes made.

Flow of Events For Main Success Scenario

→ 1.) User clicks the "Accept" button corresponding to the league invite that he wants to accept.
← 2.) System reflects the addition of the user to the specified league in the database.
→ 3.) System removes the invite from the user's profile page and notifies the user that their addition to the league was successful.

Flow of Events For Extensions (Alternative Scenarios)

← 1a.) User clicks the "Decline" button corresponding to the league invite that he wants to reject.
→ 2a..) System removes the invite from the user's profile page and notifies the user that the league invite was successfully declined.

UC#11

Use Case UC#11 - Search Stock

Related Requirements: ST-18 (Using pages that conduct transactions), ST-19 (buy stock), OSR-11 (Search widget)

Initiating/Primary Actors: Player

Goal: To search for a stock by name or symbol to purchase

Participating Actors: Finance API, Browser

Preconditions: Player is viewing the buy page for some league.

Postconditions: None

Flow of Events For Main Success Scenario

→ 1.) Player enters in their desired search terms into the search widget text box and presses search.
← 2.) System takes the search terms and pulls any stocks that match by name or symbol from the Finance API.
← 3.) System updates the player's search widget in the buy page with the matching stocks.

Flow of Events For Extensions (Alternative Scenarios)

← 2a.) System does not find any stocks that match by name or symbol from the Finance API.
← 3a.) System updates the player's search widget in the buy page with an error specifying the issue.

UC#12

Use Case UC#12 - View Instructions/FAQ

Related Requirements: ST-3, ST-4 (Can view an Instructions/FAQ page), ST-24(glossary), ST-25 (investing guide), OSR-15 (Titan Trading Lore), OSR-16 (FAQ answers), OSR-18 (Instruction/FAQ Page)

Initiating/Primary Actors: Guest, User

Goal: To view the Instructions/FAQ page

Participating Actors: Browser

Preconditions: Player is on any page of the website.

Postconditions: Player is viewing the Instructions/FAQ page.

Flow of Events For Main Success Scenario

→ 1.) Guest/User hovers over the "About" button.

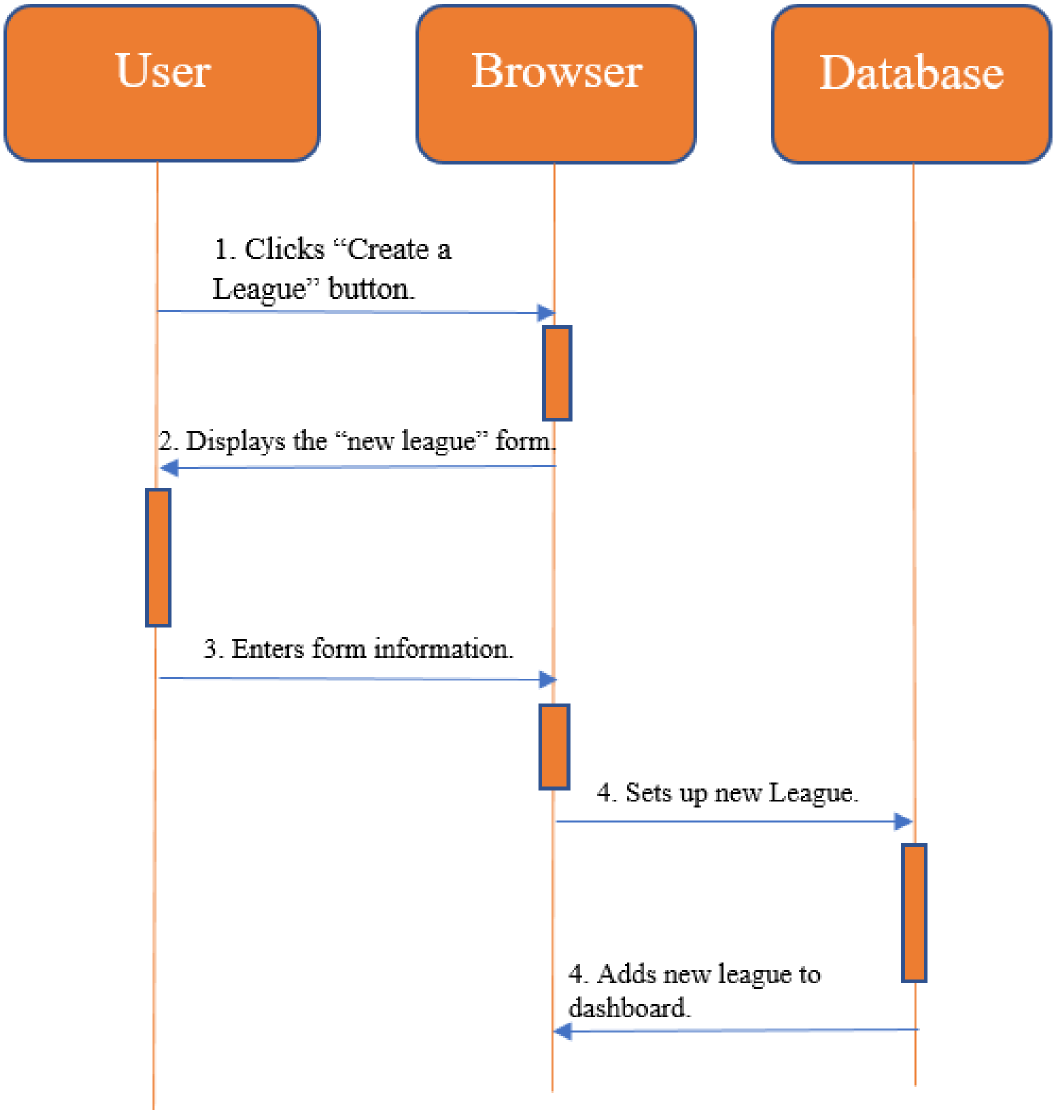
← 2.) System displays a dropdown menu with several pages, including the instruction and FAQ pages.

→ 3.) Guest/User clicks the chosen page.

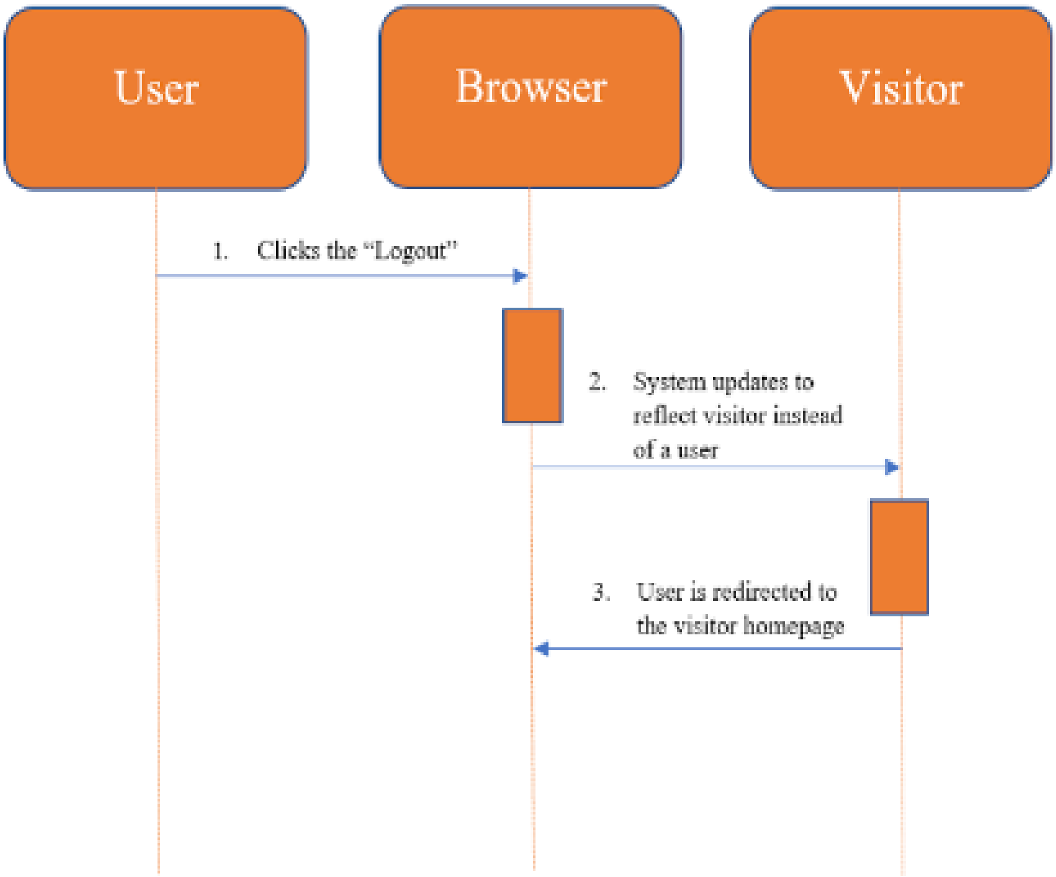
← 4.) System redirects the user to the specified page.

3.3.5 System Sequence Diagrams

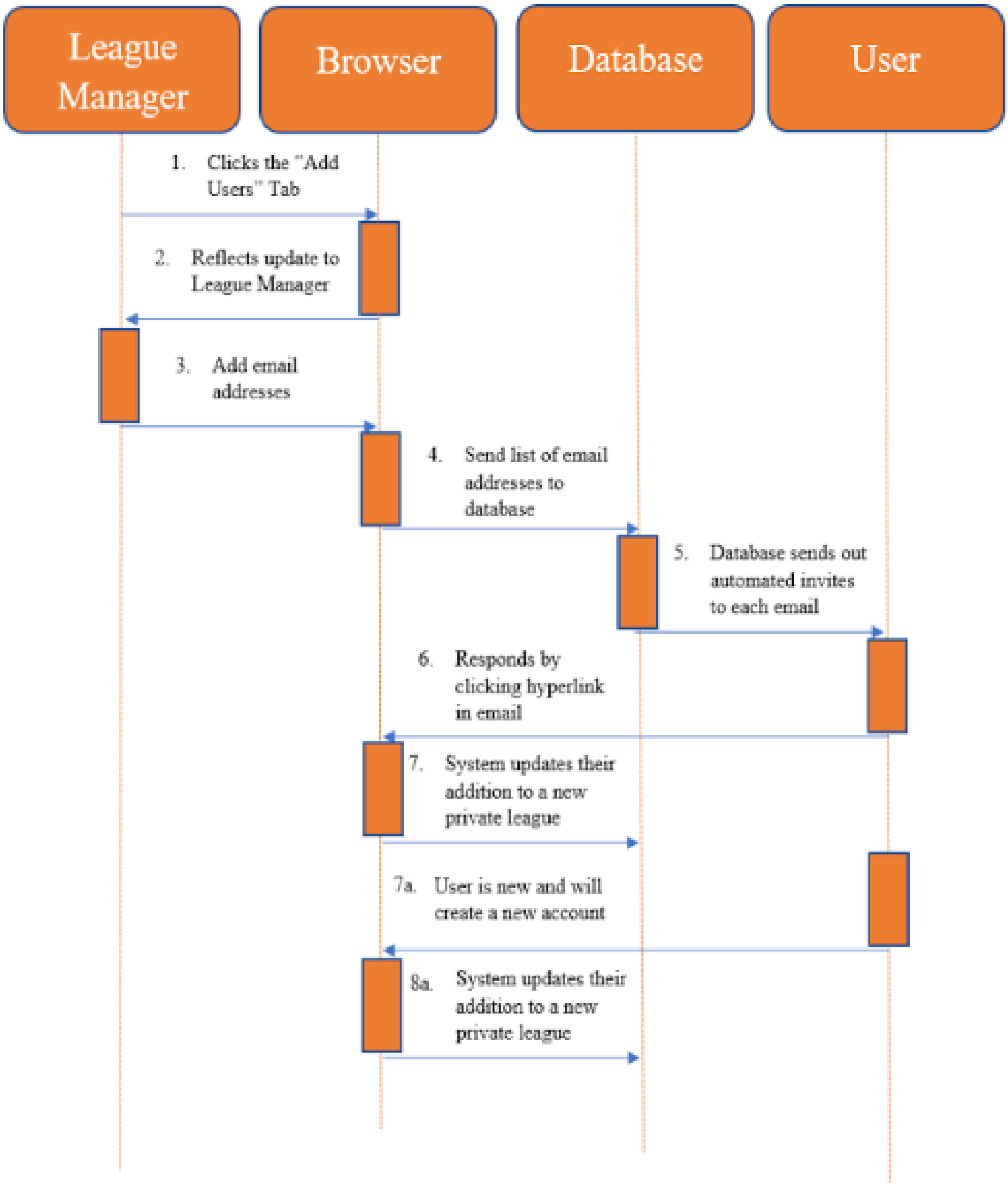
UC1 - Create a League


User
Browser
Database
1. Clicks "Create a League" button.
2. Displays the "new league" form.
3. Enters form information.
4. Sets up new League.
4. Adds new league to dashboard.

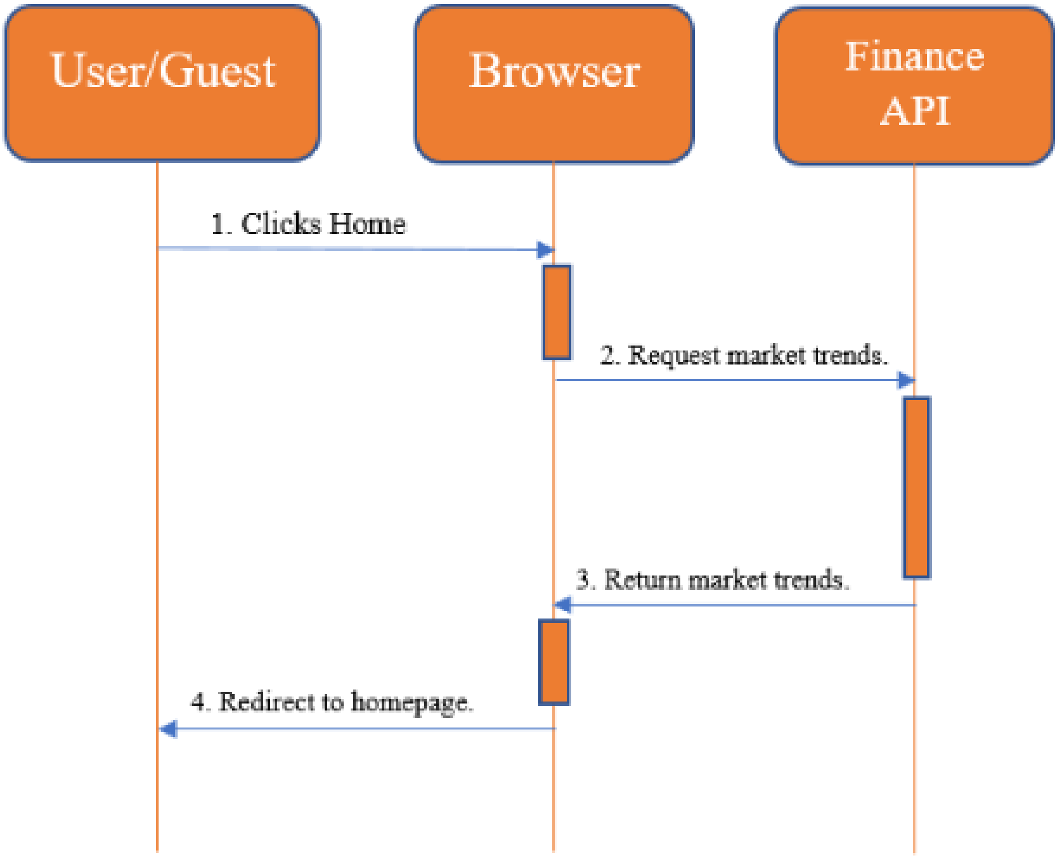
UC4 - Logout


User
Browser
Visitor
1. Clicks the "Logout"
2. System updates to reflect visitor instead of a user
3. User is redirected to the visitor homepage

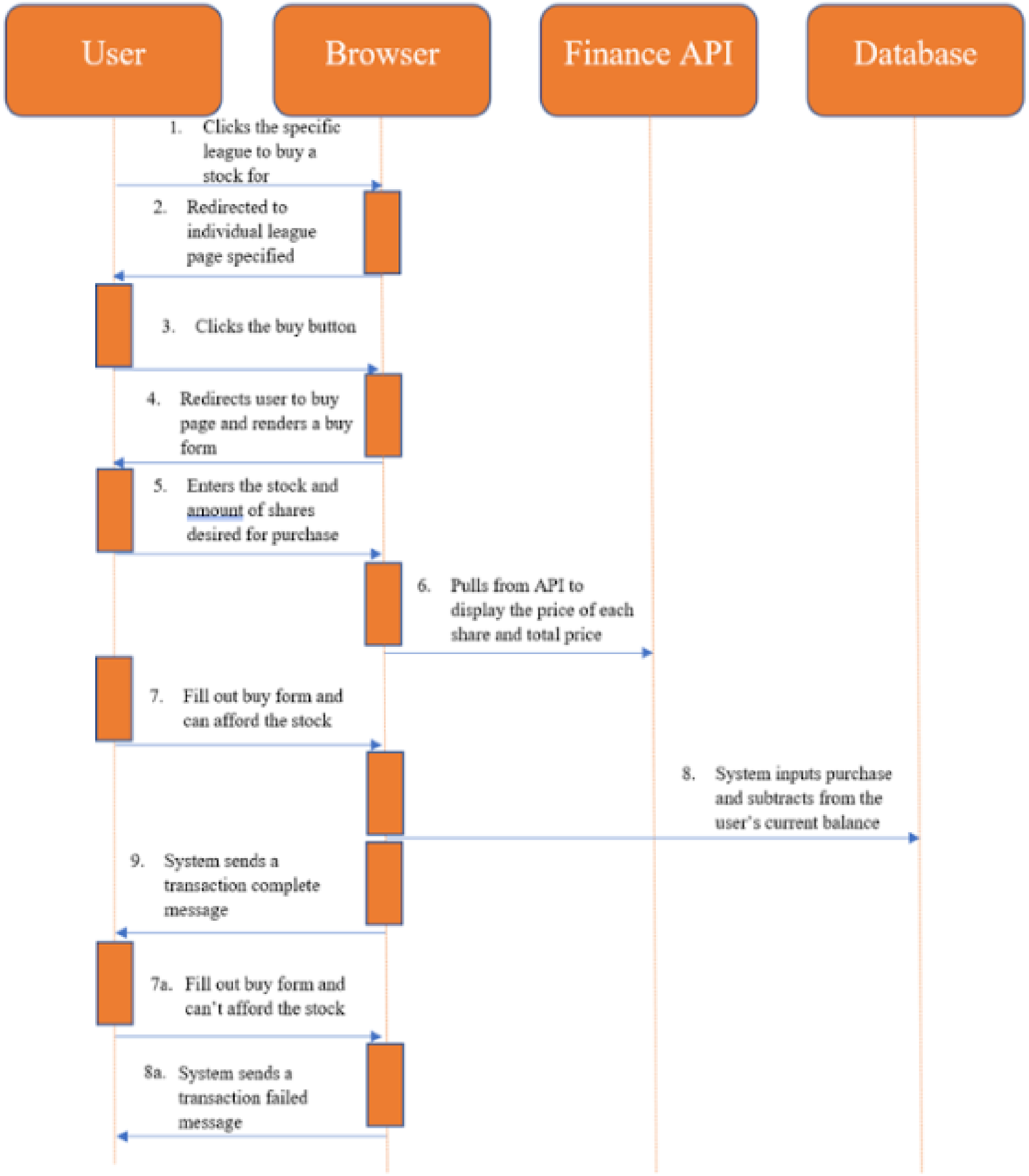
UC5 - Invite User to League


League Manager
Browser
Database
User
1. Clicks the "Add Users" Tab
2. Reflects update to League Manager
3. Add email addresses
4. Send list of email addresses to database
5. Database sends out automated invites to each email
6. Responds by clicking hyperlink in email
7. System updates their addition to a new private league
7a. User is new and will create a new account
8a. System updates their addition to a new private league

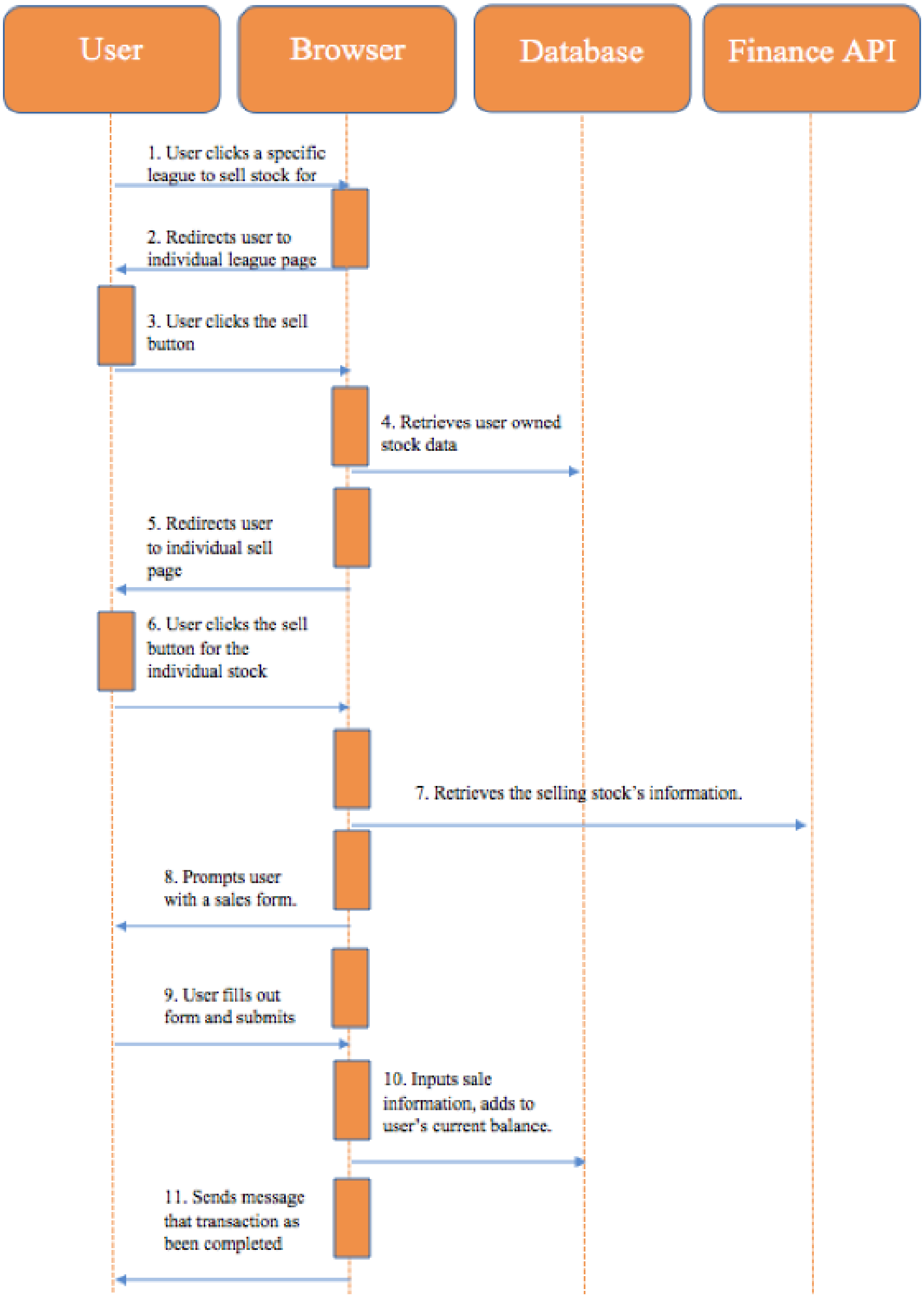
UC6 - View Market Statistics



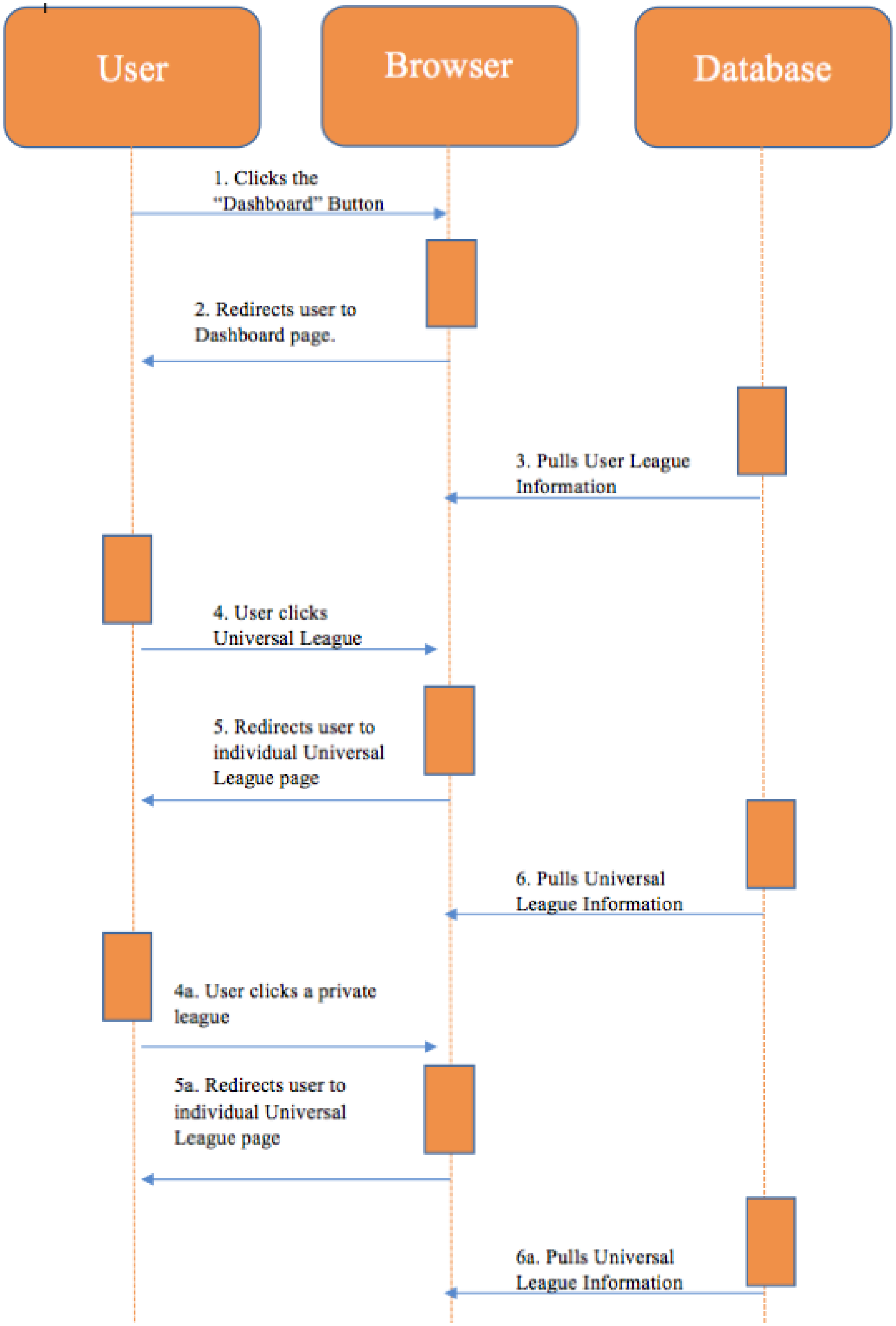
UC7 - Buying a Stock


User
Browser
Finance API
Database
1. Clicks the specific league to buy a stock for
2. Redirected to individual league page specified
3. Clicks the buy button
4. Redirects user to buy page and renders a buy form
5. Enters the stock and amount of shares desired for purchase
6. Pulls from API to display the price of each share and total price
7. Fill out buy form and can afford the stock
8. System inputs purchase and subtracts from the user's current balance
9. System sends a transaction complete message
7a. Fill out buy form and can't afford the stock
8a. System sends a transaction failed message

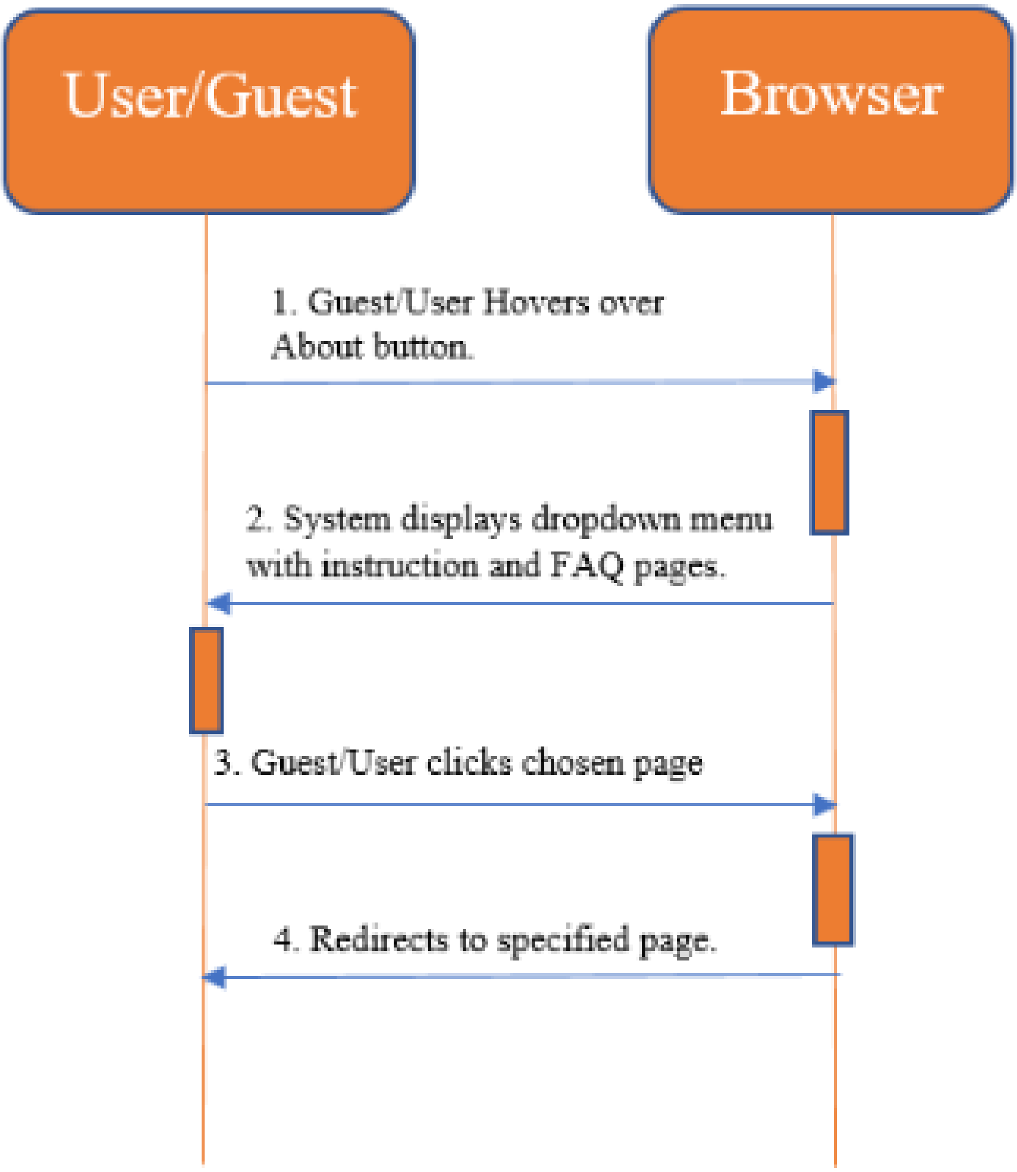
UC8 - Selling a Stock


User
Browser
Database
Finance API
1. User clicks a specific league to sell stock for
2. Redirects user to individual league page
3. User clicks the sell button
4. Retrieves user owned stock data
5. Redirects user to individual sell page
6. User clicks the sell button for the individual stock
7. Retrieves the selling stock's information.
8. Prompts user with a sales form.
9. User fills out form and submits
10. Inputs sale information, adds to user's current balance.
11. Sends message that transaction as been completed

UC9 - View Dashboard and Leaderboards


User
Browser
Database
1. Clicks the "Dashboard" Button
2. Redirects user to Dashboard page.
3. Pulls User League Information
4. User clicks Universal League
5. Redirects user to individual Universal League page
6. Pulls Universal League Information
4a. User clicks a private league
5a. Redirects user to individual Universal League page
6a. Pulls Universal League Information

UC12 - View Instructions/FAQ


User/Guest
Browser
1. Guest/User Hovers over About button.
2. System displays dropdown menu with instruction and FAQ pages.
3. Guest/User clicks chosen page
4. Redirects to specified page.

Chapter 4

User Interface Specification

4.1 Preliminary Design

Titan Trading is an online stock trading fantasy game designed to give users a look into the world of trading on the stock market. Our product is similar to what one would find in a normal fantasy sports application, where users are competing in fantasy leagues alongside their friends in an attempt to come out on top. The trades and portfolios are all fantasy, but the data and numbers are pulled in real time. Each league has its own dashboard where players can view their portfolio, buy/sell stocks, and see how their assets stack up to the rest of the league, and even the world. There are also plenty of resources to get started so that even a novice can jump right in.

4.1.1 Pages for the User

Home Page

The home page is the landing page for the Titan Trading website, and it is accessible both to visitors and registered users. It contains the most up-to-date market trends and daily stock market news, and displays the top 10 players within the Universal League. From this page, users can choose to navigate to their league dashboard, the about page, or their profile page.

Global Navigation Pane

The global navigation pane is displayed on all webpages within the Titan Trading domain. It allows for easy navigation across important pages, such as the homepage, the league dashboard, the about page, or the users profile page. If visitors click on the profile page tab, this will redirect to a page prompting the guest to register for an account.

League Dashboard

The league dashboards primary function is to navigate players to a private or Universal league page. The dashboard lists all private leagues that the user is competing in along with the Universal league, which the user can click on to be directed to that individual league page. The dashboard will also display the users total worth and rank within each league. In addition, the dashboard also has a Create New League button, so users can create a new private league and become a league manager.

If a guest visits the league dashboard, the Universal league will be listed because visitors can be spectators within the Universal league.

About (Dropdown to About Us, FAQ, Instruction, Glossary)

This dropdown menu can be found on the menu bar and gives the user access to the FQ, Instruction guide, and Glossary. All of these pages are there to serve an educational role whether its about Titan Trading, the stock market, or even the words commonly found in the finance world.

About Us

The About Us page is meant to serve as more of a static advertisement of Titan Trading. In this page we lay out our Mission Statement, as well as some history of the company. There will also be a short description of all of the hardworking developers who put their time into making Titan Trading what it is today.

FAQ

A FAQ page is a commonly employed tactic in modern websites. Our FAQ page will encompass questions regarding the website itself, trading in general, and how to setup and design your own league.

Glossary

In the financial world there are many terms that are ambiguous and foreign to a novice. Yet these terms are important for anyone who wants to get involved with the market. With this in mind, the Glossary page will serve as the layman's dictionary. Terms and ideas that are common in the finance world will be collected together and given a simple definition, so that even a novice can navigate the world of finance.

Profile Page

Any user on the site will want to keep track of their stats, achievements and settings. The profile page will serve as the central point that any user can change their personal preferences, view their achievement progress, and see their lifetime stats. It is essentially a running count of an individual user and includes other features such as a profile picture and username that other users will be able to see.

4.1.2 Pages for the Player

Individual League Pages

The individual league page is the home base for each privately owned league. Here, a player can view their competitors current buying power, and decide to either buy or sell stocks. It shows the progress and rank of all players within the league, has options to navigate to a buy or sell page, and displays important league settings. A league manager can choose the day when the competition will end, the starting balance that each player begins with, and if the players will be trading cryptocurrency or normal stocks.

Buy Page

When a user wishes to buy a stock, they have to be shown statistics and information regarding the company as well as direct information about important numbers such as the volume and price of a given stock. The Buy Page is where a user can not only see all of this information, but also execute a buy order on a stock. This page is reached through a given leagues dashboard and will be separated from the dashboards of other leagues. The interface will indicate to the user which league the transaction is occurring in. This prevents the user from mistakenly transaction in one league when it should have occured in another.

Sell Page

To stay competitive in the market, some players may decide to sell some of their assets, and this function will be available on the Sell Page. This page displays a portfolio of the players current assets and has an option to submit limit or market sell orders. This page can only be accessed from an individual league page, as each sell order is specific to one league at a time.

4.1.3 Step by Step Walkthrough with Screen Mockups

The flow of events for use case 1 is given below.

Use Case #1: Creating a new league

Flow of events:

→ User clicks the "Create a League" button

← System calls up the form for creating a new league. This includes the initial settings that are required to be set.

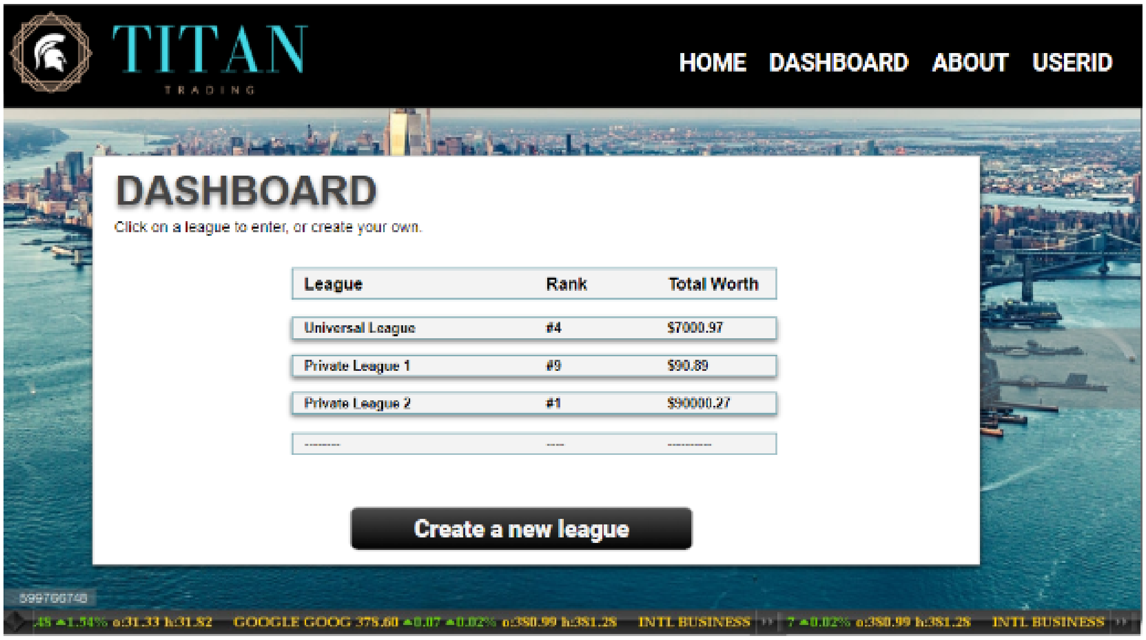
→ User completes the form by putting in their prefered settings.

→ User confirms the form is complete and that all settings are correct.

← System takes form data and updates the database with the new league

← System adds the newly created league to the user's dashboard.

→ User clicks the "Create a League" button



TITAN
TRADING
HOME
DASHBOARD
ABOUT
USERID

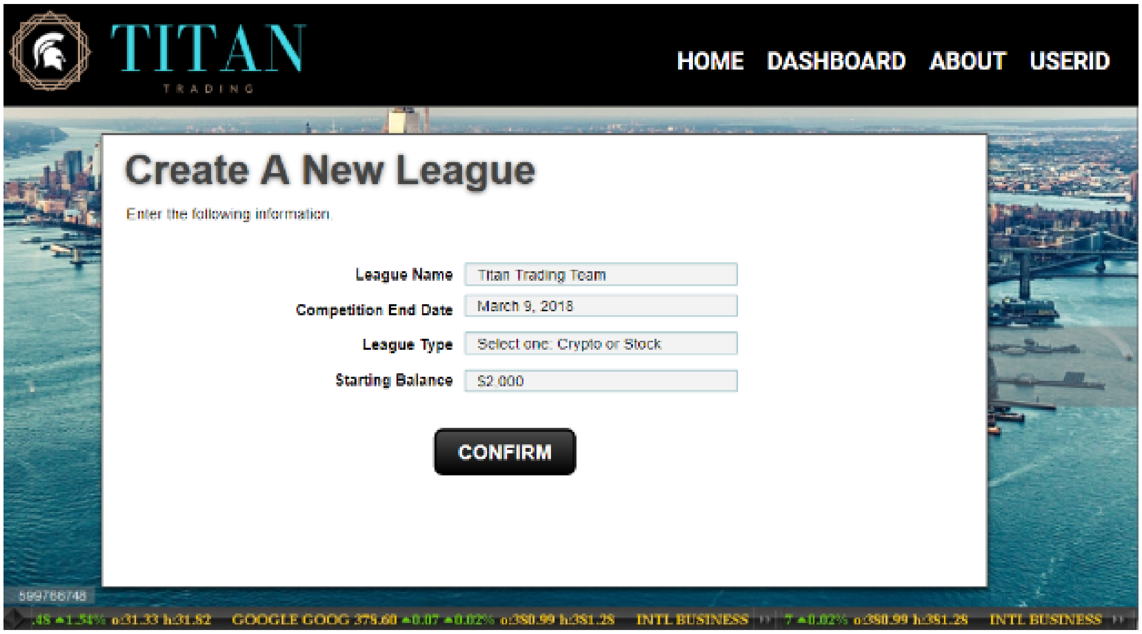
DASHBOARD

Click on a league to enter, or create your own.

League	Rank	Total Worth
Universal League	#4	$7000.97
Private League 1	#9	$90.89
Private League 2	#1	$90000.27
-------	---	--------

Create a new league
599766748
.48 ▲1.54% o:31.33 h:31.82 GOOGLE GOOG 378.60 ▲0.07 ▲0.02% o:380.99 h:381.28 INTL BUSINESS
7 ▲0.02% o:380.99 h:381.28 INTL BUSINESS

← System calls up the form for creating a new league. This includes the initial settings that are required to be set.
→ User completes the form by putting in their prefered settings.
→ User confirms the form is complete and that all settings are correct.

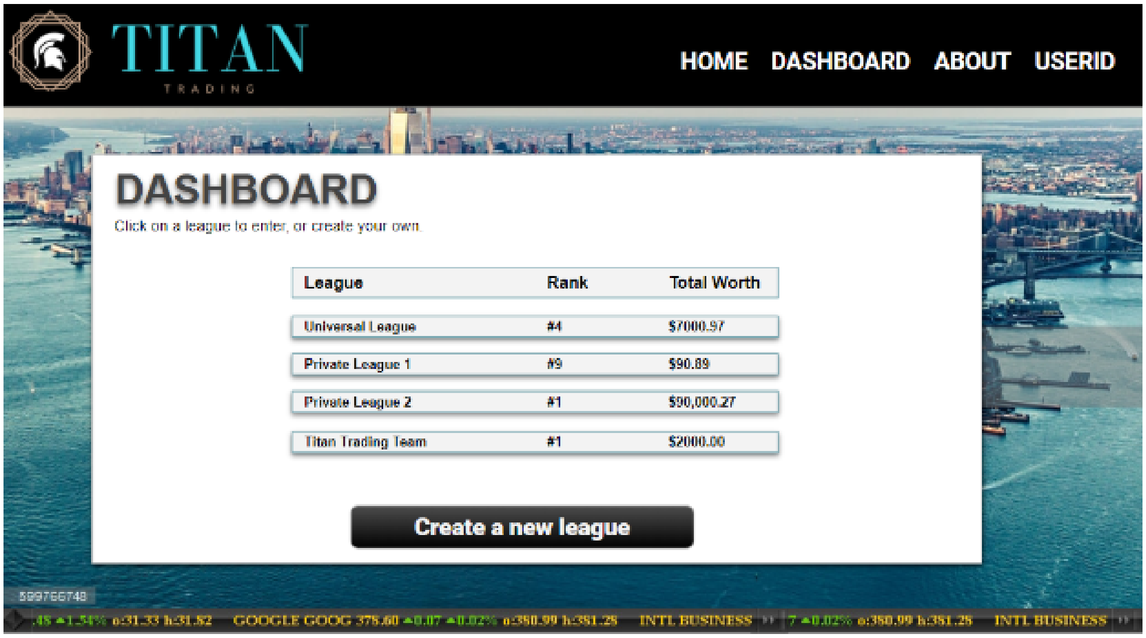

TITAN
TRADING
HOME
DASHBOARD
ABOUT
USERID
CONFIRM
599766748
.48 ▲1.54% o:31.33 h:31.82 GOOGLE GOOG 378.60 ▲0.07 ▲0.02% o:380.99 h:381.28 INTL BUSINESS 7 ▲0.02% o:380.99 h:381.28 INTL BUSINESS

Create A New League

Enter the following information.

League Name	Titan Trading Team
Competition End Date	March 9, 2018
League Type	Select one: Crypto or Stock
Starting Balance	$2,000

← System takes form data and updates the database with the new league
← System adds the newly created league to the user's dashboard.


TITAN
TRADING
HOME
DASHBOARD
ABOUT
USERID
Create a new league
599766748
.48 ▲1.54% o:31.33 h:31.82 GOOGLE GOOG 378.60 ▲0.07 ▲0.02% o:380.99 h:381.28 INTL BUSINESS 7 ▲0.02% o:380.99 h:381.28 INTL BUSINESS

DASHBOARD

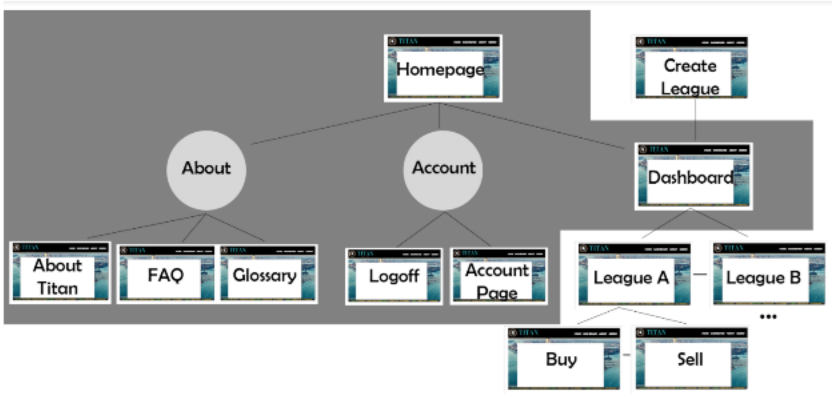
Click on a league to enter, or create your own.

League	Rank	Total Worth
Universal League	#4	$7000.97
Private League 1	#9	$90.89
Private League 2	#1	$90,000.27
Titan Trading Team	#1	$2000.00

4.2 User Effort Estimation

4.2.1 Navigation Tree

Below is a navigation tree, showing how each webpage is connected to the others. Lines between web pages indicate it is accessible from page to page by one click. Web pages within the shaded box are also accessible to each other by one click, due to the global navigation pane.



* Shaded region is one click accessible from the menu bar

Effort Estimation for Use Case 1: Creating a new league:

This will take 3 clicks along with filling out the form for their preferred settings. 2 of the clicks are for user interface navigation, while 1 click would be involved with clerical data entry.

Effort Estimation for Creating a new account:

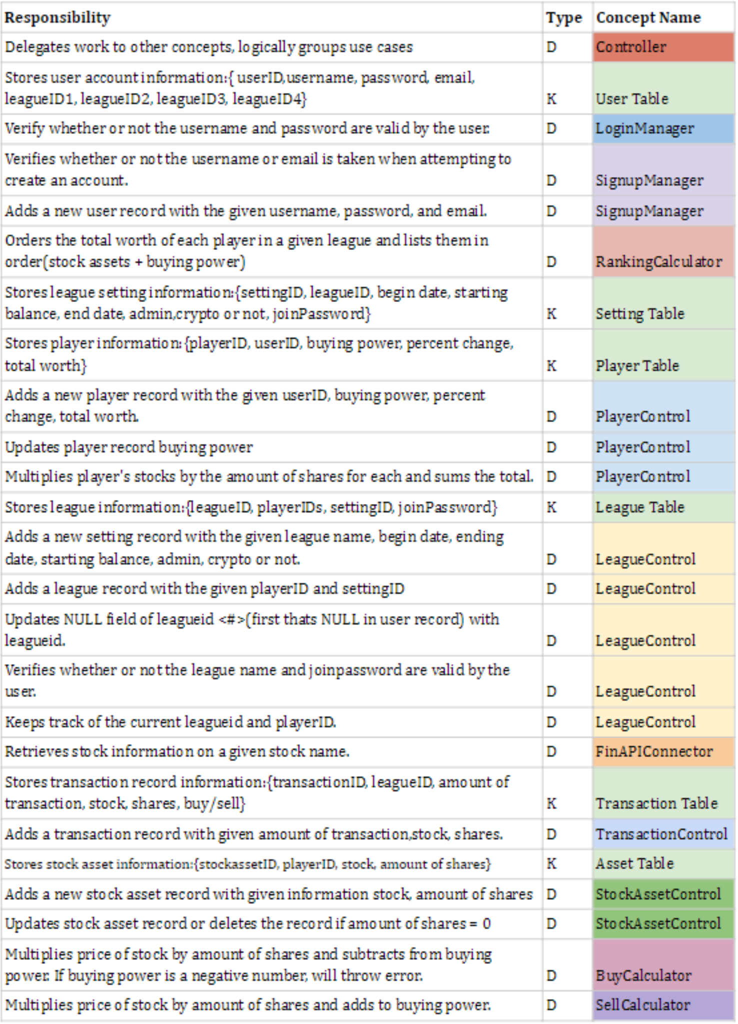
It will take 1 click to get to the account page, another to select create an account. Then the user must enter clerical information for the account details. This is 3 clicks total.

Chapter 5

Domain Analysis

5.1 Domain Model

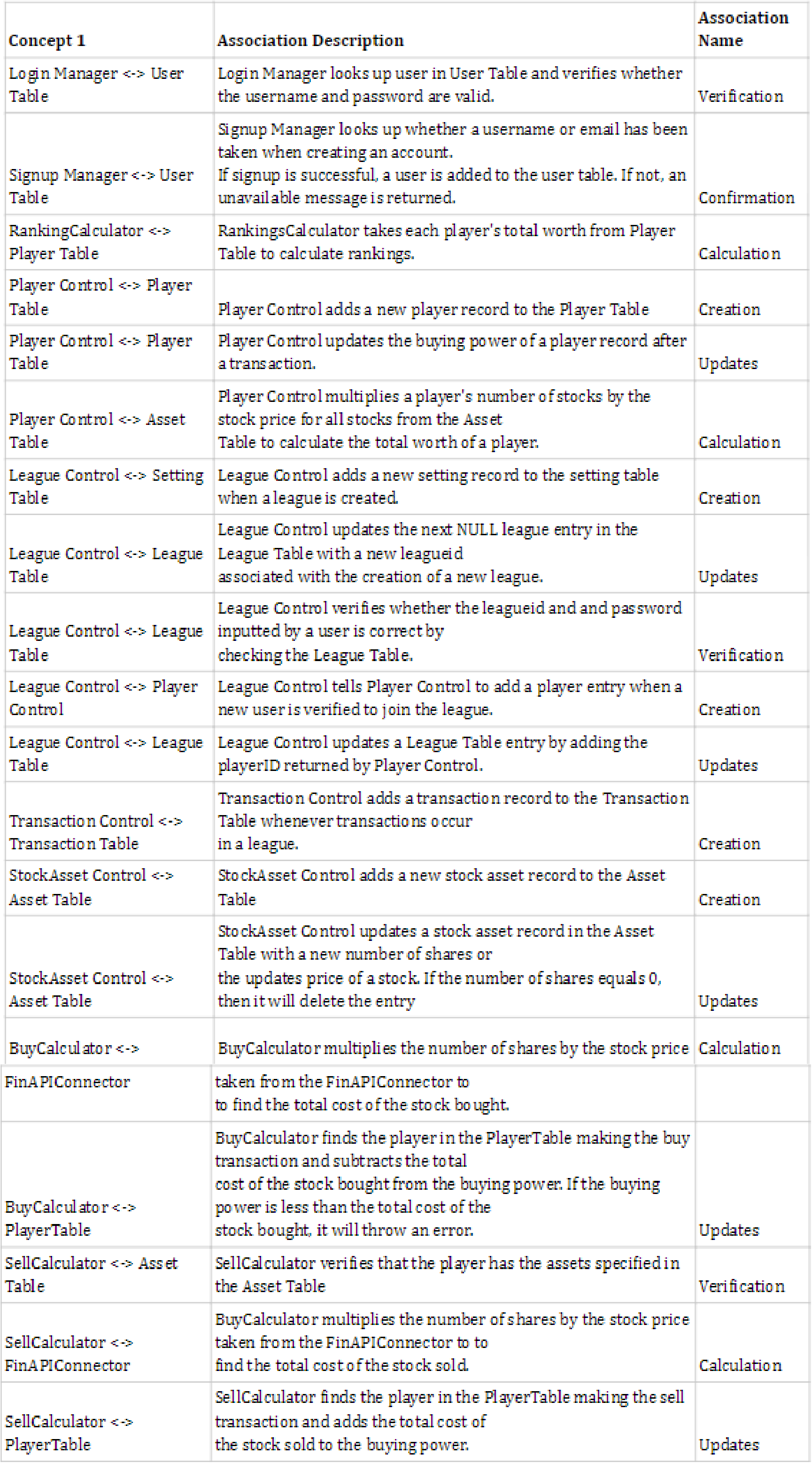
5.1.1 Concept Definition and Responsibilities



Responsibility	Type	Concept Name
Delegates work to other concepts, logically groups use cases	D	Controller
Stores user account information:{ userID,username, password, email, leagueID1, leagueID2, leagueID3, leagueID4}	K	User Table
Verify whether or not the username and password are valid by the user.	D	LoginManager
Verifies whether or not the username or email is taken when attempting to create an account.	D	SignupManager
Adds a new user record with the given username, password, and email.	D	SignupManager
Orders the total worth of each player in a given league and lists them in order(stock assets + buying power)	D	RankingCalculator
Stores league setting information:{settingID, leagueID, begin date, starting balance, end date, admin,crypto or not, joinPassword}	K	Setting Table
Stores player information:{playerID, userID, buying power, percent change, total worth}	K	Player Table
Adds a new player record with the given userID, buying power, percent change, total worth.	D	PlayerControl
Updates player record buying power	D	PlayerControl
Multiplies player's stocks by the amount of shares for each and sums the total.	D	PlayerControl
Stores league information:{leagueID, playerIDs, settingID, joinPassword}	K	League Table
Adds a new setting record with the given league name, begin date, ending date, starting balance, admin, crypto or not.	D	LeagueControl
Adds a league record with the given playerID and settingID	D	LeagueControl
Updates NULL field of leagueid <#>(first thats NULL in user record) with leagueid.	D	LeagueControl
Verifies whether or not the league name and joinpassword are valid by the user.	D	LeagueControl
Keeps track of the current leagueid and playerID.	D	LeagueControl
Retrieves stock information on a given stock name.	D	FinAPIConnector
Stores transaction record information:{transactionID, leagueID, amount of transaction, stock, shares, buy/sell}	K	Transaction Table
Adds a transaction record with given amount of transaction,stock, shares.	D	TransactionControl
Stores stock asset information:{stockassetID, playerID, stock, amount of shares}	K	Asset Table
Adds a new stock asset record with given information stock, amount of shares	D	StockAssetControl
Updates stock asset record or deletes the record if amount of shares = 0	D	StockAssetControl
Multiplies price of stock by amount of shares and subtracts from buying power. If buying power is a negative number, will throw error.	D	BuyCalculator
Multiplies price of stock by amount of shares and adds to buying power.	D	SellCalculator

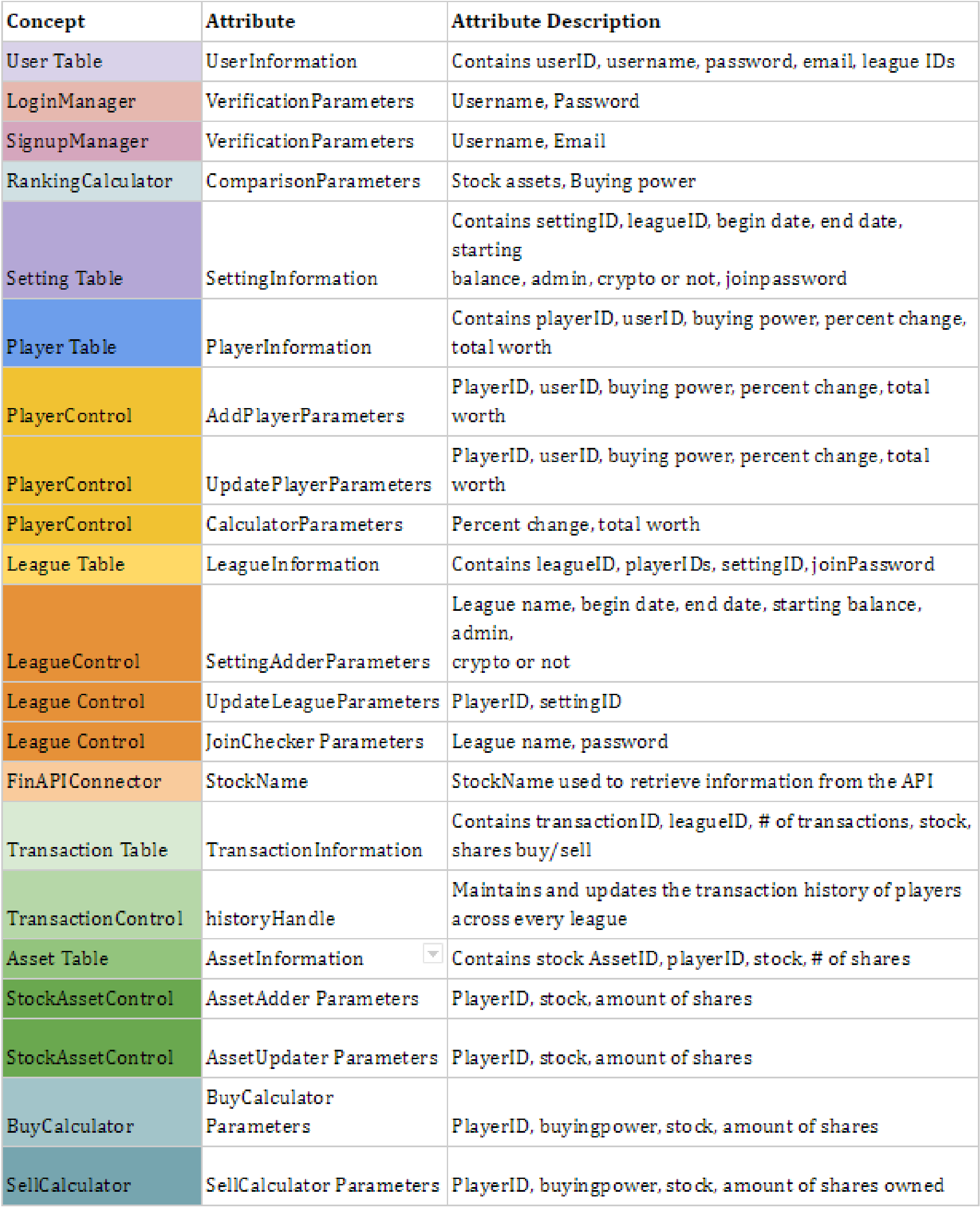
Concept Name	**Pages Used On**
Controller	N/A
User Table	Login/SignUp
LoginManager	Login Page
SignupManager	Sign Up Page
SignupManager	Sign Up Page
RankingCalculator	Home/Dashboard/Individual League Page
Setting Table	CreateLeague/
Player Table	Create League/Join League
PlayerControl	Create League/Join League
PlayerControl	Dashboard/Individual League/Buy/Sell
PlayerControl	Dashboard/Individual League
League Table	Create League/Join League/Buy/Sell
LeagueControl	Create League
LeagueControl	Create League/Join League
LeagueControl	Create League/Join League
LeagueControl	Join League
LeagueControl	Dashboard/Individual League
FinAPIConnector	Buy/Sell
Transaction Table	Buy/Sell
TransactionControl	Buy/Sell
Asset Table	Buy/Sell
StockAssetControl	Buy/Sell
StockAssetControl	Buy/Sell
BuyCalculator	Buy
SellCalculator	Sell

5.2 Concept Associations



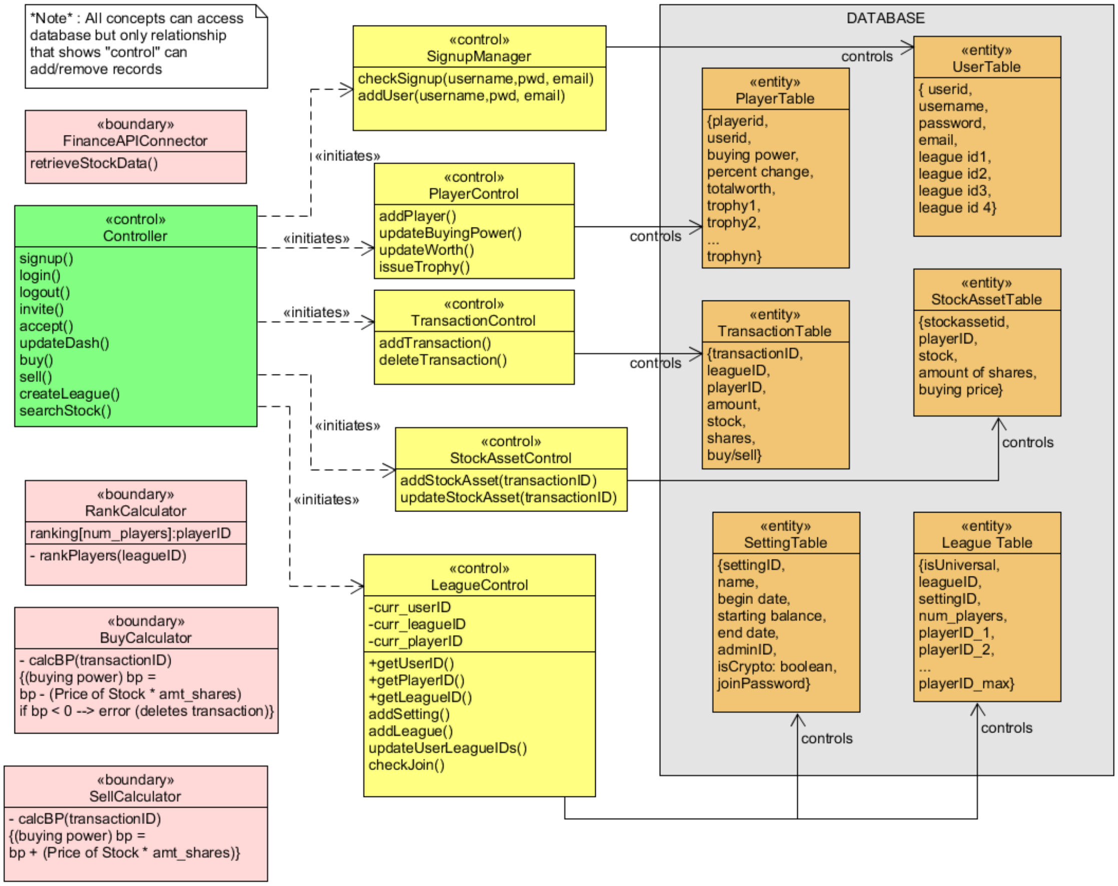
Concept 1	Association Description	Association Name
Login Manager <-> User Table	Login Manager looks up user in User Table and verifies whether the username and password are valid.	Verification
Signup Manager <-> User Table	Signup Manager looks up whether a username or email has been taken when creating an account. If signup is successful, a user is added to the user table. If not, an unavailable message is returned.	Confirmation
RankingCalculator <-> Player Table	RankingsCalculator takes each player's total worth from Player Table to calculate rankings.	Calculation
Player Control <-> Player Table	Player Control adds a new player record to the Player Table	Creation
Player Control <-> Player Table	Player Control updates the buying power of a player record after a transaction.	Updates
Player Control <-> Asset Table	Player Control multiplies a player's number of stocks by the stock price for all stocks from the Asset Table to calculate the total worth of a player.	Calculation
League Control <-> Setting Table	League Control adds a new setting record to the setting table when a league is created.	Creation
League Control <-> League Table	League Control updates the next NULL league entry in the League Table with a new leagueid associated with the creation of a new league.	Updates
League Control <-> League Table	League Control verifies whether the leagueid and and password inputted by a user is correct by checking the League Table.	Verification
League Control <-> Player Control	League Control tells Player Control to add a player entry when a new user is verified to join the league.	Creation
League Control <-> League Table	League Control updates a League Table entry by adding the playerID returned by Player Control.	Updates
Transaction Control <-> Transaction Table	Transaction Control adds a transaction record to the Transaction Table whenever transactions occur in a league.	Creation
StockAsset Control <-> Asset Table	StockAsset Control adds a new stock asset record to the Asset Table	Creation
StockAsset Control <-> Asset Table	StockAsset Control updates a stock asset record in the Asset Table with a new number of shares or the updates price of a stock. If the number of shares equals 0, then it will delete the entry	Updates
BuyCalculator <-> FinAPIConnector	BuyCalculator multiplies the number of shares by the stock price taken from the FinAPIConnector to to find the total cost of the stock bought.	Calculation
BuyCalculator <-> PlayerTable	BuyCalculator finds the player in the PlayerTable making the buy transaction and subtracts the total cost of the stock bought from the buying power. If the buying power is less than the total cost of the stock bought, it will throw an error.	Updates
SellCalculator <-> Asset Table	SellCalculator verifies that the player has the assets specified in the Asset Table	Verification
SellCalculator <-> FinAPIConnector	BuyCalculator multiplies the number of shares by the stock price taken from the FinAPIConnector to to find the total cost of the stock sold.	Calculation
SellCalculator <-> PlayerTable	SellCalculator finds the player in the PlayerTable making the sell transaction and adds the total cost of the stock sold to the buying power.	Updates

5.3 Attribute Definitions

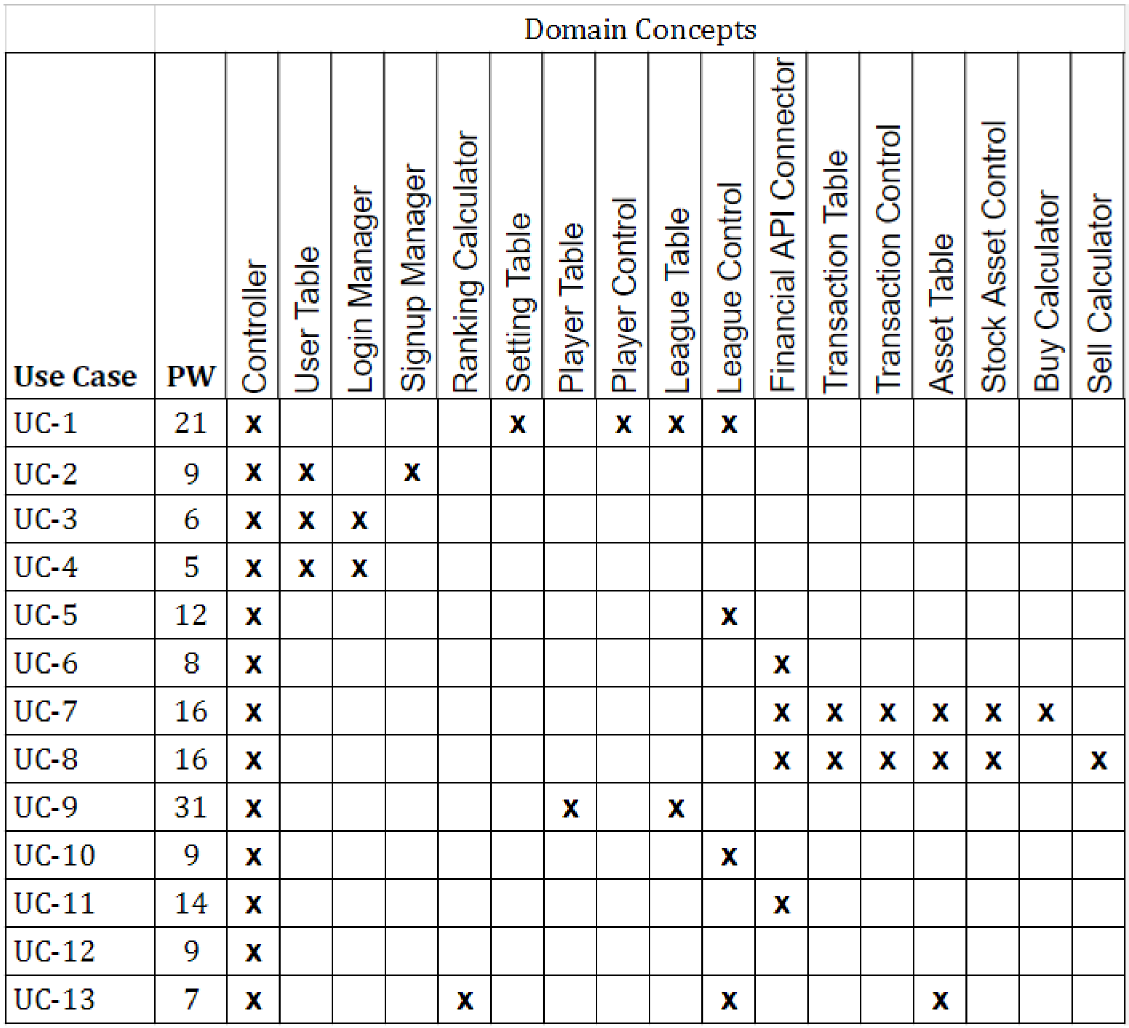


Concept	Attribute	Attribute Description
User Table	UserInformation	Contains userID, username, password, email, league IDs
LoginManager	VerificationParameters	Username, Password
SignupManager	VerificationParameters	Username, Email
RankingCalculator	ComparisonParameters	Stock assets, Buying power
Setting Table	SettingInformation	Contains settingID, leagueID, begin date, end date, starting balance, admin, crypto or not, joinpassword
Player Table	PlayerInformation	Contains playerID, userID, buying power, percent change, total worth
PlayerControl	AddPlayerParameters	PlayerID, userID, buying power, percent change, total worth
PlayerControl	UpdatePlayerParameters	PlayerID, userID, buying power, percent change, total worth
PlayerControl	CalculatorParameters	Percent change, total worth
League Table	LeagueInformation	Contains leagueID, playerIDs, settingID, joinPassword
LeagueControl	SettingAdderParameters	League name, begin date, end date, starting balance, admin, crypto or not
League Control	UpdateLeagueParameters	PlayerID, settingID
League Control	JoinChecker Parameters	League name, password
FinAPIConnector	StockName	StockName used to retrieve information from the API
Transaction Table	TransactionInformation	Contains transactionID, leagueID, # of transactions, stock, shares buy/sell
TransactionControl	historyHandle	Maintains and updates the transaction history of players across every league
Asset Table	AssetInformation	Contains stock AssetID, playerID, stock, # of shares
StockAssetControl	AssetAdder Parameters	PlayerID, stock, amount of shares
StockAssetControl	AssetUpdater Parameters	PlayerID, stock, amount of shares
BuyCalculator	BuyCalculator Parameters	PlayerID, buyingpower, stock, amount of shares
SellCalculator	SellCalculator Parameters	PlayerID, buyingpower, stock, amount of shares owned

5.4 Domain Model



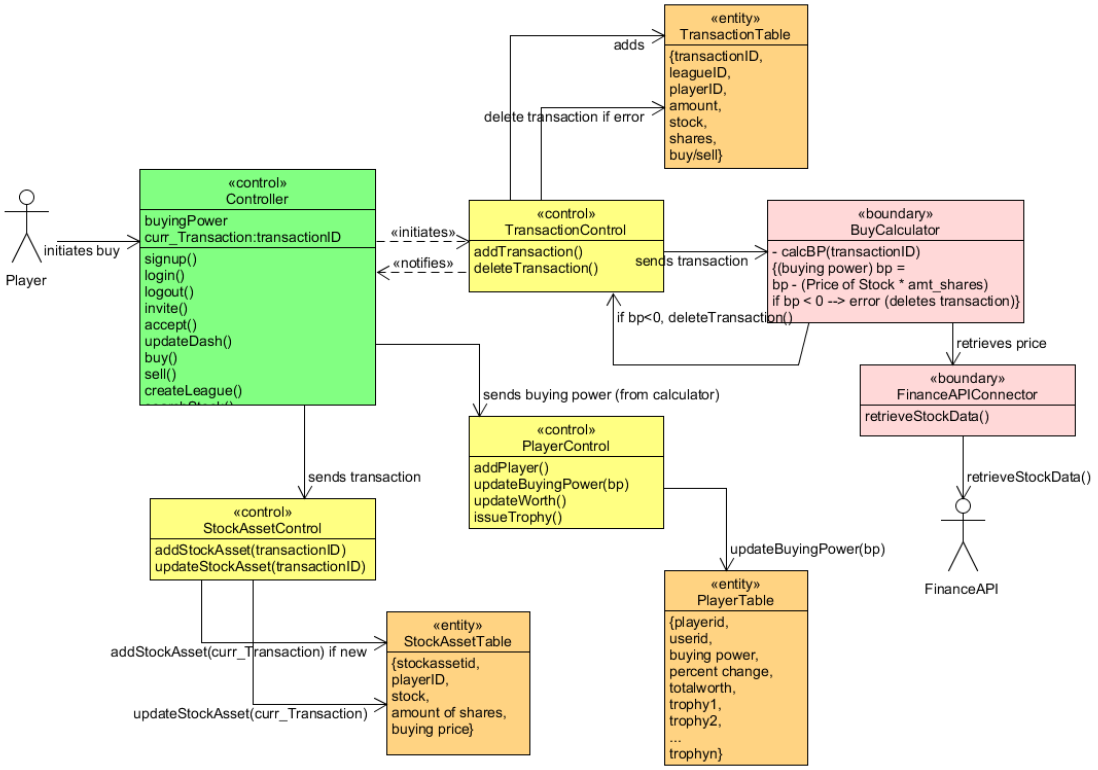
5.5 Traceability Matrix



	Domain Concepts																	
Use Case	**PW**	Controller	User Table	Login Manager	Signup Manager	Ranking Calculator	Setting Table	Player Table	Player Control	League Table	League Control	Financial API Connector	Transaction Table	Transaction Control	Asset Table	Stock Asset Control	Buy Calculator	Sell Calculator
UC-1	21	x					x		x	x	x							
UC-2	9	x	x		x													
UC-3	6	x	x	x														
UC-4	5	x	x	x														
UC-5	12	x									x							
UC-6	8	x										x						
UC-7	16	x										x	x	x	x	x	x	
UC-8	16	x										x	x	x	x	x		x
UC-9	31	x						x		x								
UC-10	9	x									x							
UC-11	14	x										x						
UC-12	9	x																
UC-13	7	x				x					x				x			

5.6 Buy Stock Use Case - UC7

The following is an example of how domain model concepts will interact while executing Use Case 7 - Buying a Stock:



5.7 System Operation Contracts

UC-1 - Create a League

- Preconditions:
  - User is involved in fewer leagues than the league limit
- Postconditions:
  - User is a league manager for the new league. User does not exceed the league limit.

UC-4 - Logout

- Preconditions:
  - User is already logged on to their account on the website
- Postconditions:
  - User successfully logged out and returned to the visitor homepage

UC-5 - Invite User to League

- Preconditions:
  - League Manager invites a user that is not already in their private league

- Postconditions:
  - User has been successfully added to the private league. The dashboard updates to reflect the league addition

UC-6 - View Market Statistics

- Preconditions:
  - User/Guest is on any page of the website
- Postconditions:
  - User/Guest is on the market statistics page (homepage)

UC-7 - Buying a Stock

- Preconditions:
  - User is on the League Dashboard page
- Postconditions:
  - User has bought the desired stock and the Database reflects the purchase

UC-8 - Selling a Stock

- Preconditions:
  - User is on the League Dashboard page
- Postconditions:
  - User has sold the desired stock and the Database reflects the sale

UC-9 - Viewing Dashboard and League Leaderboards

- Preconditions:
  - User is on any page of the website
- Postconditions:
  - User is successfully presented information on current league rankings

UC-10 - Accept/Decline Invitation

- Preconditions:
  - User is viewing his profile page
- Postconditions:
  - User may be entered into the given league depending on the choice made, and the database reflects the league changes made

UC-11 - Search Stock

- Preconditions:
  - Player is viewing the buy page for some league
- Postconditions:
  - None

UC-12 - View Instructions/FAQ

- Preconditions:
  - Player is on any page of the website
- Postconditions:
  - Player is viewing the Instructions/FAQ page

5.8 Economic and Mathematical Models

Cryptocurrency versus Stocks

In Titan Trading, there are two league types to which you can join: 1) a public equities league, or 2) a cryptocurrency league. Within our game, you cannot create a league where you can jointly trade stocks and cryptocurrency. This is due to the fact that stocks and cryptocurrency operate under very different economic models and assumptions, which require different investment strategies.

For the public equities league, we assume that stocks operate under the efficient market hypothesis (EMH). The efficient market hypothesis is an investment theory that states that stocks will always trade at their fair value, so that investors will never be able to purchase underpriced stocks or sell stocks at an inflated price. (Efficient Market Hypothesis - EMH). In our current market, investors are typically rational and are only willing to pay what an equity is actually worth.

In comparison, for the cryptocurrency league, cryptocurrency does not abide by any economic models or government regulations, since its origins only date back to 2009 (Marr, Bernard). Unlike public equity, cryptocurrencies - like Bitcoin - drastically fluctuate in price and are extremely volatile for a variety of reasons. Cryptocurrency is affected by uncertainty in its store of value, which is the function of an assets usefulness with some predictability. In addition, its current market is driven by extreme press about security breaches, rapid positive gains, or illegalization of the currency (Barker, Jonathan Todd) . Because crypto is influenced by its lack of certainty and its inconsistent media perception, cryptocurrency investors trade irrationally, thus contradicting the efficient market hypothesis and causing volatility in the crypto market.

In order to stay consistent with these two economic models, Titan Trading separates its two types of leagues, to accurately portray the separate universes in which these markets exist. This gives Titan Traders the opportunity to change their investment strategies based on if they are trading stocks or cryptocurrencies.

Perfect Competitions

In Titan Trading, we assume that all users are playing in a market with perfect competition. This principle states that no single participant has enough resources/power to control the market. (Fontinelle, Amy) In order to apply perfect competition to our fantasy league, the players will follow the following rules:

- Not one player will control the market or industries
- Every individual will have access to the same information as other investors
- One user can represent multiple players, whose assets are separate from the others
- All investors within a league start with the same amount of money so no one person has more power than everyone else
- Insider trading will not be present because stock information is uniform throughout the website

The financial APIs we will be using are representative of current market data. This, however, leads to an inconsistency in our economic model since Titan Traders will be playing in a universe of perfect competition, while the current stock and current cryptocurrency market do not exhibit this concept. However, if Titan Trading did not follow the rules of perfect competition, each player would have some authority to change the stock data in non-traditional ways, making the fantasy league even more unrealistic from real stock/cryptocurrency trading.

Buy and Sell Orders

All market prices are retrieved from two separate APIs. One API will retrieve public equity market prices, while the second API will retrieve cryptocurrency prices. All information is accessible by every player within Titan Trading.
Once a buy order is submitted, the following mathematical model will be used to calculate the traders new total worth and buying power.

buying power = current buying power - (number of assets bought)market price of asset
total worth = total worth + (number of assets bought)market price of asset
Similarly to buy orders, sell orders will use the following mathematical model.
buying power = current buying power + (number of assets sold)market price of asset
total worth = total worth + (number of assets sold)market price of asset

Total worth is variable to change based on the current market price of an asset, while buying power will stay constant.

Achievements

Once a user has met the requirements for a certain achievement, they user will be given a reward. Most mathematical models for receiving achievements will be defined like the following example.
if (number of participating leagues) == 3, get a reward

Chapter 6

Plan of Work

The plan of work is for milestones and plans for after the submission of report 1.

The plan of work is divided into Development Milestones which serve as checkpoints for development and will determine our course of action as the semester continues. Each of these milestones are broken down into a list of tasks that need to be accomplished to accomplish the milestone.

Every milestone is classified as either Major or Minor depending on the feature or design that is laid out. Major milestones must be completed by the next demo date. Minor milestones are not required but they do merit working on should time allow it.

NOTE: The initial milestones are more detailed than the later milestones. Tese milestones and the plan of work will be adjusted and given more detail as certain developmental milestones are reached and demos are constructed.

6.1 Milestone 1 - Formalised Front End Wireframes for main pages - Major

Due 2/23/18

As part of the second report, wireframes and sketches of the web pages have to be formalised and put together. These are to be done not only for the report our self, but for the use of front end development. To begin we will set the first milestone to be only the most important pages so that implementation and programming on the front end can begin:

- Homepage(Not logged in)
- Homepage(Logged in)
- Create a League form
- League Dashboard

6.2 Milestone 2 - Front End implementation of pages- Major

Due 2/25/18

The report does not call for implementation, but the pages that are wireframed should nonetheless be implemented in some front end capacity. There may be no real data that is drawn, but have a working front end is the first step to using Django and bringing our concepts into a reality. The pages that should be implemented are as follows:

- Homepage(Not logged in)
- Homepage(Logged in)
- Create a League form
- League Dashboard

6.3 Milestone 3 - Formalised Front End Wireframes for all pages - Major

Due 2/25/18

The report does not call for implementation, but the pages that are wireframed should nonetheless be implemented in some front end capacity. There may be no real data that is drawn, but have a working front end is the first step to using Django and bringing our concepts into a reality. The pages that should be implemented are as follows:

- Homepage(Not logged in)
- Homepage(Logged in)
- Create a League form
- League Dashboard

6.4 Final Milestone 4 - Implement a basic achievement system on a per user basis - Minor

Due 4/19/18

An achievement system is not a mission critical feature, but its implementation serves to enhance the user experience and give incentive to continue using the product and exploring its features. This can be done in a similar fashion to Snapchat, where the criteria is hidden and acquisition results in obtaining some award that can be displayed to other users.

6.5 Final Milestone 5 - Add cryptocurrency data with Cryptoleagues - Major

Due 4/23/18

An important addition that was desired was the addition of cryptocurrency to the typical stock trading game. At this point the project already has transaction infrastructure and the ability to well display information. The addition of cryptocurrencies complicates the product more horizontal than vertically. This means that current systems simply have to be expanded to include this new dynamic, rather than producing entire new systems from scratch.

6.6 Final Milestone 6 - Construct dynamic graphs that react to user input - Major

Due 4/29/18

The previously created static graphs can be enhanced to react to mousing over, option setting, and use different graph styles so that traders of all skill levels can get a better sense of the market, without leaving the application to go to third parties.

6.7 Final Milestone 7 - Construct a tutorial page for beginners - Minor

Due 4/29/18

At this point in production the product should be mostly complete in functionality and in ability to fulfill the goals set out. Another goal for the product is to give it an educational use. The creation of a tutorial on top of what is already constructed can give a new UX opportunity and serves as another enhancement to the product.

6.8 Final Milestone 8 - Choreograph a final demo - Major

Due 5/2/18

This demo is the final part of the project, and making sure all is smooth is important when showing off in our final demonstration. Using the lessons that were learned in the first demo, we can showcase all of the implemented features of Titan Trading.

6.9 Project Breakdown

Below are the various areas of development we have decided to group our application into. These groupings are a rough division of labor, and a common infrastructure will be used by all team members to have code that is compatible with the entire application.

6.9.1 User Login and Email Management

Avanish Mishra will develop the infrastructure to allow users to set up accounts and login with their email and personal information. See **ST-1** and **ST-2**.

6.9.2 League Management

League Creation and Invitations

Nicholas Lurski will develop the interface to create leagues, designate them as crypto-leagues or not, and invite other users to a newly created league using email identification. See **ST-6**, **ST-7**, **ST-9**, **ST-10**, and **ST-12**.

League Rules and Settings

Brooks Tawil will develop the systems rules, settings and preferences interface. Preferences are set on a league and user level and can vary widely. See **ST-8**, **ST-15** and **ST-22**.

Player Records

Safa Shaikh will develop the infrastructure to keep track of player records and information pertinent to the league, including buying power and rank. See **ST-5**, **ST-7**, **ST-13**, **ST-20**, **OSR-7**, **OSR-10**, and **OSR-13**.

6.9.3 Transactions

Buy Transactions

Timothy Liu will develop the UI that will allow users to buy stocks of their choice. This will include developing the individual buy page described by on-screen requirements **OSR-10**, **OSR-11**, **OSR-12** as well as all buying functionality listed in story **ST-18**.

Sell Transactions

Kristene Aguinaldo will develop the capability to sell currently owned stocks and assets. This will include developing the individual sell page described by on-screen requirements **OS-13** and **OS-14**, as well as all selling functionality listed in story **ST-18**.

6.9.4 Market Trends and Widgets

Kristian Wu will develop the widgets that display current market trends and relevant articles. This will include graphs for trending stocks and those that the individual search for described in **OSR-5**, **OSR-11**, and **ST-14**.

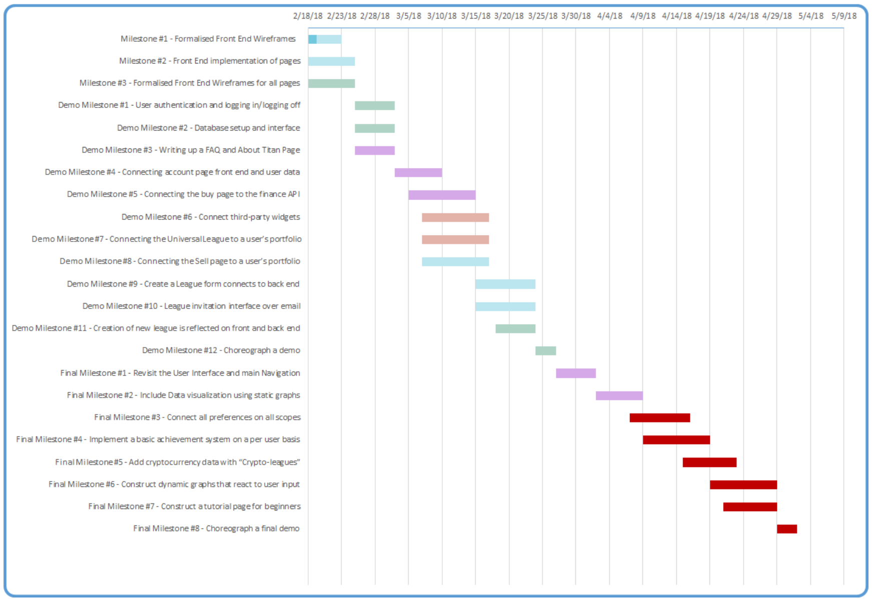
6.9.5 Stock Data and Search Capability

Steven Adler will enable users to search up various stocks by name or ticker symbol. This includes retrieving stock data at the time of buys and sells and keeping a database of which stocks are traded. See **ST-17**, **ST-19**, **ST-23**, and **OSR-11**.

6.9.6 Leaderboards and Ranking

David Zhang will implement the interface to sort and rank users in a leaderboard for each league. Sorting algorithms should conform to league settings. The universal league leaderboard will rank by total asset value. See **ST-16**, **OSR-5**, and **OSR-9**.

6.10 GANTT Chart



Task Name	Start Date	End Date	Duration (Days)	Days Complete	Days Remaining	Percent Complete
Milestone #1 - Formalised Front End Wireframes	2/18/2018	2/23/2018	5	1.25	3.75	25%
Milestone #2 - Front End implementation of pages	2/18/2018	2/25/2018	7	0.00	7.00	0%
Milestone #3 - Formalised Front End Wireframes for all pages	2/18/2018	2/25/2018	7	0.00	7.00	0%
Demo Milestone #1 - User authentication and logging in/logging off	2/25/2018	3/3/2018	6	0.00	6.00	0%
Demo Milestone #2 - Database setup and interface	2/25/2018	3/3/2018	6	0.00	6.00	0%
Demo Milestone #3 - Writing up a FAQ and About Titan Page	2/25/2018	3/3/2018	6	0.00	6.00	0%
Demo Milestone #4 - Connecting account page front end and user data	3/3/2018	3/10/2018	7	0.00	7.00	0%
Demo Milestone #5 - Connecting the buy page to the finance API	3/5/2018	3/15/2018	10	0.00	10.00	0%
Demo Milestone #6 - Connect third-party widgets	3/7/2018	3/17/2018	10	0.00	10.00	0%
Demo Milestone #7 - Connecting the Universal League to a user's portfolio	3/7/2018	3/17/2018	10	0.00	10.00	0%
Demo Milestone #8 - Connecting the Sell page to a user's portfolio	3/7/2018	3/17/2018	10	0.00	10.00	0%
Demo Milestone #9 - Create a League form connects to back end	3/15/2018	3/24/2018	9	0.00	9.00	0%
Demo Milestone #10 - League invitation interface over email	3/15/2018	3/24/2018	9	0.00	9.00	0%
Demo Milestone #11 - Creation of new league is reflected on front and back end	3/18/2018	3/24/2018	6	0.00	6.00	0%
Demo Milestone #12 - Choreograph a demo	3/24/2018	3/27/2018	3	0.00	3.00	0%
Final Milestone #1 - Revisit the User Interface and main Navigation	3/27/2018	4/2/2018	6	0.00	6.00	0%
Final Milestone #2 - Include Data visualization using static graphs	4/2/2018	4/9/2018	7	0.00	7.00	0%
Final Milestone #3 - Connect all preferences on all scopes	4/7/2018	4/16/2018	9	0.00	9.00	
Final Milestone #4 - Implement a basic achievement system on a per user basis	4/9/2018	4/19/2018	10	0.00	10.00	
Final Milestone #5 - Add cryptocurrency data with "Crypto-leagues"	4/15/2018	4/23/2018	8	0.00	8.00	
Final Milestone #6 - Construct dynamic graphs that react to user input	4/19/2018	4/29/2018	10	0.00	10.00	
Final Milestone #7 - Construct a tutorial page for beginners	4/21/2018	4/29/2018	8	0.00	8.00	
Final Milestone #8 - Choreograph a final demo	4/29/2018	5/2/2018	3	0.00	3.00	

Key:	Calculated Cell	Manual Entry Cell
	These cells will be automatically calculated based on the inputs on other cells.	These cells require manual input so the calculated cells have data to work with.

Chapter 7

References

Barker, Jonathan Todd. Why Is Bitcoin's Value So Volatile? Investopedia, 2 Jan. 2018, www.investopedia.com/articles/investing/052014/why-bitcoins-value-so-volatile.asp. [Online, accessed 17 February, 2018]

Fontinelle, Amy. Perfect Competition. Investopedia, 29 Jan. 2018, www.investopedia.com/terms/p/perfectcompetition.asp. [Online, accessed 17 February, 2018]

I. Marsic, Software Engineering. New Brunswick, USA: Ivan Marsic, 2012 Investopedia. Efficient Market Hypothesis - EMH. Investopedia, 28 Sept. 2016, www.investopedia.com/terms/e/efficientmarkethypothesis.asp. [Online, accessed 17 February, 2018]

Marr, Bernard. A Short History Of Bitcoin And Crypto Currency Everyone Should Read. Forbes, Forbes Magazine, 6 Dec. 2017, www.forbes.com/sites/bernardmarr/2017/12/06/a-short-history-of-bitcoin-and-crypto-currency-everyone-should-read/. [Online, accessed 17 February, 2018]

The Paramount Investments League. Rutgers University, 8 May. 2014, http://www.ece.rutgers.edu/ marsic/books/SE/projects/TradingLeague/2014-g1-report3.pdf. [Online, accessed 17 February, 2018]

Rutgers University Investing. Rutgers University, 12 May. 2014, http://www.ece.rutgers.edu/ marsic/books/SE/projects/TradingLeague/2014-g2-report3.pdf. [Online, accessed 17 February, 2018]

User & Design Specifications for Capital Games. Rutgers University, 3 May. 2013, http://www.ece.rutgers.edu/ marsic/books/SE/projects/TradingLeague/2013-g2-report3.pdf. [Online, accessed 17 February, 2018]