don’twastetime

Fundamentals of Investment

don’twastetime
Group 4
IS 4430-090
Professor Repak
Fall 2020

Table of Contents

Team Member Introductions

Member 1: Tyler Barr - I am currently a software developer at Monnit corporation here in Salt Lake. I mostly do back end development, fixing bugs, and adding new features. Some of my interests are movies/tv, spending time with family and friends, soccer, and playing video games.

Member 2: Bailey Goto - I am currently the information systems program assistant where I create weekly, monthly, and semester reports regarding undergraduate and masters students in the IS program. I also have certifications in Microsoft SQL server and DOMO, as well as some knowledge of Java programming. I like to play guitar, learn languages, and skateboarding.

Member 3: Gavin Brown - I am currently an Information Systems student and I have had experience with web development (Java, SQL, PHP). I enjoy cooking, reading, and camping.

Member 4: Jacob Henry - I am in the process of creating an e-commerce website for my current employer which will be a new and separate business entity. I enjoy playing the drums, hanging out with friends and family, and outdoor activities.

Member 5: Josh Adams - I work as a full stack developer for an LDS dating service. I mainly work on a portal tool that allows employees to maintain the user base. The primary frameworks/languages I use are Node, React, and MySQL. I enjoy skateboarding, watching movies, and playing video games.

DISCLAIMER:

This project is original and is not being utilized in any other courses or institutions. Some material contained within this report was inspired by the two project samples provided.

Project Selection and Analysis Report

Executive Summary

Problem

Many people may believe that the funds available in their bank account will increase over time due to interest and eventually become a comfortable number. However, the average annual interest rate for a savings account is only a meager 0.09%, which is hardly enough to outpace the 3% average annual rate of inflation. Only about 18% of 18 to 25-year-olds are currently investing in stocks, bonds, and other financial tools. The other 82% of young adults' financial assets gradually decrease in value, leaving them to suffer in the future.

Investment is a crucial tool that allows one's wealth to grow exponentially over time. Unfortunately, many are ignorant of its functionality and in consequence, are afraid to even start researching the market. The S&P 500 index tracks 500 large companies listed in the stock market and has had an average annual return of 13.6% in the past 10 years. If an investor were to begin investing at the age of 18 and contributed $20 per month, they would have the potential to earn $178,528 if they were to retire at age 60. This means you could potentially lose out on $178,528 if you are unaware of investment principles. It is paramount that people begin investing as early as possible.

Solution

In order to guide the fearful and unmotivated young adults into opening their first investment account, we have created don'twastetime, an application that will transform users from trading novices into seasoned investors. The platform will accomplish this by presenting weekly lessons. These lessons begin with the fundamentals of the stock market and the essentials of trading. Over time, the lessons gradually become more complex and involved. In addition to these lessons, we offer a simulated stock market that allows the users to apply what they have learned throughout the lessons. As the user becomes more comfortable and hones their knowledge, they will be able to easily transition into opening their own account at a brokerage firm and begin growing their wealth.

Measurable Impact

don'twastetime will allow users to gain an understanding of the market and its tools in a quick and comprehensive manner. The app will leverage young user's abundance of time to start increasing their wealth earlier in life so that they are able to achieve their financial goals. don'twastetime's mission is to increase the percentage of young investors and to aid them in reaching financial success.

Target Actors

End Users: The end user will interact with our application in order to gain an understanding for the fundamentals of investment. They will also utilize our application to engage with our simulated stock market in order to apply the knowledge they have attained through our application.

System Administrators: System administrators will be responsible for maintaining our applications security, reliability, accuracy, and functionality. System administrators must monitor each aspect of the application to ensure the stability of the application.

Cloud Services: MS Azure is a secondary actor in our system. We will utilize MS Azure to handle our databases. Azure will be interacting with our system to provide cloud services.

Social Media Platforms: Social media platforms are secondary actors in our system. Social media platforms will be utilized to share information about our users progress. Our system will be sharing progress updates to social media platforms.

Requirements Gathering

Our team chose to implement an online survey and to conduct interviews with potential users to gather requirements for our system. We administered our survey to 50 individuals. The survey questions were designed to prioritize system requirements and to understand what is important to potential users. The survey questions and results are reported on page 6.

Our team utilized interviews to receive answers to open-ended questions regarding our application and users' inclination towards investing. The interview questions we used are reported on the following page.

Interview Questions

What features do you think would be most important for a software that is designed to help first-time investors learn the fundamentals of investing?

What would be the most efficient teaching method for instilling the fundamentals of investing into a first-time investor?

If you were a first-time investor, what information would you seek first?

How much experience do you have with investing?

Would you prefer a basic introduction to how to use a program or figure it out on your own?

What could be some reasons why some people do not start investing?

If investing could be simplified, do you feel like you would be inclined to consistently invest?

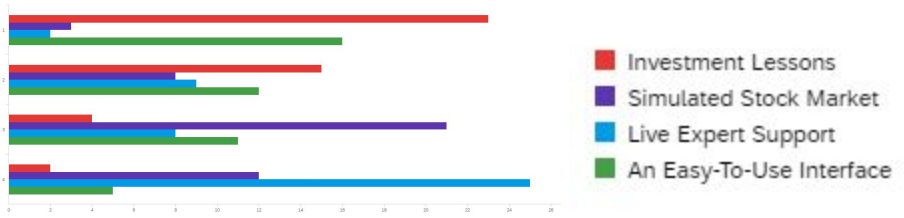
When you think about investing, the stock market, finance, etc. what are your first thoughts?

What are your goals with investing?

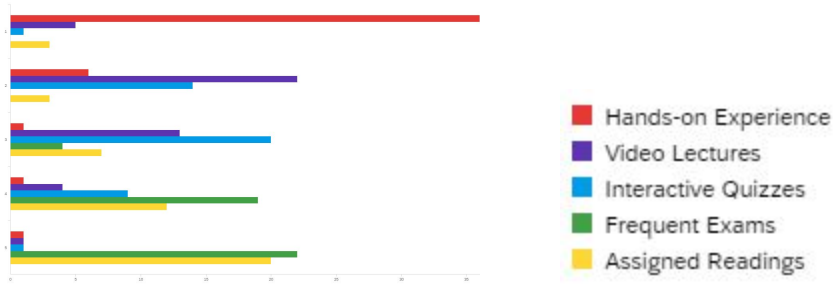
What kind of trading do you want to do? (i.e. Stocks, bonds, options, or dividends)

Survey Questions and Results

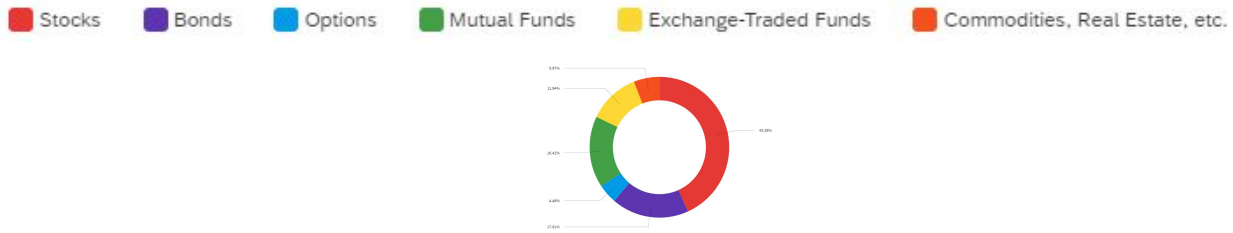
What features do you think would be most important for a software that is designed to help first-time investors learn the fundamentals of investing? Please rank the following according to their importance with number 1 being most important.



When learning about a subject, which of the following is most important? Please rank the following with number 1 being most important.

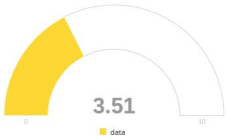


What kinds of investments have you owned/interacted with in the past? (Mark all that apply)

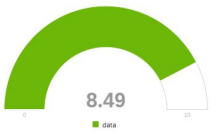


On a scale of 1-10, how interested are you in investing?

On a scale of 1-10, 1 being novice level, 10 being expert level, how would you rank your knowledge of investments and financial tools?

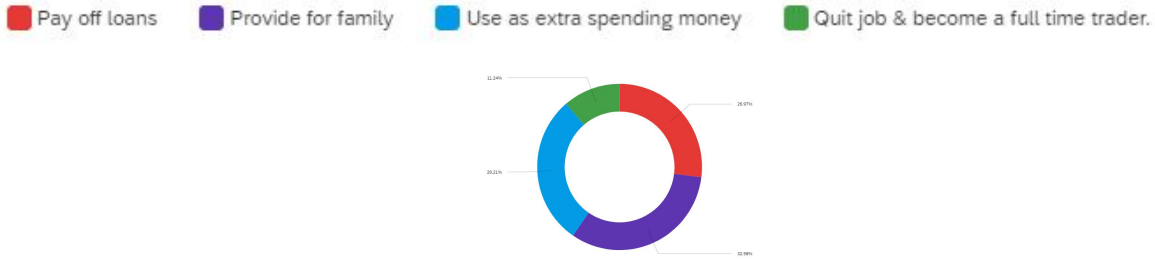


I believe it is important to understand the fundamentals of investment.

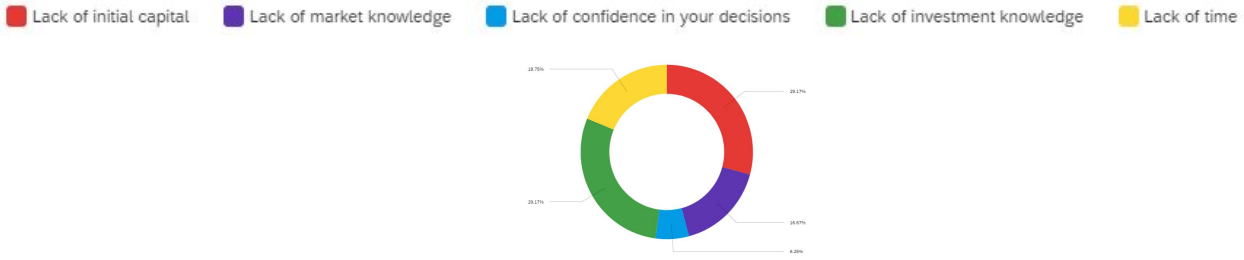


I believe it is important to make money.

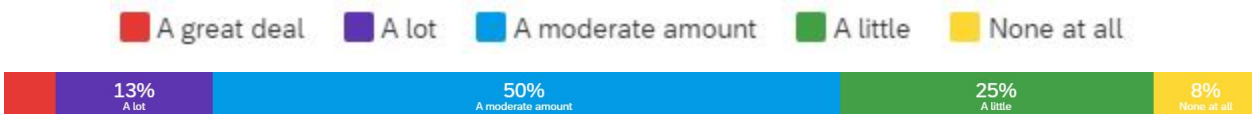
What will you do with your earnings from investments?



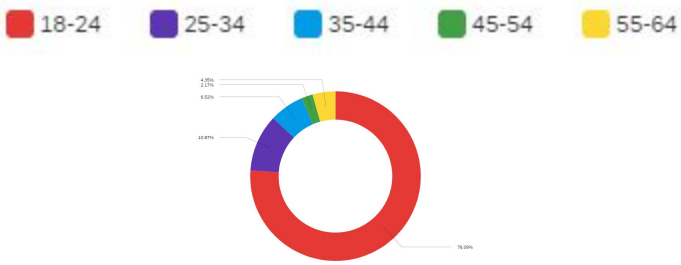
What do you feel keeps you from investing?



What is your tolerance for risk when it comes to investing?



Please select your age group.



Data Summary

The survey and interviews were conducted to obtain information about users and their understanding of investment. It was found that the majority of users are younger and less experienced investors. The majority of responses indicated that users had experience investing in stocks, bonds, mutual funds, and ETFs. The responses also indicated that users are quite interested in investing and understand the importance of it. However, it was also found that the majority felt they lacked the knowledge necessary to invest. don'twastetime's goal is to provide an educational, hands-on experience to solve this issue for it's users.

The study shows that users believe investment lessons, a simulated stock market, and an easy-to-use interface were the most important features for the application. The study also shows that hands-on experience, video lectures, and interactive quizzes were the most important teaching methods to utilize in our investment lessons. These teaching methods and features will be implemented to maximize user efficiency and success.

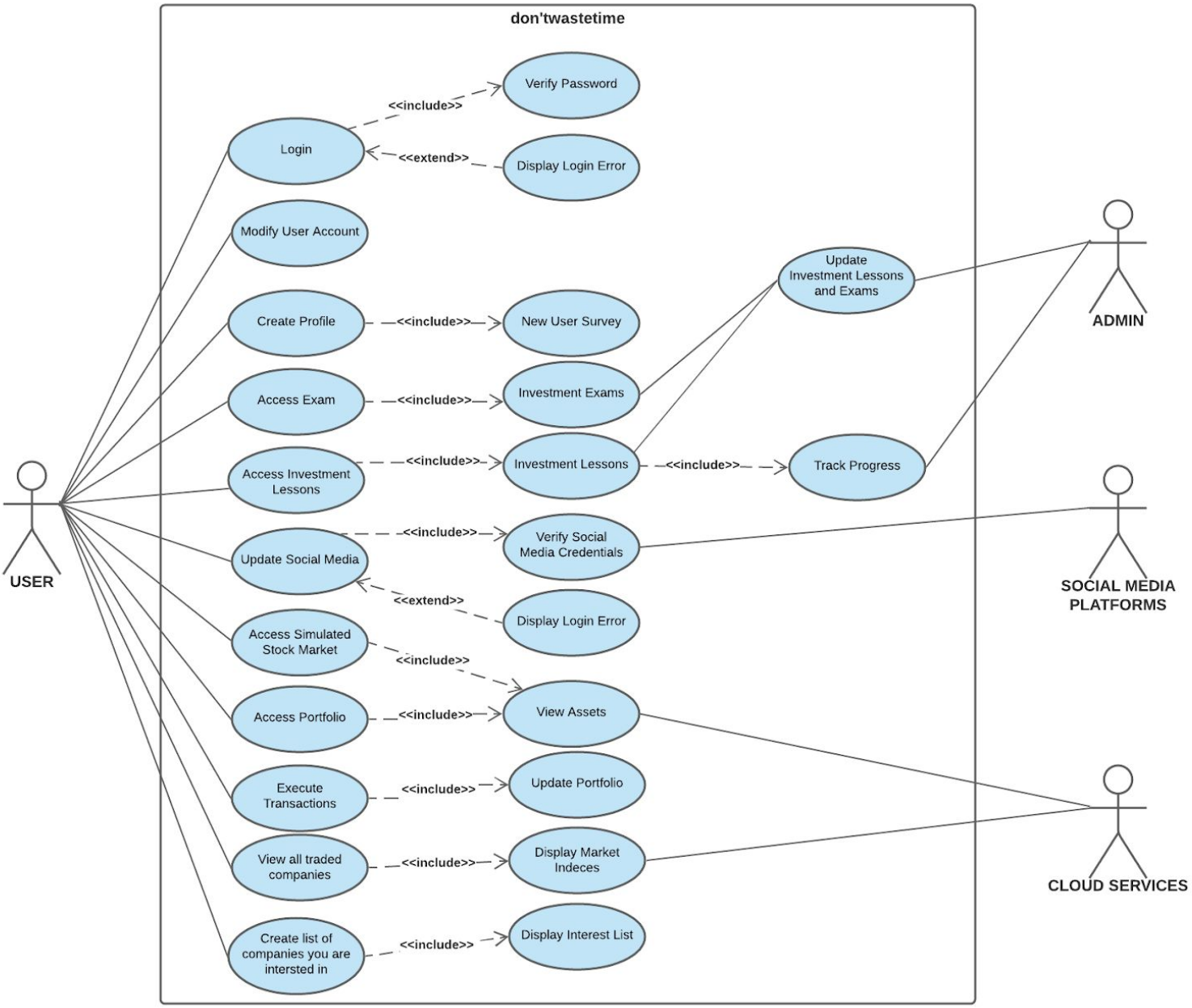
don'twastetime strives to help its users achieve their investment goals. The survey included a question regarding users plans for the profits they gain through investing. It was observed that the majority would use their earnings to provide for their families, pay off loans, and to use as discretionary income. don'twastetime is intended to be a tool that will support it's users in achieving their investment goals.

User Stories and Acceptance Criteria

User Story	Acceptance Criterion	Requirements	Type	Priority Level 1-5	Effort Level 1-5
As a user of don'twastetime, I need to be able to access and interact with the simulated stock market so that I can apply the knowledge I have attained through the investment courses.	Users can navigate to simulated stock market. Users can interact with the stock market functions	The system needs to include a simulated stock market.	F	5	5
As a user of don'twastetime, I need to be able to access and interact with the investment lessons so that I am able to gain an understanding of the concepts.	Users can navigate to the investment lessons. Users can interact with the investment lessons.	The system needs to include investment lessons.	F	5	5
As a user of don'twastetime, I need to be able to view an interface that breaks down my entire portfolio stock market so that I can better understand how my investments are spread.	Users can understand and see where their money is. Users can see profits/losses on particular investments	The system needs to include a portfolio breakdown.	F	5	3
As a user of don'twastetime, I need to be able to add stocks to a list so that I can follow a list of companies that I am interested in investing in.	Users can follow companies they are interested in.	The system needs to include all companies on the stock exchange. With ability to add them to a separate list.	F	3	2
As a user of don'twastetime, I need to be able to share my learning progress on social media websites so that I have a form of accountability.	Users have easy access to sharing functions from app to other social media	The system needs to link social media accounts to app	F	2	2
As a user of don'twastetime, I need to be able to utilize an intuitive and attractive interface so that I have easy accessibility to all functions of the app	Users can intuitively understand basic app features with little error	The system needs to be intuitive and feature an attractive UI.	NF	4	3

User Story	Acceptance Criterion	Requirements	Type	Priority Level 1-5	Effort Level 1-5
As a user of don’twastetime, I need to easily view my total funds I’ve invested so that I can evaluate future investments.	Users can view their total funds invested by viewing their profile.	The system needs to accurately display the total funds invested.	F	2	1
As a user of don’twastetime, I need to be able to take a new user survey that will place me at an appropriate starting point for my skill level.	Users can access new user survey	The system needs to administer new user survey upon user account creation	F	5	5
As a user of don’twastetime, I need to be able to view and modify my personal/account information so that I can keep it updated	Users can access and change basic information about their account.	The system must include a basic account page to display this information and save changes	F	3	2
As a user of don’twastetime, I need to be able to view completed lessons and scores so that I can review my progress	Users can review their progress through lessons offered in the app Users can review their past performance on any exam or the simulated stock market	The system must display user progress through content on a separate page or integrated with an existing page (simple progress bar, with nodes representing test scores)	F	3	3
As a user of don’twastetime, I want to be able to access the application in an efficient and reliable manner so that I am engaged with the application.	Users can interact with the application in an efficient and reliable manner	The application must be efficient and reliable.	NF	3	4

Use Case Diagram



Use Case Narratives

Use Case Name	**Access Investment Lessons**
Description	Provide users with access to our investment lessons.
Actors	App User
Pre-Condition	User has created an account and navigated to the investment lessons within the application.
Post-Condition	Users will complete investment lessons and eventually complete investment quizzes.
Basic Success Flow	
1. The user launches the application, which opens the home page 2. The user taps on the icon labeled “Investment Lessons” 3. The user is presented with the first lesson 4. The user understands the lesson and proceeds to the following lessons	
Variations In Success Flow	
1. The user launches the application, which opens the home page 2. The user taps on the icon labeled “Investment Lessons” 3. The user is presented with the first lesson 4. The user does not understand the lesson and chooses to take a quiz to help reinforce the lessons concepts 5. The user completes the quiz and can continue taking the investment lessons	
Alternate Paths	
1. The user launches the application, which opens the home page 2. The user taps on the icon labeled “Simulated Stock Market” 3. The user is presented with the simulated stock market information 4. The user begins building their portfolio 5. The user lacks confidence in their investment knowledge and taps the icon labeled “Help” 6. The user is redirected to the investment lessons to reinforce their knowledge	
Related Use Cases	Access simulated stock market

Use Case Name	**Access Simulated Stock Market**
Description	Provide users with access to our simulated stock market
Actors	App User
Pre-Condition	User has created an account and navigated to the simulated stock market within the application.
Post-Condition	Users will build their portfolio and begin trading in the simulated stock market.
Basic Success Flow	
1. The user launches the application, which opens the home page 2. The user taps on the icon labeled “Simulated Stock Market” 3. The user is presented with the simulated stock market 4. The user builds their portfolio 5. The user begins interacting with and trading on the simulated stock market	
Variations In Success Flow	
1. The user launches the application, which opens the home page 2. The user taps on the icon labeled “Simulated Stock Market” 3. The user is presented with the simulated stock market 4. The user attempts to build their portfolio but lacks confidence in their decisions 5. The user taps the icon labeled “Help” 6. The user is redirected to the investment lessons to reinforce their knowledge 7. The user navigates back to the simulated stock market and applies their knowledge.	
Alternate Paths	
1. The user launches the application, which opens the home page 2. The user taps on the icon labeled “Simulated Stock Market” 3. The user is presented with the simulated stock market information 4. The user begins building their portfolio 5. The user lacks confidence in their investment knowledge and taps the “Help” icon 6. The user is redirected to the investment lessons to reinforce their knowledge	
Related Use Cases	Access investment lessons

Use Case Name	**Update Social Media**
Description	Provide ways in which primary actors can share learning progress to others on personal social media accounts
Actors	Primary actors (app users)
Pre-Condition	User has completed at least one lesson
Post-Condition	Users have connected their social media accounts and have access to sharing function

Basic Success Flow

1. The user launches the application, which opens the home page
2. The user is prompted to create an account with email, username, and password
3. Upon creation of the account, the user is presented with an option to connect social media accounts (Instagram, Twitter, Facebook, etc.)
4. The user connects x account(s) to app account
5. The user taps on the icon labeled “Investment Lessons”
6. The user successfully completes a lesson
7. The user is presented with the option to share a post on x social media account(s)
8. The user selects x social media account and posts their progress on their personal site.

Variations In Success Flow

1. The user launches the application, which opens the home page
2. The user is prompted to create an account with email, username, and password
3. Upon creation of the account, the user is presented with an option to connect social media accounts (Instagram, Twitter, Facebook, etc.)
4. The user exits connection option without linking any social media account(s)
5. The user taps on the icon labeled “Investment Lessons”
6. The user successfully completes a lesson
7. The user is presented with the option to connect social media account(s)
8. The user selects to connect x account(s) to app account
9. The user is presented with the option to share a post on x social media account(s)
10. The user selects x social media account and posts their progress on their personal site.

Alternate Paths

1. The user launches the application, which opens the home page
2. The user is prompted to create an account with email, username, and password
3. Upon creation of the account, the user is presented with an option to connect social media accounts (Instagram, Twitter, Facebook, etc.)
4. The user connects x account(s) to app account
5. The user taps on the icon labeled “Simulated Stock Market”
6. The user is presented with the simulated stock market
7. The user builds their portfolio
8. The user begins interacting with and trading on the simulated stock market
9. The user receives a notification regarding a notable gain on portfolio/asset
10. The user is presented with the option to share a post on x social media account(s)
11. The user selects x social media account and posts their progress on their personal site.

Related Use Cases	Access investment lessons, Access simulated stock market

Use Case Name	**Update Investment Lessons and Exams**
Description	System will integrate intermittent updates on investment lessons and exams
Actors	Secondary (systems admin)
Pre-Condition	Investment Lessons and Exams are functional and operational
Post-Condition	Updated lessons and exams are accessible to primary users
Basic Success Flow	
1. System requests scheduled update to systems administrator 2. Opens back-end software in test environment 3. Inserts new investment lessons and/or exams 4. Closes back-end software in test environment 5. Runs application in test environment 6. Application successfully implements updates 7. Publish and runs application in operational environment	
Variations In Success Flow	
1. System requests scheduled update to systems administrator 2. Opens back-end software in test environment 3. Inserts new investment lessons and/or exams 4. Closes back-end software in test environment 5. Runs application in test environment 6. Application fails to implement updates 7. Fix bugs in software 8. Rerun application in test environment 9. Application successfully implements updates 10. Publish and run application in operational environment	
Alternate Paths	
1. System requests scheduled update to systems administrator 2. Dismisses alert as there is no new lessons or exams	
Related Use Cases	

Use Case Name	**New User Survey**
Description	Users will take a survey that tests their investment knowledge level and will assign them their appropriate starting point for investment lessons
Actors	Primary actor (users)
Pre-Condition	New User Survey is complete
Post-Condition	User is able to access and complete survey
Basic Success Flow	
1. User creates new account with email, username, and password 2. User is prompted to take new user survey 3. User completes new user survey 4. System assigns appropriate starting point for user's investment knowledge level 5. User begins investment lessons	
Variations In Success Flow	
1. User creates new account with email, username, and password 2. User is prompted to take new user survey 3. User selects "I know my level" 4. System redirects user to "investment lesson tests" 5. User will select investment lesson tests to test into higher levels of investment knowledge 6. User begins investment lessons based on score	
Alternate Paths	
1. User creates new account with email, username, and password 2. User is prompted to take new user survey 3. User completes new user survey 4. System assigns appropriate starting point for user's investment knowledge level 5. User selects "Test into higher level" 6. System redirects user to "investment lesson tests" 7. User will select investment lesson tests to test into higher levels of investment knowledge 8. User begins investment lessons based on score	
Related Use Cases	UI/UX

Use Case Name	**Access Portfolio**
Description	Users will be able to view their total funds invested through their account.
Actors	Primary actor (users)
Pre-Condition	Account has been created.
Post-Condition	Users are able to view their funds invested.
Basic Success Flow	
1. User launches the app with a pre-existing token. 2. User taps on the profile icon. 3. User selects the “My Investments” tab. 4. System returns a graph showing each investment within an accumulative pie chart if funds have been invested.	
Variations In Success Flow	
1. User launches the application, which opens the home page 2. User is prompted to create an account with email, username, and password 3. Upon creation of the account, the user is presented with an option to connect social media accounts (Instagram, Twitter, Facebook, etc.) 4. User exits connection option without linking any social media account(s) 5. User taps on the profile icon. 6. User selects the “Portfolio” tab. 7. If no funds have been invested the system returns a header prompting them to start investing.	
Alternate Paths	
1. User launches the application, which opens the home page 2. User is prompted to create an account with email, username, and password 3. Upon creation of the account, the user is presented with an option to connect social media accounts (Instagram, Twitter, Facebook, etc.) 4. The user connects x account(s) to app account 5. User taps on the profile icon. 6. User selects the “My Investments” tab. 7. System returns a graph showing each investment within an accumulative pie chart if funds have been invested.	
Related Use Cases	-

Use Case Name	**Track Progress**
Description	Users will be able to view their lessons scores through their account page.
Actors	Primary actor (users)
Pre-Condition	Account has been created and has taken at least 1 lesson
Post-Condition	Users can view their lesson scores.
Basic Success Flow	
1. User launches the app with a pre-existing token. 2. User taps on the profile icon. 3. User selects the “Lessons” tab. 4. System returns data showing each lesson taken and the score received.	
Variations In Success Flow	
1. User launches the application, which opens the home page 2. User is prompted to create an account with email, username, and password 3. Upon creation of the account, the user is presented with an option to connect social media accounts (Instagram, Twitter, Facebook, etc.) 4. User exits connection option without linking any social media account(s) 5. User taps on the profile icon. 6. User selects the “My Lessons” tab. 7. No lessons have been started/finished so the system returns a header prompting them to start learning.	
Alternate Paths	
1. User launches the application, which opens the home page 2. User is prompted to create an account with email, username, and password 3. Upon creation of the account, the user is presented with an option to connect social media accounts (Instagram, Twitter, Facebook, etc.) 4. The user connects x account(s) to app account 5. User taps on the profile icon. 6. User selects the “My Lessons” tab. 7. System data showing each lesson taken and the score received	
Related Use Cases	-

Use Case Name	**Modify User Account**
Description	Displays users' basic account information and allows changes to be made
Actors	App User
Pre-Condition	User has created an account an navigated to the account page
Post-Condition	User can view the information associated with their account and make changes

Basic Success Flow

1. User launches the app, taking them to the home page
2. User selects the 'Profile' icon
3. User selects the 'Settings' tab
4. System displays all profile information
5. User must input correct login credentials in order to manage/change profile info

Variations In Success Flow

1. User launches the app, taking them to the home page
2. User selects the 'Profile' icon
3. User selects the 'Settings' tab
4. System displays profile information
5. User successfully inputs correct login credentials
6. All fields containing profile information become mutable
7. User may select to revert or save once changes have been made

Alternate Paths

1. User launches the app, taking them to the home page
2. User selects the 'Profile' icon
3. User selects the 'Settings' tab
4. System displays profile information
5. User fails to successfully enter the correct login credentials and remains on profile page containing immutable profile information
6. User successfully inputs correct login credentials and make changes to profile info
7. User selects revert changes and the original information is restored, requiring the user to input login credentials to edit again

Related Use Cases	-

Use Case Name	**Execute Transactions**
Description	Users are able to place orders for securities using the simulated stock market.
Actors	App user
Pre-Condition	User has opened trading account
Post-Condition	User has successfully traded a security
Basic Success Flow	
1. User logs into account 2. Opens User account 3. Opens simulated stock market 4. Creates portfolio for trading securities 5. Searches and finds desired security on simulated stock market 6. Trades security at given conditions and price 7. Order is processed 8. Order placement success pop-up 9. Security is found in User portfolio	
Variations In Success Flow	
1. User logs into account 2. Opens User account 3. Opens simulated stock market 4. Creates portfolio for trading securities 5. Searches and finds desired security on simulated stock market 6. Trades security at given conditions and price 7. Order is processed 8. Error pop-up 9. Readjust order given constraints of error 10. Reorder 11. Order is processed 12. Order placement success pop-up 13. Security is found in User portfolio	
Alternate Paths	
1. User logs into account 2. Opens User account 3. Opens simulated stock market 4. Creates portfolio for trading securities 5. User closes simulate stock market	
Related Use Cases	Simulated stock market

Use Case Name	**View All Traded Companies**
Description	Users can follow companies they are interested in.
Actors	App User
Pre-Condition	User has an account
Post-Condition	User adds companies they want to track to various lists
Basic Success Flow	
1. User logs into their account 2. User is brought to their home screen with overall portfolio performance 3. User clicks the search icon 4. User searches for a publicly traded company they are interested in 5. User sees how they are currently performing 6. User clicks '+' icon on page that adds the company to a preexisting list or has the user create and then add that company to a new list	
Variations In Success Flow	
1. User logs into their account 2. User is brought to their home screen with overall portfolio performance 3. User clicks the search icon 4. User searches for a publicly traded company they are interested in 5. User sees how they are currently performing for various time frames 6. User clicks '+' icon on page that adds the company to a preexisting list or has the user create and then add that company to a new list	
Alternate Paths	
1. User logs into their account 2. User is brought to their home screen with overall portfolio performance 3. User sees list of companies they are tracking 4. User clicks on one of these companies to see how it is performing 5. At the bottom of the page, the app shows list of similar companies 6. Gives user the option to add them to a list	
Related Use Cases	-

Project Plan

Pre-implementation Schedule

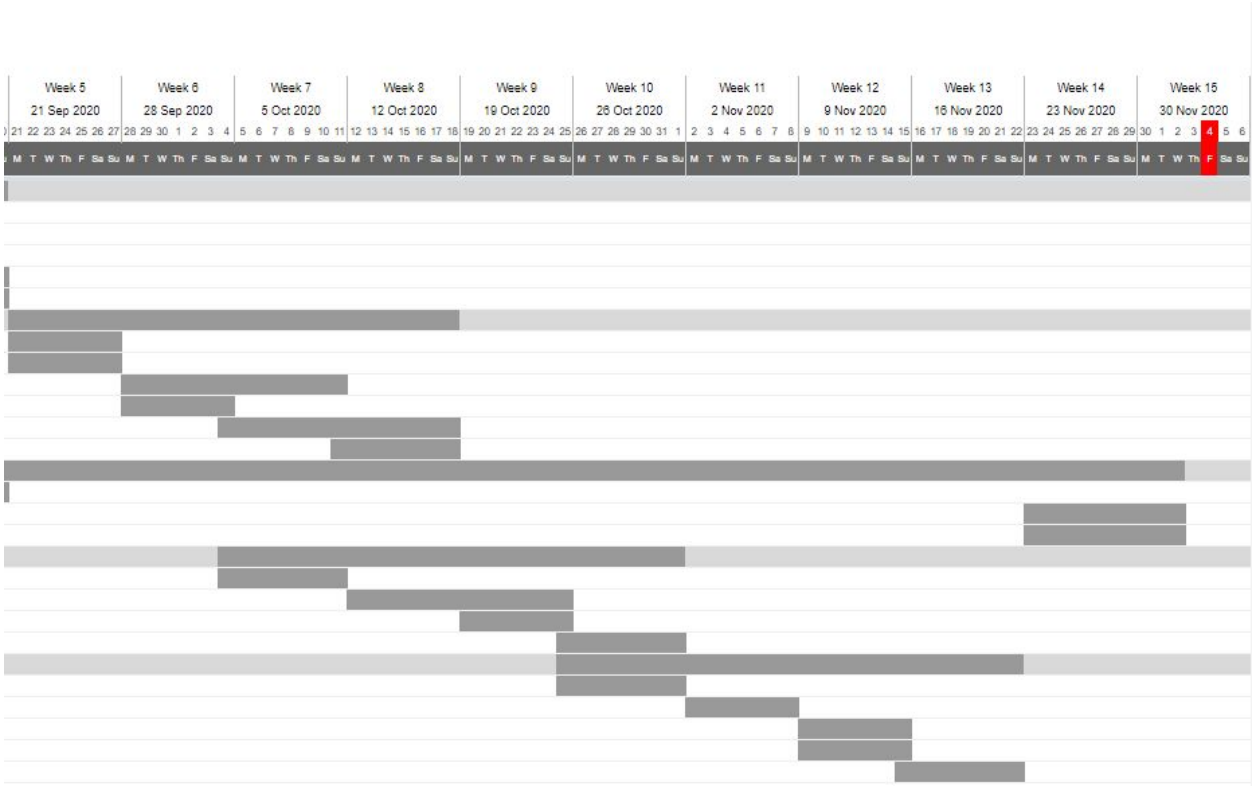
don'twastetime Project Schedule

don'twastetime

Project Start Date: 8/24/2020 (Monday) Display Week: 1

Project Manager: Team 4

WBS	Task	Start	End	Days	% Done	Work Days
1.3	Review Sample Executive Summary	Mon 9/07/20	Sun 9/13/20	6	100%	5
1.4	Begin Writing Team Executive Summary	Sun 9/13/20	Sun 9/20/20	7	100%	5
1.5	Complete Selection & Feasibility Assignment	Thu 9/10/20	Sun 9/20/20	10	100%	7
2	**Requirements Gathering & Analysis**	Mon 9/21/20	Sun 10/18/20	27	100%	20
2.1	Target Actors identification and description	Mon 9/21/20	Sun 9/27/20	6	100%	5
2.2	Identify and create requirements gathering techniques / tools	Mon 9/21/20	Sun 9/27/20	6	100%	5
2.3	Gather requirements and perform analysis	Mon 9/28/20	Sun 10/11/20	13	100%	10
2.4	Create high-level scope (user stories)	Mon 9/28/20	Sun 10/04/20	6	100%	5
2.5	Write use case descriptions	Sun 10/04/20	Sun 10/18/20	14	100%	10
2.6	Create use case diagram	Sun 10/11/20	Sun 10/18/20	7	100%	5
3	**Project Plan**	Sun 9/13/20	Wed 12/02/20	80	100%	58
3.1	Create pre-implementation project schedule	Sun 9/13/20	Sun 9/20/20	7	100%	5
3.2	Calculate Software Development Costs	Mon 11/23/20	Wed 12/02/20	9	100%	8
3.3	Create post-design project schedule	Mon 11/23/20	Wed 12/02/20	9	100%	8
4	**Analysis Documentation**	Sun 10/04/20	Sun 11/01/20	28	100%	20
4.1	Create logical ERD	Sun 10/04/20	Sun 10/11/20	7	100%	5
4.2	Create logical DFD	Mon 10/12/20	Sun 10/25/20	13	100%	10
4.3	Build CRUD Matrix	Mon 10/19/20	Sun 10/25/20	6	100%	5
4.4	Perform Buy vs Build Analysis	Sun 10/25/20	Sun 11/01/20	7	100%	5
5	**Design Documentation**	Sun 10/25/20	Sun 11/22/20	28	100%	20
5.1	Create physical ERD	Sun 10/25/20	Sun 11/01/20	7	100%	5
5.2	Crate physical DFD	Mon 11/02/20	Sun 11/08/20	6	100%	5
5.3	Describe system architecture	Mon 11/09/20	Sun 11/15/20	6	100%	5
5.4	Describe data storage system	Mon 11/09/20	Sun 11/15/20	6	100%	5
5.5	Create mock user interfaces	Sun 11/15/20	Sun 11/22/20	7	100%	5



Post-Design Schedule

don'twastetime Post-Design Implementation Schedule

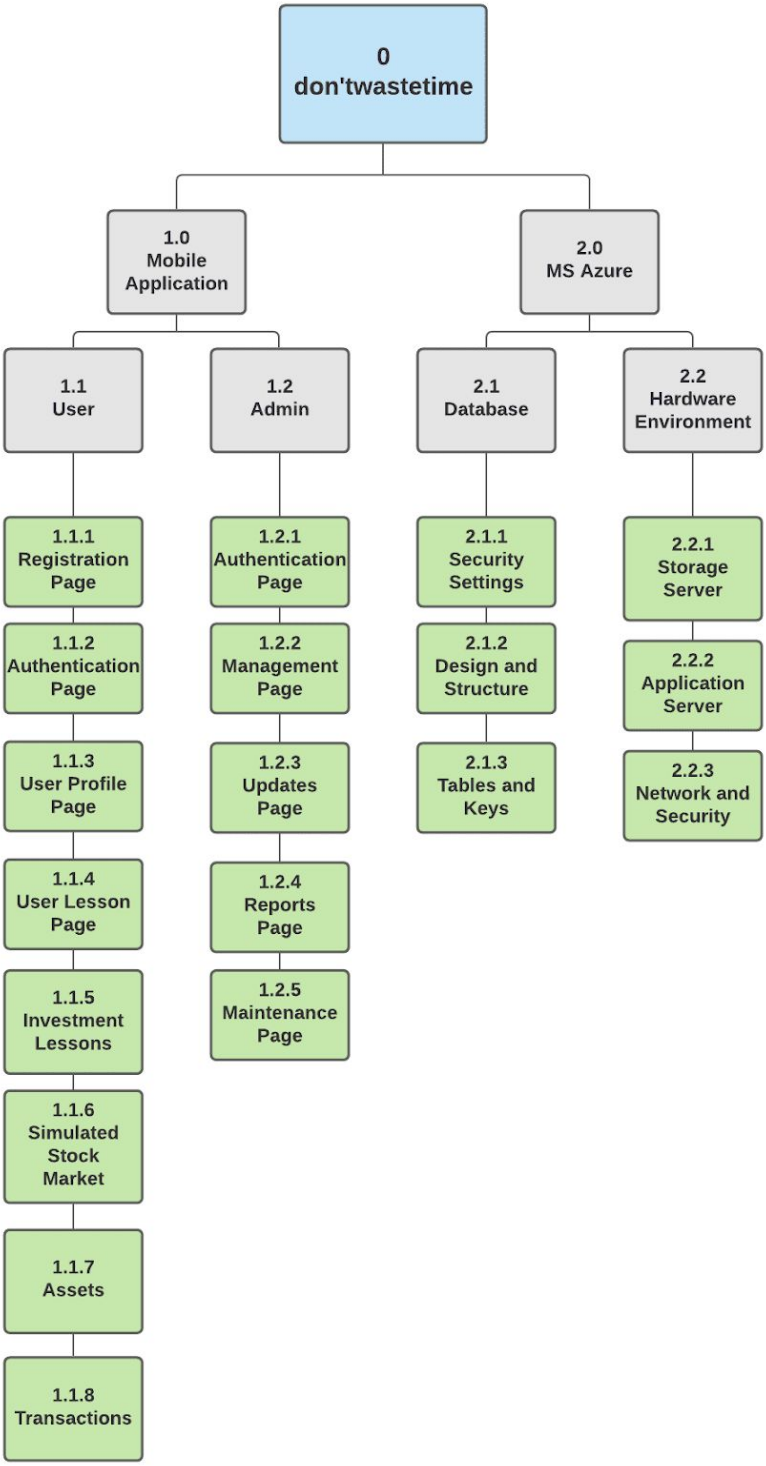
don'twastetime

Project Start Date: 12/6/2020 (Sunday) Display Week: 1

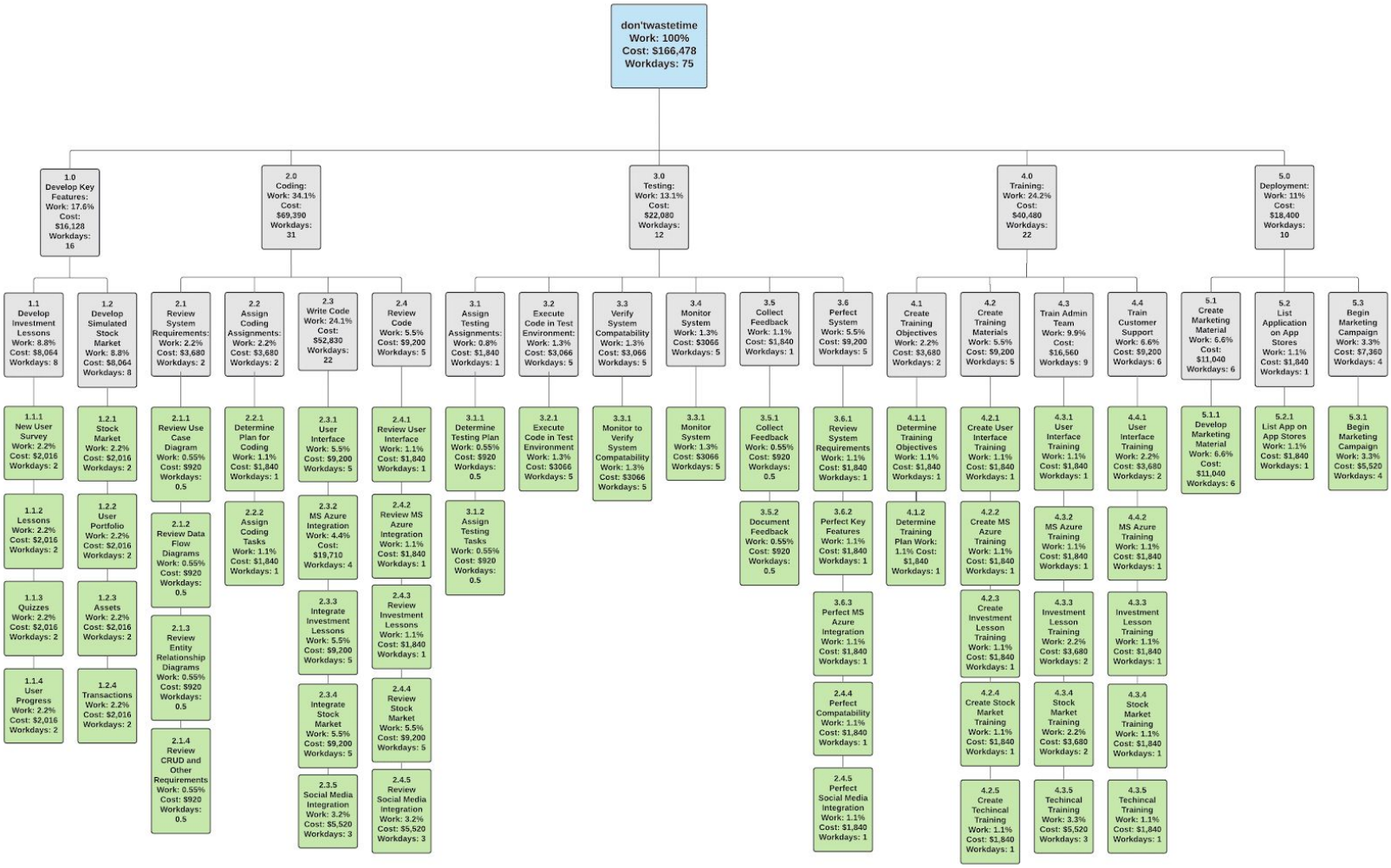
Project Manager: Team 4

WBS	Task	Start	End	Days	% Done	Work Days
1	**Develop Key Features**	Fri 12/11/20	Fri 1/01/21	21	0%	16
1.1	Develop Investment Lesson Material	Fri 12/11/20	Tue 12/22/20	11	0%	8
1.2	Develop Simulated Stock Market	Wed 12/23/20	Fri 1/01/21	9	0%	8
2	**Coding**	Mon 12/07/20	Mon 1/18/21	42	0%	31
2.1	Review System Requirements	Mon 12/07/20	Tue 12/08/20	1	0%	2
2.2	Assign Coding Assingments	Wed 12/09/20	Thu 12/10/20	1	0%	2
2.3	Write Code	Fri 12/11/20	Mon 1/11/21	31	0%	22
2.4	Review Code	Tue 1/12/21	Mon 1/18/21	6	0%	5
3	**Testing**	Tue 1/19/21	Wed 2/03/21	15	0%	12
3.1	Assign Testing Assignments	Tue 1/19/21	Tue 1/19/21	0	0%	1
3.2	Execute Code in Test Environment	Wed 1/20/21	Tue 1/26/21	6	0%	5
3.3	Verify System Compatability	Wed 1/20/21	Tue 1/26/21	6	0%	5
3.4	Monitor System	Wed 1/20/21	Tue 1/26/21	6	0%	5
3.5	Collect Feedback	Wed 1/27/21	Wed 1/27/21	0	0%	1
3.6	Perfect System	Thu 1/28/21	Wed 2/03/21	6	0%	5
4	**Training**	Thu 2/04/21	Fri 3/05/21	29	0%	22
4.1	Create Training Objectives	Thu 2/04/21	Sun 2/07/21	3	0%	2
4.2	Create Training Materials	Mon 2/08/21	Sun 2/14/21	6	0%	5
4.3	Train Admin Team	Mon 2/15/21	Thu 2/25/21	10	0%	9
4.4	Train Customer Support	Fri 2/26/21	Fri 3/05/21	7	0%	6
5	**Deployment**	Sat 3/06/21	Sat 3/20/21	14	0%	10
5.1	Create Marketing Material	Sat 3/06/21	Mon 3/15/21	9	0%	6
5.2	List Application on App Stores	Tue 3/16/21	Tue 3/16/21	0	0%	1
3.4	Begin Marketing Campaign	Tue 3/16/21	Sat 3/20/21	4	0%	4

Product-Oriented Work Breakdown Structure



Phase-Based Work Breakdown Structure



Estimation

Below is an estimation summary detailing the cost of hiring five software engineers to work on the development of the system and three investment specialists to develop the investment lesson material. We used average hourly rates for these employees to estimate the total labor cost. The average hourly wage information was sourced from Indeed.com and Glassdoor.com. For more information, please reference the references page on page 39.

Estimation Summary

			Software Engineer				Finance Professor				Investment Analyst				Investment Manager				
Task Type	Task	Workdays	Count	Wage	Hours	Cost	Count	Wage	Hours	Cost	Count	Wage	Hours	Cost	Count	Wage	Hours	Cost	Total Cost
Mobile Application	**User**	50	5	$46	2000	$92,000	1	$47	128	$6,016	1	$39	128	$4,992	1	$40	128	$5,120	$92,000.00
Mobile Application	Registration Page	3	5	$46	120	$5,520	0	$47	0	$0	0	$39	0	$0	0	$40	0	$0	$5,520.00
Mobile Application	Authentication Page	4	5	$46	160	$7,360	0	$47	0	$0	0	$39	0	$0	0	$40	0	$0	$7,360.00
Mobile Application	User Profile Page	5	5	$46	200	$9,200	0	$47	0	$0	0	$39	0	$0	0	$40	0	$0	$9,200.00
Mobile Application	User Lesson Page	5	5	$46	200	$9,200	0	$47	0	$0	0	$39	0	$0	0	$40	0	$0	$9,200.00
Mobile Application	Investment Lessons	8	5	$46	320	$14,720	1	$47	64	$3,008	1	$39	64	$2,496	1	$40	64	$2,560	$14,720.00
Mobile Application	Simulated Stock Market	8	5	$46	320	$14,720	1	$47	64	$3,008	1	$39	64	$2,496	1	$40	64	$2,560	$14,720.00
Mobile Application	User Portfolio	7	5	$46	280	$12,880	0	$47	0	$0	0	$39	0	$0	0	$40	0	$0	$12,880.00
Mobile Application	Assets	5	5	$46	200	$9,200	0	$47	0	$0	0	$39	0	$0	0	$40	0	$0	$9,200.00
Mobile Application	Transactions	5	5	$46	200	$9,200	0	$47	0	$0	0	$39	0	$0	0	$40	0	$0	$9,200.00
Mobile Application	**Admin**	19	5	$46	760	$34,960	0	$47	0	$0	0	$39	0	$0	0	$40	0	$0	$34,960.00
Mobile Application	Authentication Page	3	5	$46	120	$5,520	0	$47	0	$0	0	$39	0	$0	0	$40	0	$0	$5,520.00
Mobile Application	Management Page	2	5	$46	80	$3,680	0	$47	0	$0	0	$39	0	$0	0	$40	0	$0	$3,680.00
Mobile Application	Updates Page	5	5	$46	200	$9,200	0	$47	0	$0	0	$39	0	$0	0	$40	0	$0	$9,200.00
Mobile Application	Reports Page	5	5	$46	200	$9,200	0	$47	0	$0	0	$39	0	$0	0	$40	0	$0	$9,200.00
Mobile Application	Maintenance Page	4	5	$46	160	$7,360	0	$47	0	$0	0	$39	0	$0	0	$40	0	$0	$7,360.00
MS Azure	**Database**	3	5	$46	120	$5,520	0	$47	0	$0	0	$39	0	$0	0	$40	0	$0	$5,520.00
MS Azure	Security Settings	1	5	$46	40	$1,840	0	$47	0	$0	0	$39	0	$0	0	$40	0	$0	$1,840.00
MS Azure	Design and Structure	1	5	$46	40	$1,840	0	$47	0	$0	0	$39	0	$0	0	$40	0	$0	$1,840.00
MS Azure	Tables and Keys	1	5	$46	40	$1,840	0	$47	0	$0	0	$39	0	$0	0	$40	0	$0	$1,840.00
MS Azure	**Hardware Environment**	3	5	$46	120	$5,520	0	$47	0	$0	0	$39	0	$0	0	$40	0	$0	$5,520.00
MS Azure	Storage Server	1	5	$46	40	$1,840	0	$47	0	$0	0	$39	0	$0	0	$40	0	$0	$1,840.00
MS Azure	Application Server	1	5	$46	40	$1,840	0	$47	0	$0	0	$39	0	$0	0	$40	0	$0	$1,840.00
MS Azure	Network and Security	1	5	$46	40	$1,840	0	$47	0	$0	0	$39	0	$0	0	$40	0	$0	$1,840.00
Labor Cost Total:		75			3000	$138,000			128	$6,016			128	$4,992			128	$5,120	$154,128.00

Logical ERD Description

Entities

User: Contains user information and a unique identifier for each user.

User Portfolio: Contains portfolio information and unique identifier for each instance of a user's portfolio. Includes portfolio value, count of assets, and description.

User Lesson: Contains information about individual user's investment lessons and user progress.

Investment Lesson: Contains lesson information and a unique identifier for each lesson. Includes lesson type, descrition, and material.

Quiz: Contains quiz information and a unique identifier for each quiz. Includes quiz type, description, and tasks.

Transaction: Contains information about specific transaction instances made by users for their portfolio and a unique identifier for transactions. Includes date, type, price, and description of transaction.

Order: Contains order information and a unique identifier for each instance of an order. Includes assets included in order, asset count, order type, order date, order total, and an order description.

Asset: Contains asset information and a unique asset identifier. Includes asset type, value, count, and description.

Relationships and Cardinalities

USER has 0 or more USER LESSON, but USER LESSON belongs to one and only one USER

INVESTMENT LESSON includes 0 or more USER LESSON, but USER LESSON comes from one and only one INVESTMENT LESSON

USER LESSON includes 0 or one QUIZ, and QUIZ is part of 0 or one USER LESSON

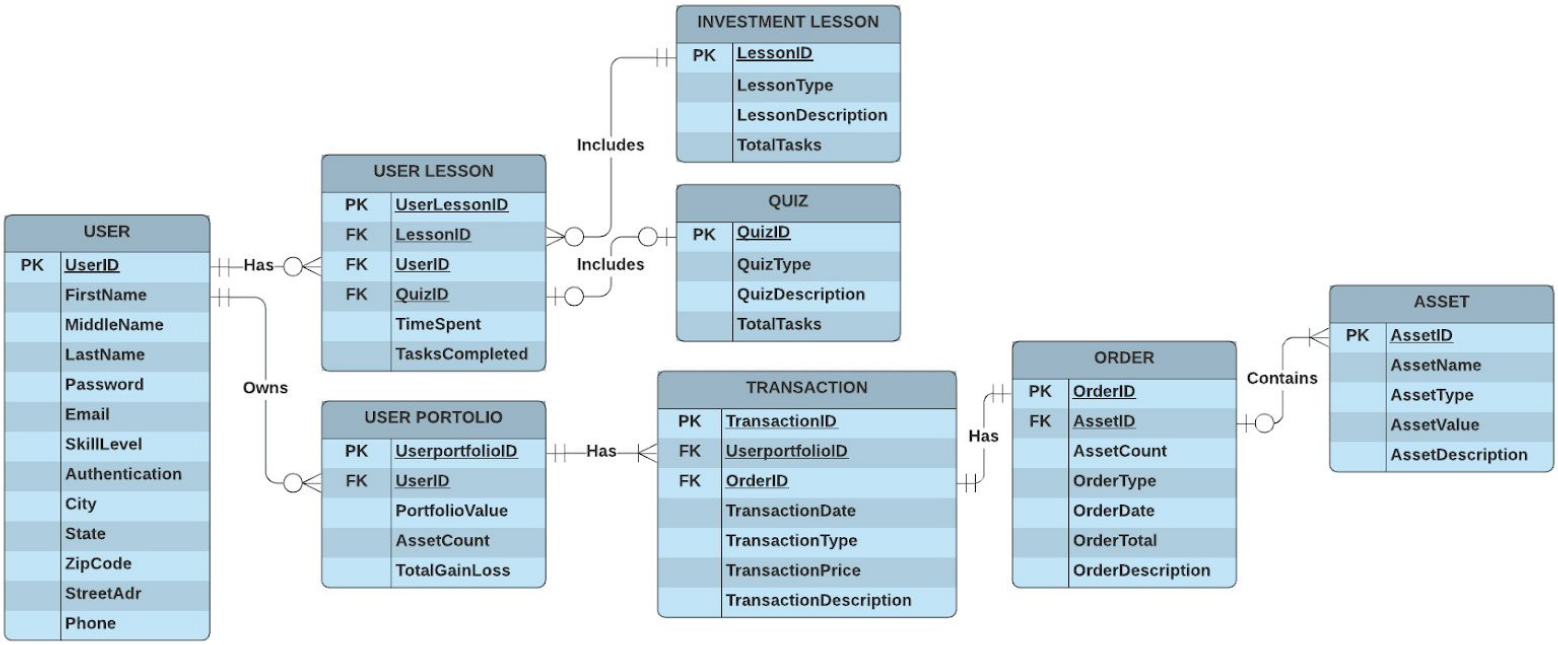
USER owns 0 or more USER PORTFOLIO, but USER PORTFOLIO belongs to one and only one USER

USER PORTFOLIO has one or more TRANSACTION, but TRANSACTION is sourced from one and only one USER PORTFOLIO

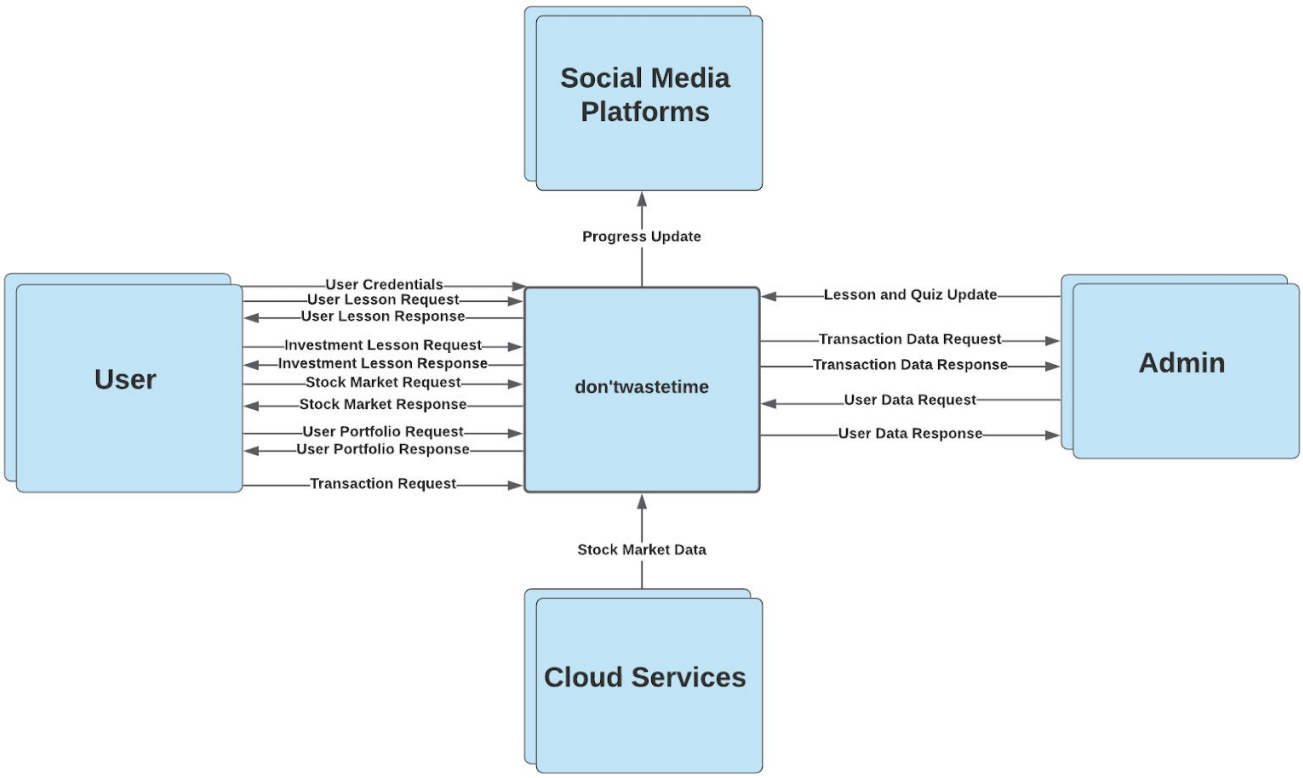
TRANSACTION has one and only one ORDER, and ORDER is part of one and only one TRANSACTION

ORDER contains one or more ASSET, but ASSET is part of 0 or one ORDER.

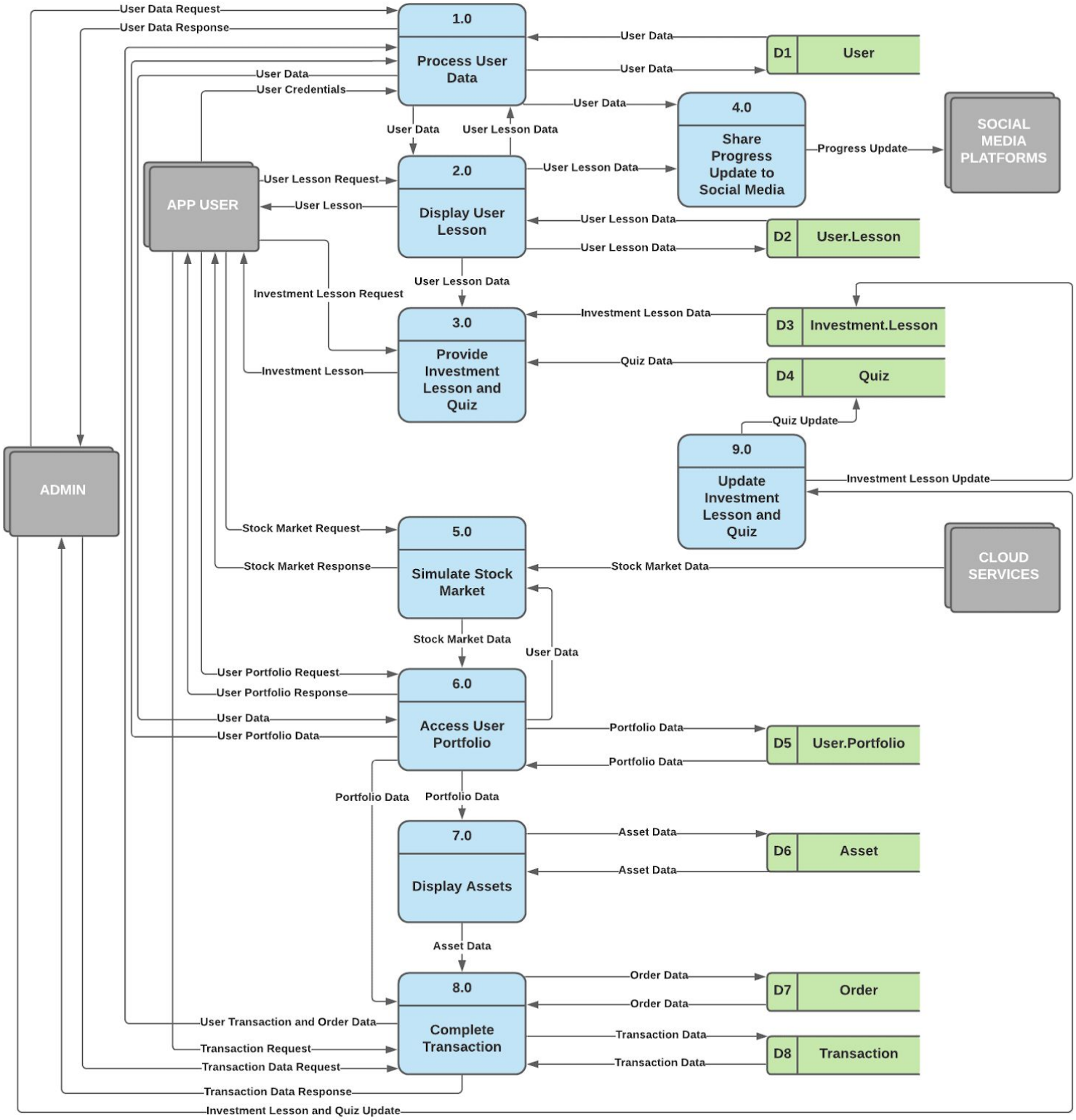
Logical ERD Model



Context Level DFD



Level 0 DFD


1.0
Process User Data
2.0
Display User Lesson
3.0
Provide Investment Lesson and Quiz
4.0
Share Progress Update to Social Media
5.0
Simulate Stock Market
6.0
Access User Portfolio
7.0
Display Assets
8.0
Complete Transaction
9.0
Update Investment Lesson and Quiz
D1 User
D2 User.Lesson
D3 Investment.Lesson
D4 Quiz
D5 User.Portfolio
D6 Asset
D7 Order
D8 Transaction
APP USER
ADMIN
SOCIAL MEDIA PLATFORMS
CLOUD SERVICES
User Data Request
User Data Response
User Data
User Credentials
User Lesson Data
User Lesson Request
User Lesson
Investment Lesson Request
Investment Lesson
Investment Lesson Data
Quiz Data
Quiz Update
Investment Lesson Update
Progress Update
Stock Market Request
Stock Market Response
Stock Market Data
User Portfolio Request
User Portfolio Response
User Portfolio Data
Portfolio Data
Asset Data
User Transaction and Order Data
Order Data
Transaction Request
Transaction Data
Transaction Data Request
Transaction Data Response
Investment Lesson and Quiz Update

CRUD Matrix

	User	User.Lesson	Quiz	Investment.Lesson	User.Portfolio	Asset	Transaction	Order
Process User Data	CRUD	R	-	-	R	-	R	R
Display User Lesson	R	R		-	-	-	-	
Provide Investment Lesson and Quiz	R	R	R	R	-	-	-	
Share Progress Update to Social Media	R	R		-	-	-	-	
Simulate Stock Market	R	-		-	-	-	-	-
Access User Portfolio	-	-		-	CRUD	-	-	
Display Assets	-	-		-	R	CRUD	-	-
Complete Transaction	-	-		-	R		CRUD	CRUD
Update Investment Lesson and Quiz	-	-	CRUD	CRUD	-	-	-	

Build vs Buy Analysis

Overview

don'twastetime's mission is to educate its users through an engaging and intuitive software focused on increasing both their investment knowledge and skill. The information gathered from the surveys and interviews indicates that the software must contain interactive investment lessons as well as a fully-functional simulated stock market. It is essential that the software incorporates these features seamlessly. The software must also be capable of all system requirements listed previously in this report. Below is an analysis for developing the software.

In-House Build

Completing an in-house build will allow our development team to gain an understanding for the software that would not be possible if it were developed outside of our organization. Developing in-house will also allow our team flexibility in their development and give us full access to every aspect of the development. An in-house build will also allow us to accommodate unique non-functional requirements of the software. However, an in-house build can also consume a great number of resources and time. It is important to remember the total cost of ownership when considering these options.

A software engineer makes $46 an hour on average. We estimate that the development of our system will require 75 workdays to complete with five engineers working simultaneously. Assuming the average workday is eight hours and it takes 75 workdays to complete, that is a total of 600 hours. Further, there are five software engineers working 600 hours. Therefore, a total of 3,000 hours will be necessary to complete this project. If this project takes 3,000 hours to complete and we pay the engineers $46 an hour, the labor cost for our software development would be $138,000.

To develop the investment lesson material and design the simulated stock market we plan to hire an investment analyst, an investment manager, and a finance professor. On average, an investment analyst makes $39 an hour, an investment manager makes $40 an hour, and a finance professor makes $47 an hour. We estimate that developing the investment lesson material will require 16 workdays to complete with the three employees working simultaneously. If the workday is eight hours and it requires 16 days to complete, that is a total of 128 hours. Further, the investment analyst will cost a total of $4,992, the investment manager will cost $5,120, and the finance professor will cost $6,016. Therefore, the total cost of labor for developing the investment lesson material will be $16,128.

In-House Build Estimated Labor Cost: ~$154,128

Evaluation Criteria	Relative Weight	**In-House Build**	Score (1-5)	Weighted Score
Technical Issues:				
Complexity	20	High Complexity	2	40
Amount of Work	10	High	2	20
Technical Risk	10	High Failure Risk	2	20
Economic Issues:				
Development	25	Large Development Costs	2	50
Operational	5	Operational Expenses	4	20
Organizational Issues:				
Scalability	10	Difficult	3	30
Flexibility	10	Acceptable	5	50
Compatibility	10	Compatible	5	50
TOTAL	**100**			**280**

Packaged System

Utilizing a packaged system would allow for faster development time because a base system is provided to customize to fit specific needs. This would reduce the amount of technical resources required for the initial development of the software. A packaged system is typically a reliable and proven software which will decrease the overall development time required to create a functional software. Packaged systems are also typically supported by their provider so maintenance and upgrades are included in the package as well. However, packaged systems may not be perfect. Some packaged systems may not be customizable and incapable of integrating with legacy systems.

Packaged System Estimated Cost: Varies Depending on Provider

Evaluation Criteria	Relative Weight	**Packaged System**	Score (1-5)	Weighted Score
Technical Issues:				
Complexity	20	Reduces Initial Complexity	3	60
Amount of Work	10	Moderate	3	30
Technical Risk	10	Proven Reliability May Require Customization	4	40
Economic Issues:				
Development	25	Initial Package Cost	3	75
Operational	5	Licensing and Maintenance Fees	3	15
Organizational Issues:				
Scalability	10	Not Guaranteed	2	20
Flexibility	10	Not Guaranteed	2	20
Compatibility	10	Not Guaranteed	1	10
TOTAL	**100**			**270**

Outsourcing

Outsourcing portions of the project will ultimately reduce costs by decreasing the use of in-house resources. It would also allow us to find third parties that are professionals in a specific area. Outsourcing a project to an expert third party can produce a more professional and efficient result than building in-house without the expertise of the third-party. However, outsourcing the project also eliminates the opportunity for in-house learning and depends entirely on the third party to deliver. Outsourcing can also compromise confidential information.

We have decided to use Microsoft's Azure general purpose cloud services to host our mobile app and its accompanying database. Because this is a Platform-as-a-Service (Paas), we will have plenty of administrative control over the virtual machines running the servers without

having to worry about the minutiae of maintaining and updating them ourselves. Assuming that our database will be memory-intensive, we will use the A7 Instance plan that offers 56 GB of RAM and an additional 2,040 GB of storage. This plan costs an estimated $1.41/hour, totalling $1,030 monthly or $12,350 annually assuming that the servers run nonstop.

Estimated Cost: $12,350 annually

Evaluation Criteria	Relative Weight	**Outsourcing**	Score (1-5)	Weighted Score
Technical Issues:				
Complexity	20	Requires Least In-House Resources	5	100
Amount of Work	10	Low	4	40
Technical Risk	10	Transferred to 3rd Party	4	40
Economic Issues:				
Development	25	Outsourcing Cost	4	100
Operational	5	Outsourcing and Maintenance Fees	2	10
Organizational Issues:				
Scalability	10	Scalable	5	50
Flexibility	10	Flexible	3	30
Compatibility	10	Not Guaranteed	3	30
TOTAL	**100**			**400**

Decision

We have decided to develop our software and key features in-house. Developing the system in-house will give us the control and understanding we require to perfect this system. We were unable to find a packaged software that would suit the needs of our system. However, we have decided to utilize Microsoft Azure for our cloud storage to outsource that process.

Design Documents

Architecture

Our client is presented through a mobile first browser based approach to allow for seamless use between phones. The stack will consist of React, Sass, Material UI and Flux. The crux of our front-end will be React.js to create the user interface for the client while maintaining easy re-usability within our codebase. React will be used in combination with the Material library to speed up development.

Our styling will be done with Sass in order to allow for easier readability and management. In terms of interacting with our API, we'll be using Axios for promise focused HTTP requests and a flux data flow pattern to allow for easy maintenance and debugging. Utilizing this architecture will allow our user interface to be attractive and intuitive. It will also allow our application to be efficient and reliable.

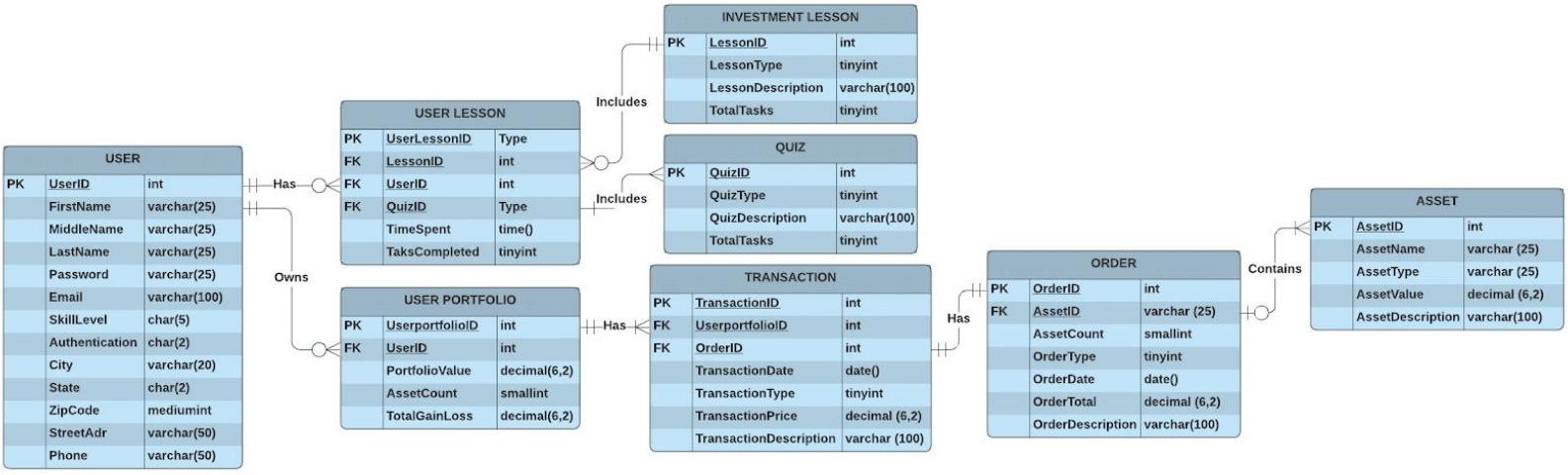
Data Storage System

We plan on utilizing cloud storage via Microsoft Azure. Utilizing cloud storage allows for a very scalable and user friendly data storage experience. In terms of managing the database we will use MySQL to create, update and delete data. Our API will consist of a combination of Java for server maintenance, Typescript for logic based evaluation, and Node to create routes for the client to hit and feed back information to the user.

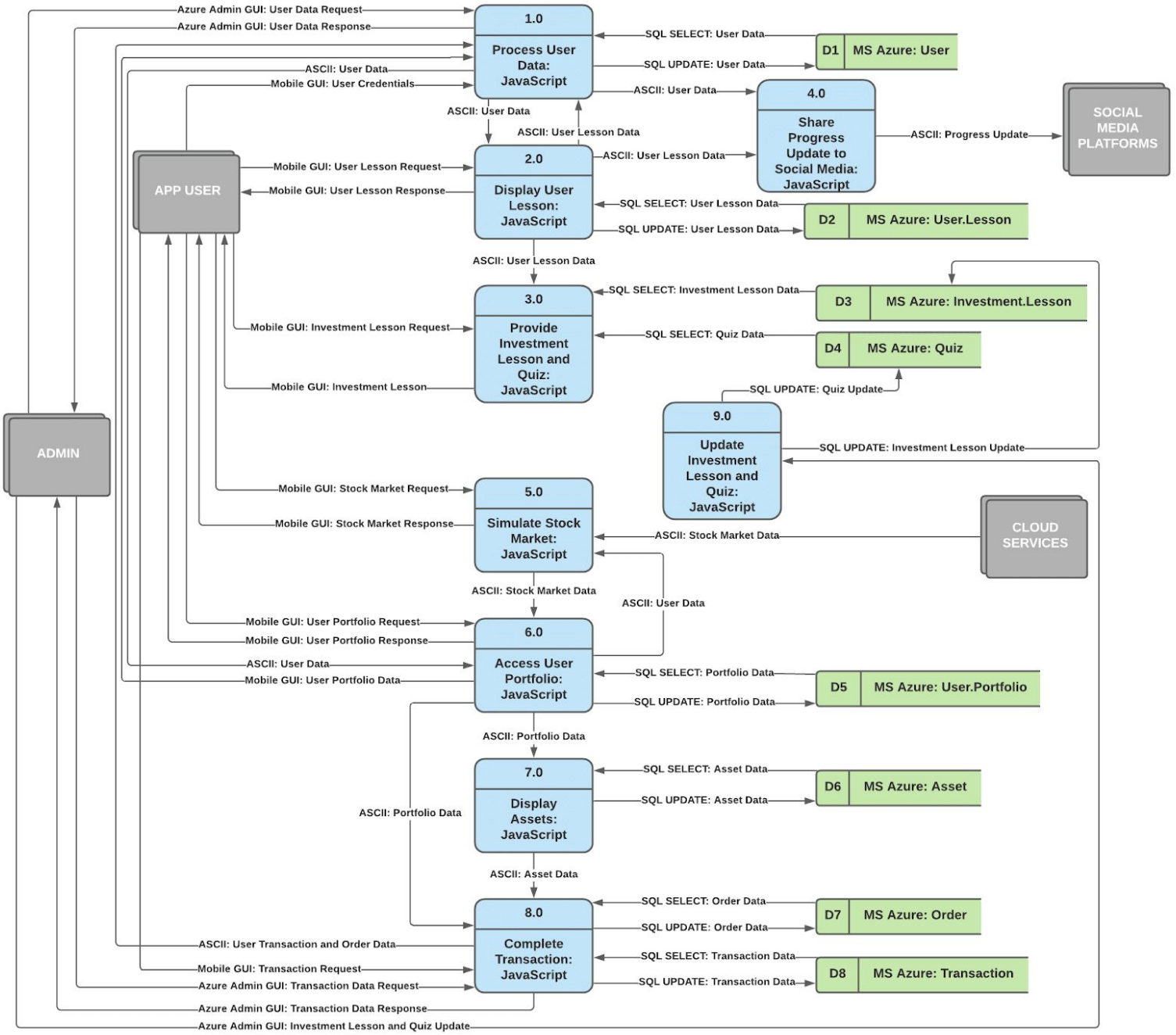
Any data that is created from the app will be stored on Microsoft Azure. Due to the uncertainty of required throughput for the application, don'twastetime will utilize Azure's tiered storage options to ensure efficiency between usage and cost. Further, with low latency and low average downtime, users can expect a real-time trading platform that will reflect their skills they have garnered from lessons.

Azure is also equipped with scalable storage sizes in order to match the requirements of their clients, as well as encryption technology that ensures security for any sensitive data a client might store. As one of the leaders of cloud storage technology, Azure is a reliable source to utilize for don'twastetime's database.

Physical ERD



Physical DFD

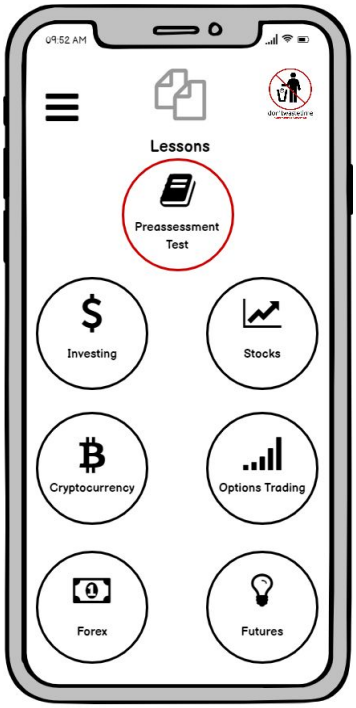


Mock User Interface

Homepage

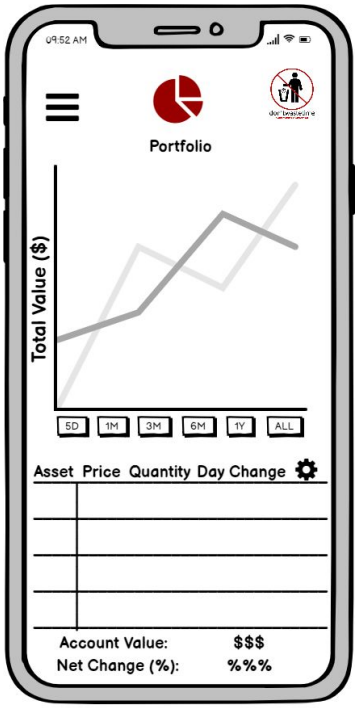
Lessons Dashboard

Menu



Lessons

Simulated Stock Market



Portfolio

References

MS Azure Information: https://azure.microsoft.com/en-us/services/storage/
Software Engineer Salary Information: https://www.indeed.com/career/software-engineer
Investment Manager Salary Information: https://www.indeed.com/career/investment-manager/salaries
Investment Analyst Salary Information: https://www.indeed.com/career/investment-analyst/salaries
Finance Professor Salary Information:
https://www.glassdoor.com/Salaries/finance-professor-salary-SRCH_KO0,17.htm
Executive Summary Information:

- https://www.bankrate.com/banking/savings/average-savings-interest-rates/#:~:text=The%20average%20interest%20rate%20for,account%20interest%20you'll%20earn
- https://www.bankrate.com/investing/millennials-slow-to-start-investing-in-stock-market-bankrate-survey-finds/#:~:text=In%20our%20survey%2C%20just%20one,18%20and%2025%20are%20investing
- https://www.businessinsider.com/personal-finance/average-stock-market-return#:~:text=Between%202010%20and%202020%2C%20however,in%20the%20past%2010%20years

Software Utilized

Balsamiq: UI Wireframing
Google Docs: Document Collaboration
Google Drive: Document Sharing
Google Sheets: Spreadsheet Collaboration
Lucidchart: Diagramming
Qualtrics: Survey Distribution and Management
Zoom: Collaboration