

Section 1 - Project Description

1.1 Project

FantasyStock

1.2 Description

FantasyStock is made to have fun and easy to use, risk-free experience for our game'ified version of the stock market.

1.3 Revision History

Date	Comment	Author

Software Design Document

Contents

[Section 1 - Project Description](#)

[1.1 Project](#)

[1.2 Description](#)

[1.3 Revision History](#)

[Section 2 - Overview](#)

[2.1 Purpose](#)

[2.2 Scope](#)

[2.3 Requirements](#)

[2.3.1 Estimates](#)

[2.3.2 Traceability Matrix](#)

[Section 3 - System Architecture](#)

[Section 4 - Data Dictionary](#)

[Section 5 - Software Domain Design](#)

[5.1 Software Application Domain Chart](#)

[5.2 Software Application Domain](#)

[5.2.1 Domain X](#)

[5.2.1.1 Component Y of Domain X](#)

[5.2.1.1.1 Task Z of Component Y1 of Domain X](#)

[Section 6 – Data Design](#)

[6.1 Persistent/Static Data](#)

[6.1.1 Dataset](#)

[6.1.2 Static Data](#)

[6.1.3 Persisted data](#)

[6.2 Transient/Dynamic Data](#)

[6.3 External Interface Data](#)

[6.4 Transformation of Data](#)

[Section 7 - User Interface Design](#)

[7.1 User Interface Design Overview](#)

[7.2 User Interface Navigation Flow](#)

[7.3 Use Cases / User Function Description](#)

[Section 8 - Other Interfaces](#)

[8.1 Interface X](#)

[Section 9 - Extra Design Features / Outstanding Issues](#)

[Section 10 – References](#)

[Section 11 – Glossary](#)

Software Design Document

Section 2 - Overview

2.1 Purpose

This software design document describes the architecture and system design of FantasyStock. This is intended for use for anyone working on the project that needs specific details about a specific aspect.

2.2 Scope

The scope of this module is to provide specific information about any aspect of the project.

2.3 Requirements

User Registration, User Login, User Logout, Create League, View League, Join League, View Registered Leagues, Search Active Leagues, Post Comment on League, Edit Comment on League, Delete Comment on League, Reply to Comment on League, View Profile, Edit Profile, View Other People's Profile, Manage Friends, Send Friend Request, Accept Friend Request, Decline Friend Request, Add Friend, Delete Friend, View Store, Purchase From Store, Toggle Dark Mode.

2.3.1 Estimates

#	Description	Hrs. Est.
1	User Registration	5.00
2	User Login	12.50
3	User Logout	1.25
4	Create League	2.50
5	View League	2.50
6	Join League	2.50
7	View Registered Leagues	2.50
8	Search Active Leagues	2.50
9	Post Comment on League	2.50
10	Edit Comment on League	2.50
11	Delete Comment on League	2.50
12	Reply to Comment on League	2.50
13	View Profile	2.50
14	Edit Profile	2.50
15	View Other People's Profile	2.50
16	Manage Friends	2.50
17	Send Friend Request	2.50
18	Accept Friend Request	2.50
19	Decline Friend Request	2.50
20	Add Friend	2.50
21	Delete Friend	2.50
22	View Store	2.50
23	Purchase From Store	2.50
24	Toggle Dark Mode	1.25
	TOTAL:	70.00

Software Design Document

2.3.2 Traceability Matrix

SRS Requirement	SDD Module
User Registration	5.2.1.4.1 Request of Registration of User
User Login	5.2.1.3.1 Request of Authentication of User
User Logout	5.2.1.3.2 Request of Logout of User
Create League	5.2.2.1 Creation of League
View League	5.2.2.3 Information of League
Join League	5.2.2.2 Joining of League
View Registered Leagues	5.2.2.3.1 Information of Registered Leagues
Search Active Leagues	5.2.2.3.2 Search of Active Leagues
Post Comment on League	5.2.2.4.1 Creation of Comment of League
Edit Comment on League	5.2.2.4.2 Edit of Comment of League
Delete Comment on League	5.2.2.4.4 Deletion of Comment of League
Reply to Comment on League	5.2.2.4.5 Reply of Comment of League
View Profile	5.2.1.2 Information of Users
Edit Profile	5.2.1.2.1 Edit of Information of User
View Other People's Profile	5.2.1.2 Information of Users
Manage Friends	5.2.1.1 Friend Request of Users
Send Friend Request	5.2.1.1.3 Send Friend Request of User
Accept Friend Request	5.2.1.1.1 Accept Friend Request of Friend Request of User
Decline Friend Request	5.2.1.1.2 Decline Friend Request of Friend Request of User
Add Friend	5.2.1.2.2 Add Friend
Delete Friend	5.2.1.2.3 Delete Friend
View Store	5.2.5.1 View of Store
Purchase From Store	5.2.5.2 Purchase from Store
Toggle Dark Mode	5.2.6.1 Toggle Dark Mode

Section 3 - System Architecture

Users are the only users; Users interact with the user's visual interface to the backend which interfaces with our database.

Software Design Document

Section 4 - Data Dictionary

Table

User		
Field	Notes	Type
ID	Unique Identifier	OBJECT ID
Google ID	Token from Google	STRING
Username	Created by user for public display	STRING
Display Name	Display name provided by Google	STRING
Family Name	Family name provided by Google	STRING
Given Name	Given name provided by Google	STRING
Photo	Photo provided by Google	STRING
Bio	Created by user	STRING
Friends	List of friends	[OBJECT ID (USER)]
Friend Requests	List of friend requests	[OBJECT ID (USER)]
Currency	Currency rewarded and used	NUMBER
Active Leagues	List of leagues not yet passed	[OBJECT ID (LEAGUE)]
Passed Leagues	List of leagues passed	[OBJECT ID (LEAGUE)]
Icons	List of icons purchased	[STRING]
Active Icon	Icon currently in use	STRING

League		
Field	Notes	Type
ID	Unique Identifier	OBJECT ID
Host	Person who made the league	OBJECT ID (USER)
Title	Created by user to give a league a name	STRING
Players	List of players in a league	[PLAYER]
Visibility	Input from user to see if league is listed in search results	STRING (ENUM: "PUBLIC", "PRIVATE")
Start Date	The day the league begins	DATE
End Date	The day the league ends	DATE
Active	Sees if league is still active	BOOLEAN
Comment Section	List of user comments	[USER_COMMENT]

Software Design Document

Player		
Field	Notes	Type
ID	Unique Identifier	OBJECT ID
User	Person who is in the league	OBJECT ID (USER)
Stock Picks	List of stocks chosen by person	[STOCK PICK]

Stock_Pick		
Field	Notes	Type
ID	Unique Identifier	OBJECT ID
Ticker	Ticker of the desired	STRING
Quantity	Quantity of the desired stock	NUMBER
Position	Indicator of what user wants to do with stock	STRING (ENUM: "LONG", "SHORT")
Price at Time	Price at the time the user made the stock	NUMBER

Stock		
Field	Notes	Type
ID	Unique Identifier	OBJECT ID
Ticker	Ticker of stock	STRING
Name	Full name of stock	STRING
Sector	Sector which the stock belongs to	STRING
Price	Price of the stock that the last time it was updated	NUMBER
Created At	Time the stock was first recorded by us	DATE
Updated At	Time the stock was last updated by us	DATE

News		
Field	Notes	Type
ID	Unique Identifier	OBJECT ID
Source	Source of article	SOURCE
Author	Author of article	STRING
Title	Title of article	STRING
Description	Description of article	STRING
URL	URL to article	STRING
URL to Image	URL to image used in article	STRING
Published At	Timestamp of when the article was published	DATE
Content	Content of article	STRING
Created At	Timestamp of when object was created	DATE
Updated At	Timestamp of when object was last updated	DATE

Software Design Document

Source		
Field	Notes	Type
ID	Unique Identifier	OBJECT ID
Source ID	Information source	STRING
Name	Full name of information source	STRING

Section 5 - Software Domain Design

5.1 Software Application Domain Chart

User contains other users, and leagues. League contains users, players, and user comments. Player contains users and stock picks. Stock picks retrieve data from stock. Stock contains data from API. Store contains nothing. News contains data from API.

5.2 Software Application Domain

5.2.1 Users

Users contain friends in the form of other users and game information in the form of leagues.

5.2.1.1 Friend Request of Users

Friends are added after the friend or user sends a friend request to the other.

5.2.1.1.1 Accept Friend Request of Friend Request of User

After a friend request has been sent and after it is accepted they are added to each other's friends list.

5.2.1.1.2 Decline Friend Request of Friend Request of User

After a friend request has been sent and after it is declined they are removed from each other's friends request list.

5.2.1.1.3 Send Friend Request of User

After a friend request has been sent, it will be added to the other user's friend request list.

5.2.1.2 Information of Users

Users can edit their data after a request has been sent.

5.2.1.2.1 Edit of Information of User

After a request has been sent then it will update the user information.

5.2.1.2.2 Add Friend

Sends a friend request to the user selected.

5.2.1.2.3 Delete Friend

Deletes a friend off of your friends list.

5.2.1.3 Authentication of Users

Users can authenticate themselves after a request has been sent.

5.2.1.3.1 Request of Authentication of User

After a request has been sent then it will authenticate the user.

5.2.1.3.2 Request of Logout of User

After a request has been sent then it will logout the user.

5.2.1.4 Registration of Users

Users can register after a request has been sent.

5.2.1.4.1 Request of Registration of User

After a request with their google account has been sent then it will register the user.

5.2.2 League

Leagues contain members, comments, and stock information.

5.2.2.1 Creation of League

Leagues are created after a user sends a create league request.

5.2.2.1.2 Pick Stocks of Creation of League

Users pick stocks before creating a league.

5.2.2.2 Joining of League

Users can join an active league after a request has been sent to join with their stock information.

5.2.2.2.1 Pick Stocks of Joining of League

Users pick stocks before joining the league.

5.2.2.3 Information of League

Users can view a league after requesting it.

5.2.2.3.1 Information of Registered Leagues

Users may view the leagues they are registered in.

5.2.2.3.2 Search of Active Leagues

Users may search for and view leagues that are active.

5.2.2.4 Comments of League

Users can leave a comment on a league after making a request with a comment.

5.2.2.4.1 Creation of Comment of League

Users must create a comment before making a request to create a comment.

5.2.2.4.2 Edit of Comment of League

Users must own a comment before making a request to edit their comment.

5.2.2.4.3 View of Comment of League

Users can view a comment upon requesting a comment.

5.2.2.4.4 Deletion of Comment of League

Users must own a comment before making a request to delete their comment.

5.2.2.4.5 Reply of Comment of League

Users must be able to view a comment before making a request to reply to a comment.

5.2.3 Stocks

Stocks contain ticker, name, sector, and price.

5.2.3.1 Creation of Stocks

Stocks create themselves automatically and fill out the ticker, name, sector and current price.

5.2.3.2 Update of Stocks

Stocks update themselves automatically and mutate the price to the current price.

5.2.4 News

New articles contain a source, author, title, description, url, image, published time, and article content.

5.2.4.1 Creation of News

News articles create themselves automatically and fill out source, author, title, description, url, image, published time, and article content.

5.2.4.2 Deletion of News

News articles delete themselves automatically.

5.2.5 Store

Store contains icons or badges.

5.2.5.1 View of Store

Users can view the store upon requesting.

5.2.5.2 Purchase from Store

Users can buy badges from the store provided there are unbought badges and they have sufficient funds.

5.2.6 Settings

New articles contain a source, author, title, description, url, image, published time, and article content.

5.2.6.1 Toggle Dark Mode

News articles create themselves automatically and fill out source, author, title, description, url, image, published time, and article content.

Section 6 – Data Design

6.1 Persistent/Static Data

Describe/illustrate the logical data model or entity relationship diagrams for the persistent data (or static data if static)

6.1.1 Dataset

Users are their own objects and relate to leagues in that they can join a league and have what leagues they are currently playing in to be recorded on their account as well. Additionally, we have stock and news data that is recorded and is never modified with respect to other data, however leagues do use the data from stocks to get the up to date information on the stocks it uses to calculate who's winning. News on the other hand is displayed directly to the user and doesn't interface with other data in any way.

6.1.2 Static Data

The store is based on static data. This is because items in the store don't really change, and they work in conjunction with the user's account which keeps record of what they have bought from the store.

6.1.3 Persisted data

FantasyStock does not contain any infrequently used data.

6.2 Transient/Dynamic Data

FantasyStock does not contain any high-frequency transient or dynamic data.

6.3 External Interface Data

Stock API: AlphaVantage

News API: News API

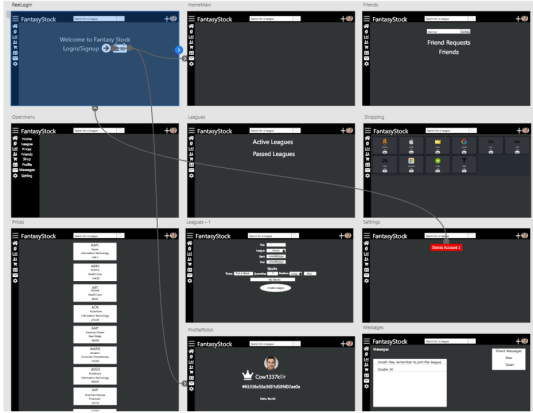
Authentication API: Google OAuth2

6.4 Transformation of Data

FantasyStock does not transform any data from one format to another.

Section 7 - User Interface Design

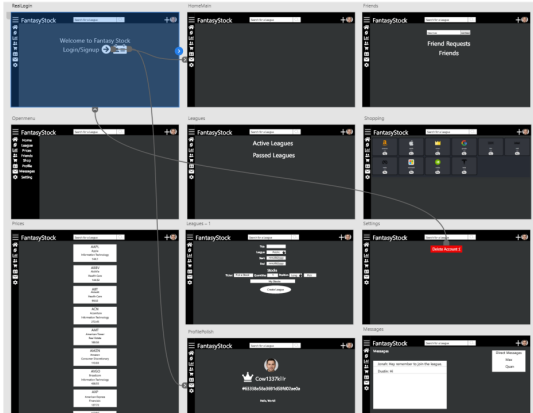
7.1 User Interface Design Overview



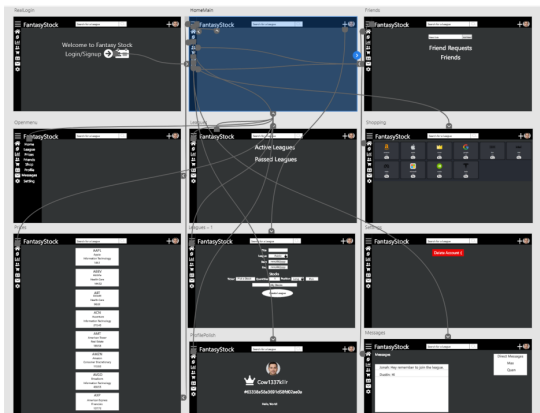
The requirements would be to make multiple pages that contain a login screen, home page, friends list page, leagues list page, store page, a stocks used page, user info page, league creation form page, a delete page, a league view page, and have a navigation system to navigate the application.

7.2 User Interface Navigation Flow

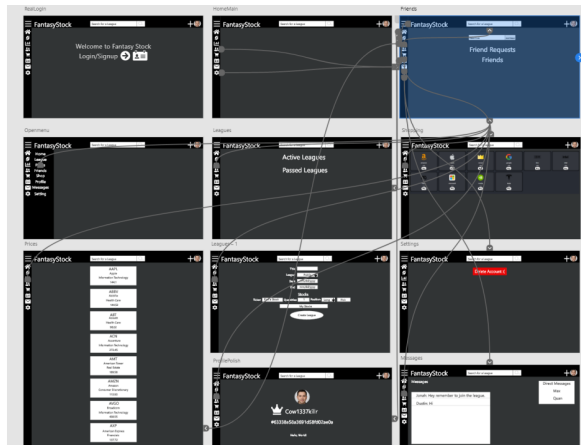
This picture shows the flow from the login page, which allows the user to gain access to the rest of the site.



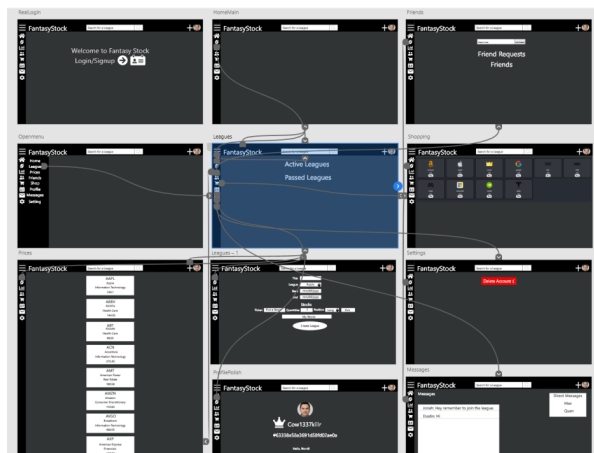
This picture shows the flow once the user is logged in and can access the navigation system.



This picture shows the user flow once the user goes to the friend management page, which allows the user to access the rest of the site and allows the user to access their friend's user profile.

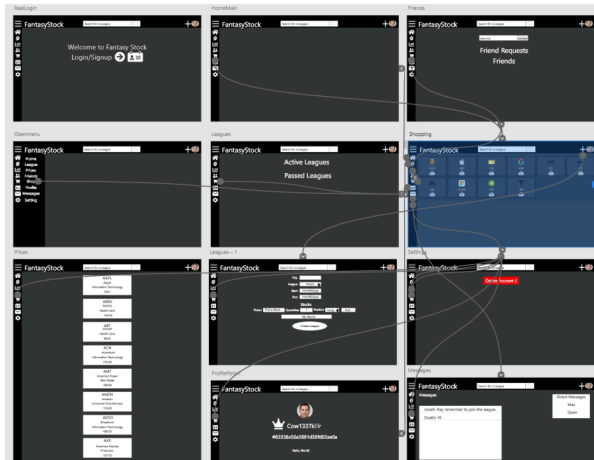


This picture shows the user flow once the user goes to the league management page, which allows the user to access the rest of the site and allows the user to access their active and passed leagues.

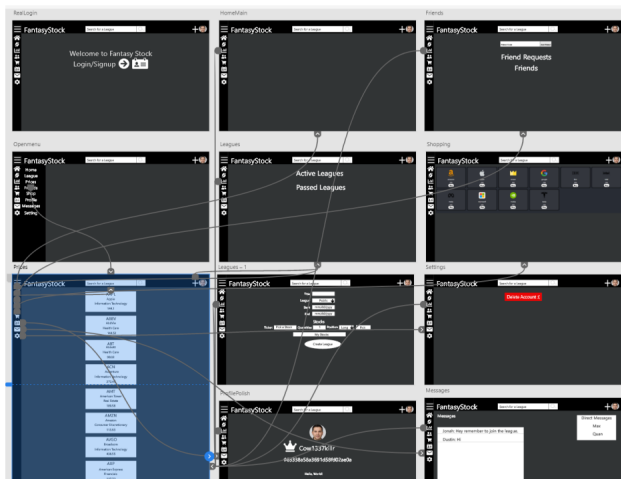


Software Design Document

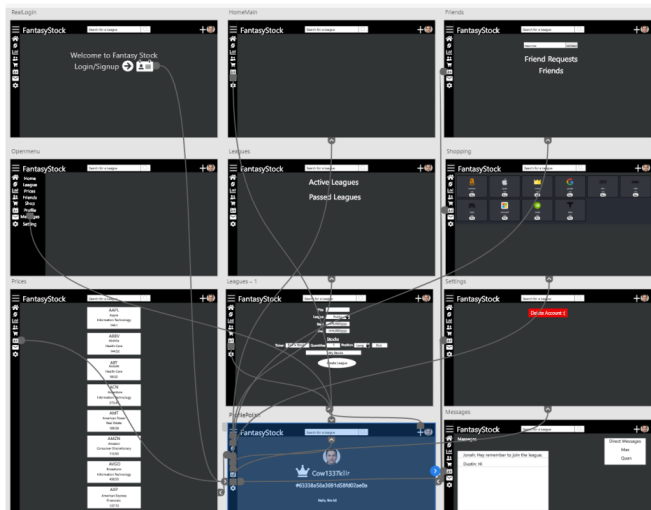
This picture shows the user flow once the user goes to the store page, which allows the user to access the rest of the site and allows the user to access the store where they can purchase badges.



This picture shows the user flow once the user goes to the stock prices page, which allows the user to access the rest of the site.



This picture shows the user flow once the user goes to the profile page, which allows the user to access the rest of the site, and view and edit their profile.



7.3 Use Cases / User Function Description

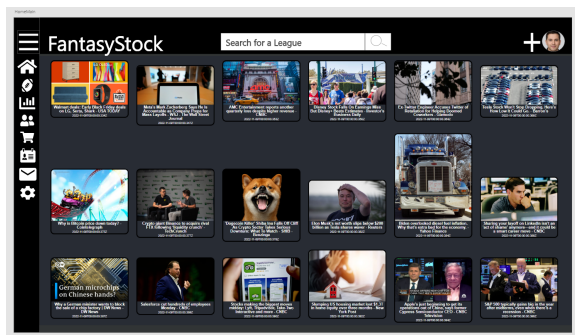
Login

This use case or functionality is the main page that redirects the user into logging into fantasy stock. Every user would be put upon this page when first entering FantasyStock.



Home Interface

The use case or functionality of the home page is so that after users are able to log in, they are able to view the news and see what is with the current stocks.

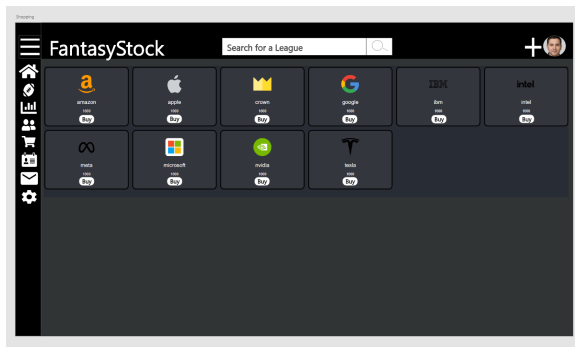


Software Design Document

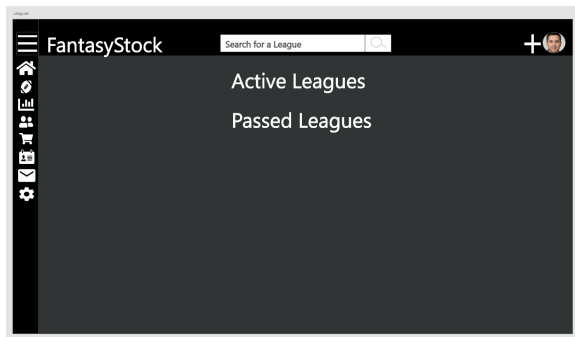
The friend user case allows users to add each other as friends through their own unique IDs. After adding, users are able to accept or decline other users' requests.



The use case for price allows users to buy more stocks or see the current prices of each stock.



The league use case allows users to access or see the leagues that have passed or are ongoing. This occurs after the user has joined a league or the league that the user played is over.

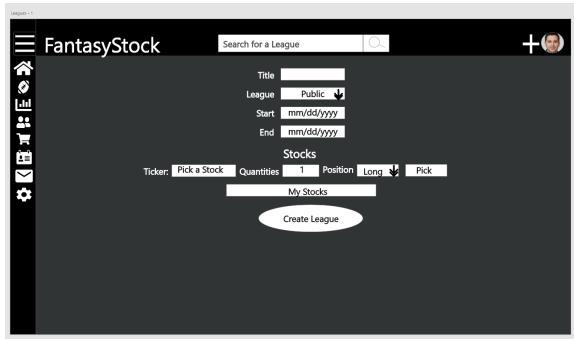


Software Design Document

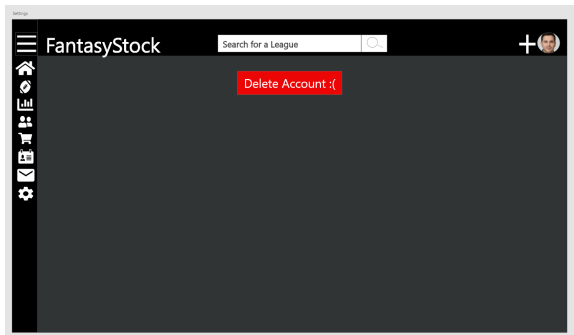
In this use case, opening the menu, allows users to use the sidebar on the top corner to open and see the lists of all the icons in words.



This use case allows all users to create their own league and pick their stock. They are able to change the date from start to finish and determine the name of their stock.

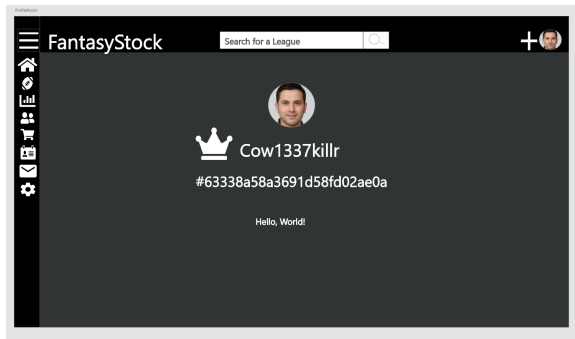


The use case of deleting one's own account.

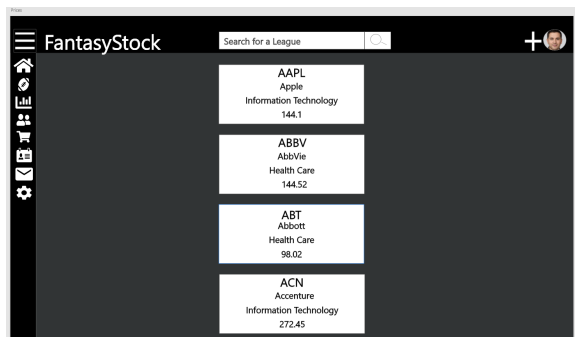


Software Design Document

The use case profile. They are able to change their username and bio. They are also able to change their icon next to their username when purchasing from the shop.



This user case shows the pricing of the stocks. Users are able to see the price changes and see where the values are based on the stock API.



Software Design Document

Section 8 - Other Interfaces

Identify any external interfaces used in the execution of this module, include technology and other pertinent data

8.1 Interface X

This application interfaces with several APIs such as Google Authentication, Alpha Vantage, and News API. Alpha Vantage and News API are interfaces with an API key and used in the request. Google authentication uses a client id and secret provided to authenticate the user through them for the backend, which is then used for authorization and authentication in middleware.

Section 9 - Extra Design Features / Outstanding Issues

No extra design features or outstanding issues for this project.

Section 10 – References

No documents are necessary or useful to understand this document.

Section 11 – Glossary

League: A game where people join together and select their portfolio to face off and see who's portfolio is best.