UNIVERSITEIT VAN PRETORIA
UNIVERSITY OF PRETORIA
YUNIBESITHI YA PRETORIA

Department of Computer Science
Faculty of Engineering, Built Environment & IT
University of Pretoria

TradeSim

Software Requirements Specification

AiPi

May 2022

Rachel Hamilton, u20453478

A. Introduction:

This documentation has all the necessary documentation from the AiPi team.

Team members:

* Michael Viljoen

* Rachel Hamilton

* Regan Shen

* Sean Nkosi

* Sibusiso Mngomezulu

C. Project:

TradeSim

1 Introduction

1.1. Purpose of product:

The purpose of the following documentation is to provide the software requirements (functional and non functional), acceptance criteria, constraints and an overview of the TradeSim system. Our role is to build a TradeSim system that will be responsible for containing all information for a user to customize their own ETF's and set their own rules. The user will be able to track how well their ETF is doing, and use this to decide where they want to invest their real money in the future.

1.2 Scope of product:

TradeSim users will be able to sign up and sign into their account and have access to their profile where they are able to customize ETF's and set their own rules. The goal is to create a system that will consist of a user's customized account, to help the user construct a portfolio of stocks, and define rules on how they should be traded to mimic an index tracking ETF by creating their own mini-ETF. The result will show the user how much capital they'll need to invest in the various stocks to make their own "mini-ETF" competitive against the established ETF on the market. The user is able to download these rules to a file, so they can use them for future reference when

investing with their real money. By using the rules the user has created, they'll be able to enhance the performance of their own mini ETF.

1.3 Acronyms, Abbreviations and Definitions:

- FR – Functional Requirements
- NFR – Non Functional Requirements
- API - Application Programming Interface.
- UI - User Interface
- C – Constraint
- ETF – Exchange-traded fund

1.4. Overview:

The document follows the following scheme:

- An overall description
- Specific Requirements
- Acceptance Criteria

2 Overall Description

2.1 Product Function:

- The TradeSim user logs in to his/her profile
- The TradeSim user customizes their own mini-ETF's by setting rules.
- The TradeSim user imports rules for their own mini-ETF's.
- The TradeSim user visualises the mini-ETF's performance over time.

• The TradeSim user selects ETF's and their mini-ETF's to compare.

• The TradeSim user visualises selected ETF's and the mini-ETF's over time.

• The TradeSim user selects the mini-ETF's to share and can download the rules as a file.

2.2 User Characteristics:

The TradeSim system is intended for the use of: Users who want to invest in an ETF through the stock exchange.

2.4 Constraints:

C1. The system must be maintained and managed by the AiPi team.

C2. All system implementations and documentation must be done by the AiPi team.

C3. The TradeSim system design must be the design the AiPi team follows throughout the project.

C4. The frontend engineers must use a python framework to complete this project.

C5. The backend engineers must use Python to complete this project.

C6. The API Engineer of AiPi must use Symfin or Finhub to complete this project.

C7. The Data Engineer of AiPi must use SQL to complete this project.

2.5 Assumption and Dependencies:

- Assumptions:
  - The user is someone who wants to invest in an ETF.
  - The user has an internet connection.
  - The user is using a PC that has a UI.

- Dependencies:
  - Time management - Our group is dependent on time management to be able to complete our project.
  - Lack of knowledge to create a solution for a requirement - Lack of knowledge to implement a feature has a big effect on our team completing this project.

3 Specific Requirements

3.1 Functional Requirements:

- FR.1. A user should be able to define the rules against which the system should trade (including the amount to be invested). Our team's rules consist of the following:
  - FR.1.1. The system must allow the user to request certain companies by name or ticker.
    - FR.1.1.1. The system must allow the user to set a percentage in their specified company they want to invest in.
  - FR.1.2. The system must allow the user to request a percentage in a specific sector.
    - FR.1.2.1. The system must allow the user to set a percentage in their specified sector they want to invest in.
  - FR.1.3. The system must allow the user to request percentage in a specific industry.
    - FR.1.3.1. The system must allow the user to select a specific percentage in their specified industry they want to invest in.
  - FR.1.4. The system must allow the user to request a minimum amount of companies they want to invest in.
  - FR.1.5. The system must allow the user to set a time period in which it must reconsider its stocks.

  - FR.1.6. The system must allow the user to define a percentage that a stock can drop before the ETF sells the stock automatically.
  - FR.1.7. The system must allow the user to set the market cap min and max values.
  - FR.1.8. The system must allow the user to set the earnings min and max value.
  - FR.1.9. The system must allow the user to set an amount to invest.
  - FR.1.10. The system must allow the user to reject specific companies by name or ticker.
  - FR.1.11. The system must allow the user to reject specific sectors by name or ticker.
  - FR.1.12. The system must allow the user to reject specific industries by name or ticker.
  - FR.1.13. The system must allow the user to invest in companies based in specific countries.
    * FR.1.13.1. The system must allow the user to set a percentage based on a country of a specific company.
  - FR.1.14. The system must allow the user to reject companies based in specific countries.
  - FR.1.15. The system must allow the user to set a minimum and maximum price for shares.
  - FR.1.16. The system must allow the user to set a balance period and/or balance threshold percentage).
- FR.2. The system must apply the rules against historic market data.
- FR.3. The system must compare the performance of the mini-ETF against other indexes in the market.
- FR.4. The system must allow the user to be able to save their rules and also test these different rules against each other.

3.2 External Interface Requirements:

3.2.1. User Interfaces

- The TradeSim system can only be accessed by a PC with a UI.

3.2.2. Hardware Interfaces

- Connection to the internet.

3.2.3. Software Interfaces

- Web browser.

3.3 Performance Requirements:

All the features of the TradeSim system must function as expected just as any trading system on a web app should.

3.4 Design Constraints:

The TradeSim system is dependent on the design requests from the project owners, and therefore we cannot go forward without communication with the project owners informing us about the design requirements needed.

3.5 Quality Requirements:

1. The basic design of the TradeSim system should have a look that aligns with the UI of a trading web app.

2. Data integrity - The TradeSim user must have access to certain information such as their rules created, that must remain private. This information cannot be leaked and should only be accessed by the TradeSim user.

3. Security - Certain information on the users TradeSim account that needs to remain private, for the user's eyes only, can not be accessed by anyone else unless they login to their TradeSim account.

4. Performance - All the features of the TradeSim system must function as expected according to the information and requirements given by the project

owners.

3.6 Architectural Requirements:

3.6.1 Flexibility:

- The TradeSim system must function on a device that has a UI.
- The TradeSim system must function using all web browsers.

3.6.2 Maintainability:

- There must be constant communication between developers of the AiPi team and the project owners in order to identify and fix errors to improve the quality of the feature.
- Clear documentation of requirements must be provided to ensure maintainability.

3.6.3 Security

- Certain information on the TradeSim user's account that needs to remain private, for the user's eyes only, can not be accessed by anyone else unless they login to the company representatives profile.

3.6.4 Availability:

- The TradeSim user's account must be available and easily accessed by any device, desktop or mobile with a UI.
- The TradeSim user's account must be available at all times.

3.6.5 Reliability:

- TradeSim is reliable on internet connection.
- The TradeSim system must ensure a stable experience to the user.

3.6.6 Usability:

- TradeSim must be mobile friendly.

4 Attributes

- Availability
  - The TradeSim system must be accessible to the TradeSim users at all needed times.
- Security
  - Private information belonging to the TradeSim user cannot be leaked, and must remain private.

5 Acceptance criteria

4.1 All information that the TradeSim user gives to the system must be in the database and displayed on the user interface without hiccups. (e.g. rules).

4.3 The system must allow the user to be able to import and set rules for their mini-ETF.

4.4 The system must allow the user to login to their account.

6 Appendices

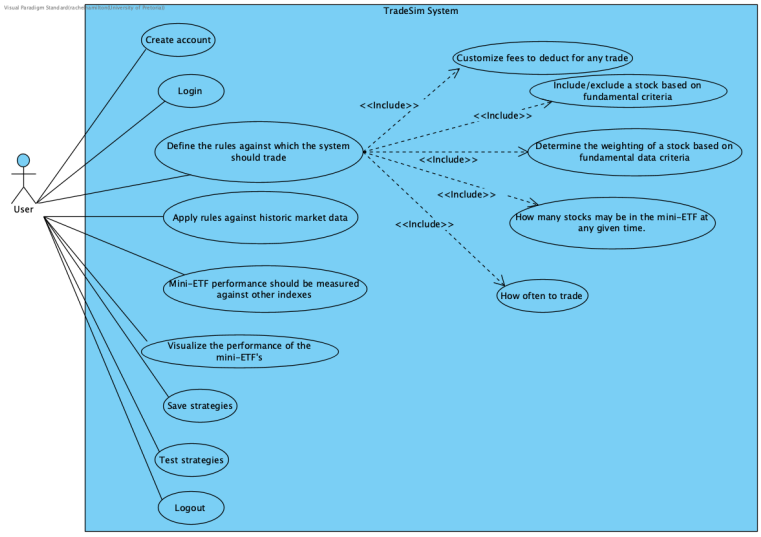

Figure 1 is a use-case diagram representing the TradeSim system.

7 Service Contracts

Service Contracts

Use Case ID	UC1
Use Case Name	Create account
Priority	High
Actors	TradeSim user
Precondition	The user is on the registration page.
Reason	The user wants to create an account with TradeSim
Main Scenario	1: The User inputs their details (Full Name, Email Address, Phone Number, Password, etc.) 2: The User presses the 'Register' button. 3: The system will validate the users' details. 4.1: If the users' details are invalid, prompt the user to try again. 4.2: If the users' details are valid, add them to the database and notify the user that the account has been successfully created. 5: The verification request will be added to the admins' 'verification requests'.
Alternate Scenario	None
Postcondition	The user will be redirected to the front page of the website, now logged into their account.

Use Case ID	UC2
Use Case Name	Login to TradeSim
Priority	High
Actors	TradeSim user
Precondition	The user must have an existing account with TradeSim
Reason	The user wants to login into their respective account.
Main Scenario	1: The user presses the 'Login' button. 2: The user enters their details. 3: The user presses the 'Submit' button. 4: The system validates the user's details, using the database. 5.1: The details are valid, the system notifies the user they have successfully logged in. 5.2: The details are incorrect, the system prompts the user to try again.
Alternate Scenario	None
Postcondition	The user must be now logged into their account and be redirected to their respective home page.

Use Case ID	UC3
Use Case Name	Define the rules against which the system should trade.
Priority	High
Actors	Tradesim user
Precondition	The user must be on their TradeSim account
Reason	The user wants to define rules against the system should trade.
Main Scenario	1: The user selects the rule from a dropdown menu which they want to apply to their mini-ETF. 2: The system prompts the user to confirm their decision. 3: The user confirms their decision, and the rules are applied to their mini-ETF.
Alternate Scenario	None
Postcondition	The user can examine their rules on their mini-ETF.

Use Case ID	UC4
Use Case Name	Apply rules against historic mock data.
Priority	High
Actors	Tradesim user
Precondition	The user must be on their TradeSim account and have chosen their rules.
Reason	The user wants to apply rules against historic mock data.
Main Scenario	1: Once the user has created their rules for their ETF, the user will be able to see the performance of the ETF from previous times. 2: They will be able to view what their ETF would be worth at a previous date 3: This will be visualised through a graph.
Alternate Scenario	None
Postcondition	The user can track the performance of their ETF over time.

Use Case ID	UC5
Use Case Name	Mini-ETF performance should be measured against other indexes.
Priority	High
Actors	Tradesim user
Precondition	The user must be on their TradeSim account and have applied their rules against historic mock data.
Reason	The system compares the performance of the mini-ETF's and the and other indexes.
Main Scenario	1: The user will be able to see an index on a market and see that index's performance compared to their own mini-ETF's performance when they view their own mini-ETF. 2: A separate page on the website will view this.
Alternate Scenario	None
Postcondition	The user will be able to see the performance of the ETF compared to other indexes on a page.

Use Case ID	UC6
Use Case Name	Visualise the performance of the mini-ETF's
Priority	High
Actors	Tradesim user
Precondition	The system must have measured the performance of the mini-ETF against other indexes.
Reason	The user wants to visualise the performance of the mini-ETF's to see how successful their ETF is doing using their rules.
Main Scenario	1: This will be displayed in the form of a graph. 2: Performance of the ETF will be visualised on this page.
Alternate Scenario	None
Postcondition	The user can save or test their strategy.

Use Case ID	UC7
Use Case Name	Save Strategies
Priority	High
Actors	Tradesim user
Precondition	The user must be on their TradeSim account and have chosen their rules.
Reason	The user wants to apply rules against historic mock data.
Main Scenario	1: Once the user has created their rules, their rules are saved on a database. 2: The strategies are saved automatically
Alternate Scenario	None
Postcondition	The rule is used on their mini-ETF

Use Case ID	UC8
Use Case Name	Log out of TradeSim
Priority	High
Actors	Tradesim user
Precondition	The user must be currently logged into their account.
Reason	The user wants to log out of their account.
Main Scenario	1: The user presses the 'Logout' button. 2: The system prompts the user to confirm their decision. 3.1: The user confirms their decision, and the user is logged out of their account. 3.2: The user denies their decision and is returned to the previous page.
Alternate Scenario	None
Postcondition	The user is redirected to the login page.