Software Requirements Specification

for

BASIS Automated Stock Investment Software

Version 1 approved

Prepared by Vinayak Mathur

and Anurag Upadhya

Case Western Reserve University

September 21th 2020

Table of Contents

Revision History

Name	Date	Reason For Changes	Version
Vinayak Mathur	09/10/20	first draft	1.0
Vinayak Mathur	09/15/20	second draft.	1.1
Anurag Upadhya	09/16/20	third draft. Changes to System Features (3.1) and External Interface Requirements (4)	1.2
Anurag Upadhya	09/17/20	Added System Features (3.2, 3.3)	1.3
Anurag Upadhya	09/20/20	Added Inspection Report and Non Functional Requirement. Added System Features (3.4-3.7)	1.4

1.　Introduction

1.1　Purpose

This SRS describes the software functional and nonfunctional requirements for release 1.0 of the BASIS Automated Stock Investment Software. This document is intended to be used by the members of the BASIS project team that will implement and verify the correct functioning of the system. Unless otherwise noted, all requirements specified here are high priority and committed for release 1.

1.2　Project Scope

BASIS is an application that will provide novice to intermediate investors (users) stock market equity portfolio recommendations based on the answers provided by them to several test questions. This will allow users to select a portfolio that aligns with their risk-tolerance without going through the hassle of researching investments or avoiding the monetary cost of using a wealth management firm. A more detailed outlook of the Project Scope is provided under *4. Scope and Limitations.*

1.3　References

1. "U.S. Census Bureau QuickFacts: United States." Census Bureau QuickFacts, 2019, www.census.gov/quickfacts/fact/table/US/HSD410218.
2. Hayes, Adam. "Retail Investor Definition." *Investopedia*, Investopedia, 28 Aug. 2020, www.investopedia.com/terms/r/retailinvestor.asp.
3. "Modern Portfolio Theory." *Modern Portfolio Theory | TD Ameritrade*, www.tdameritrade.com/education/articles/modern-portfolio-theory.page.

2.　Business Requirements

2.1　Background, Business Opportunity, and Customer Needs

According to the Pew Research Center, over 50% of American households are in some way or another invested in the US equity stock market and over 50 million are individual retail investors. A majority of these investors opt to make their own investment decisions and sometimes allocate their portfolio without hedging for risk. A most recent instance is the COVID19 market crash which saw a loss of 33% on the S&P 500. An uninformed retail investor would have lost a significant portion of their investments. A regular retail investor does not have the tools required to make as informed investments as a corporate fund without spending a significant amount on the too;s required. Apart from the reasons stated above, most people working full time jobs cannot devote much time to financial analysis, research and allocation of their investments. BASIS will make things much easier for investors and provide them with viable portfolios.

2.2   Business Objectives and Success Criteria

BO-1: BASIS should make it easy for a novice / new user to understand the control questions.
BO-2: BASIS should reduce time spent by investors on financial analysis, research and allocation.
BO-3: Use more detailed questions for more experienced users to narrow down on the right portfolio.

SC-1: Retain users for at least a 4-month period or the maximum time period on the recommended portfolio, whichever comes last.
SC-2: Reduce portfolio risk by a minimum of 5% through Markowitz allocation.

2.3   Business Risks

RI-1: Users may not trust portfolio allocations if they are given abnormal weightages.
RI-2: Users might not trust stock selections if the portfolio includes underperforming stocks.
RI-3: Users might not be comfortable / willing to provide their current portfolio stock holdings.

3.   Vision of the Solution

3.1   Vision Statement

BASIS Automated Stock Investment Software seeks to allow retail investors to be at par with large corporate investment funds in minimizing risk and maximizing return on investment by utilizing principles of Modern Portfolio Theory and Markowitz Mean Variance. This will be achieved by asking the user a set of control questions. Each subsequent question will narrow down the stocks for the portfolio. A set of multiple portfolios will then be presented to the user for them to select. These portfolios will be optimized to suit the user's criteria and will minimize risk while maximizing return on investment.

3.2   Major Features

FE-1: Make a completely new portfolio from scratch for a novice investor based on the answers to questions.
FE-2: Modify a portfolio based on the list of stocks already owned by an investor and the answers to the questions.
FE-3: Allocate a portfolio based on the list of stocks already owned by an investor without modification.

3.3   Assumptions and Dependencies

AS-1: The user can legally invest in United States equity stock markets (NYSE, NASDAQ and NYSEARCA).
DE-1: The user has access to a device that can connect to the internet. This device may be mobile (Android,iPhone) or a PC (Mac, Desktop, Laptops).

4. Scope and Limitations

4.1 Scope of Initial and Subsequent Releases

Feature	Release 1	Release 2	Release 3
FE-1	Stocks will be selected from a list of 50 predetermined stocks.	Stocks will be selected from the stocks in the S&P 500.	Stocks will be selected from a dynamic list of stocks including the S&P 500, ETFs, and more.
FE-2	Not implemented.	The stocks in the list must be from a list of stocks in the S&P 500.	The stocks in the list must be available for trade on the NYSE, NASDAQ or NYSEARCA.
FE-3	The stocks in the list must be from a list of 50 predetermined stocks.	The stocks in the list must be from a list of stocks in the S&P 500.	The stocks in the list must be available for trade on the NYSE, NASDAQ or NYSEARCA.

4.2 Limitations and Exclusions

LE-1: Stocks may sometimes be frozen for trading. This will occur during a circuit breaker. Since the system is not privy to this information, it will treat it as if still tradeable.
EX-1: OTC stocks and private shares will not be used for allocation.

5.   Business Context

5.1   Project Priorities

Dimension	Driver	Constraint	Degree of Freedom
Schedule	Project Milestones to be met.		Release 1 (demo) planned to be available by 10/19/2020, release 2 (demo) by 11/02/2020; and release 3 (final) by 12/04/2020
Features		All features scheduled for release 1 must be fully operational	High priority features to be implemented by release 1, Medium priority features to be implemented by release 2 and Low priority features to be implemented by release 3.
Staff	Team consists of one leader (Vinyak Mathur) and one member (Anurag Upadhya)		

6. Overall Description

6.1 Product Perspective

BASIS is a new standalone, self-contained application with the following . Figure 1 shows how each of the individual components interact with each other.
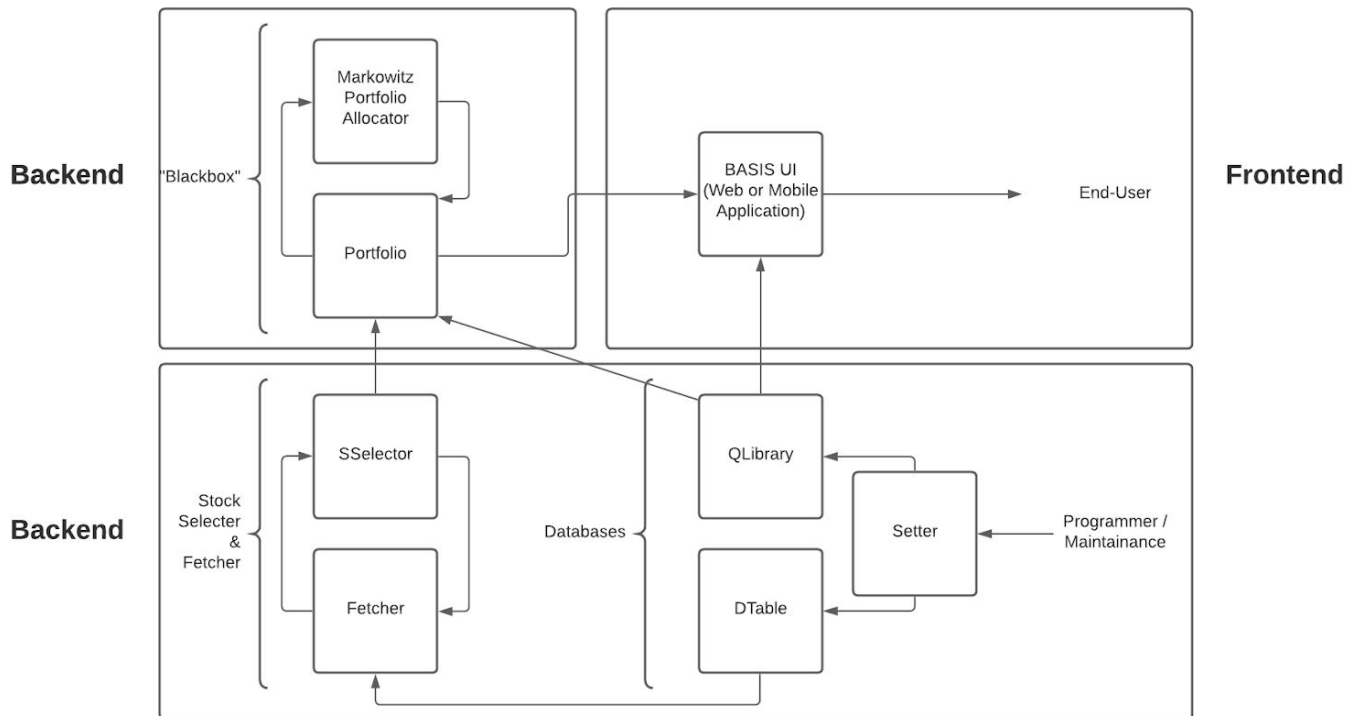
6.2 User Cases and Characteristics

Novice (favored): A Novice is a first-time or beginner level equity stock market investor. A Novice would be a person with little to no prior experience with investment research, financial analysis and portfolio allocation. Novices will be given less specific and easy to understand, non-technical questions from QLibrary. Novices have little to medium risk-tolerance, looking for both stock picks and portfolio allocation for medium to long term investments and can be given the choice of value investing or growth investing. A Novice will be shown one portfolio form the many that BASIS will compile.

Small Scale Fund Manager: A small scale fund manager is a person responsible for the wealth management of several clients. A fund manager will have low risk tolerance and will primarily be interested in the Markowitz Portfolio Allocator of BASIS. They would have already conducted research

and analysis of the stocks they have chosen. The time-period of investment will also depend on the type of fund. BASIS may be a secondary tool for them for portfolio allocation.

High Volume Trader: A high volume trader is a person with intermediate to advanced stock trading knowledge and experience and whose source of income is dependent on investing. High volume traders will have the widest risk tolerance and shortest time-period for investments. Like the fund manager, a high volume trader will also be looking primarily for portfolio allocation to minimize risk.

Experienced Investor: An experienced investor is a person who has prior experience investing in the equity stock market. An experienced investor will hold positions longer than any other user type and has a higher risk tolerance compared to others. They will be interested in growth investing and picking stocks. Portfolio allocation and risk hedging is not as high a priority for this investor as consistent growth is.

6.3   Operating Environment

OE-1: BASIS will operate on any HTML5, CSS3 and Javascript supported browser including but not limited to Google Chrome (ver. 66+), Opera (ver. 45+), Safari (ver. 11.1+), Microsoft Edge (ver. 15+) and Mozilla Firefox (ver. 55+).

6.4   Design and Implementation Constraints

CO-1: All HTML code will conform to HTML5 standards.
CO-2: All CSS code will conform to CSS3 standards.
CO-3: All scripts must be coded in Javascript.
CO-4: Finhub.io API calls are limited to 60/min and 30/s

7. System Features

7.1 QLibrary

7.1.1 Description and Priority

A set of standard questions chosen from this library will be asked to the user. Answers to questions will initiate a response asking for answers for further questions.

Priority: 9 (High) as it forms the basis for our recommendation.

7.1.2 Stimulus/Response Sequences

Stimulus: User says they are interested in either value or growth investing.
Response: System queries set of questions and asks the user a question.
Stimulus: User gives industries of interest.
Response: System notes down answers and asks further questions.
Stimulus: User says they are done answering questions.
Response: System uses answers given by the user to provide a list of recommended
         portfolios.

7.1.3 Functional Requirements

Basis.QLibrary: This system is a component of the database and shall store the questions that need to be asked to the user.
Basis.QLibrary.add: This system is used to add a question and data range to the question library. Does not replace the question if it already is present in the library.
Basis.QLibrary.replace: This system is similar to Basis.QLibrary.add but will replace the question and data range if they exist in the library.
Basis.QLibrary.find: If the given question exists in the library then this system will respond with the respective data range.
Basis.QLibrary.display: displays all the questions along with their respective data ranges.
Basis.QLibrary.get: This system responds by sending the entire data range.
Basis.QLibrary.remove: This system removes the question and its respective data range.
Basis.QLibrary.exists: This system checks if the given question exists in the library.
Basis.QLibrary.clear: This system clears (removes all) the questions and data ranges from the library.

7.2   DTable

7.2.1   Description and Priority

The user after answering questions will get recommendations from this feature. This feature contains historical information about a large number of stocks.

Priority: 9 (Medium)

7.2.2   Stimulus/Response Sequences

Stimulus: User finishes answering questions
Response: Database is queried to select stocks with best risk/reward ratios
Stimulus: User requests the removal of certain stocks
Response: System removes the stocks from the list
Stimulus: User approves list of stocks
Response: System saves the list for the user

7.2.3   Functional Requirements

Basis.DTable: This system is a component of the database and shall store the list of stock tickers and information like historical data, current market price, company name, company description, link, 52H, 52L, etc.
Basis.DTable.get: This system responds by sending information about the ticker with the given criteria.
Basis.DTable.add: This system adds the given ticker to the list of tickers.

7.3   Setter

7.3.1   Description and Priority

The programmer may add, remove, or change questions and data from the QLibrary and DTable features.
Priority: 9 (High) since the nature of questions and number of stocks provide better quality and quantity for investments.

7.3.2   Stimulus/Response Sequences

Stimulus: Users report that portfolios are not to their liking.
Response: System adds additional questions provided by the programmer.
Stimulus: Answers provided by users to certain questions does not give the desired portfolio.
Response: System removes questions from QLibrary selected by the programmer.
Stimulus: Stock in list has an increase in unexpected risk.
Response: Programmer removes stock from DTable.

7.3.3   Functional Requirements

Basis.Setter: This system shall interact with DTable and QLibrary to add, remove or modify the data in them.
Basis.Setter.addQLibrary: This system shall add the given question and data range to Qlibrary.
Basis.Setter.addDTable: This system shall add the given ticker to DTable.
Basis.Setter.removeQLibrary: This system shall remove the given question and data range from the QLibrary,
Basis.Setter.removeDTable: This system shall remove the given ticker from the DTable.
Basis.Setter.clearQLibrary: This system shall clear contents of the QLibrary.

7.4   Fetcher

7.4.1   Description and Priority

SSelector may fetch data from the list of stocks and add it to the portfolio.

Priority: 9 (High)

7.4.2   Stimulus/Response Sequences

Stimulus: User sends criteria.
Response: Fetcher goes through DTable and finds those that match the given criteria.

7.4.3   Functional Requirements

Basis.Fetcher: retrieves individual stocks from DTable and sends them to SSelector.
Basis.Fetcher.screen: screens and finds the stocks in the DTable that match the given criteria.

7.5   SSelector

7.5.1   Description and Priority

System SSelector will select stocks from the data based on the responses to the questions form QLibrary.

Priority: 9 (High)

7.5.2   Stimulus/Response Sequences

Stimulus: User indicates an interest in technology stocks.
Response: System selects stocks from the technology sector.
Stimulus: User indicates an interest in dividends.
Response: System selects stocks which give dividends from the list

7.5.3   Functional Requirements

Basis.SSelector: System shall select stocks from the data based on the responses to the questions form QLibrary.

7.6   Markowitz Portfolio Allocator

7.6.1   Description and Priority

System will allocate the selected stocks and form a portfolio with the selected stocks by utilizing Modern Portfolio Theory and Markowitz Mean Variance.

Priority: 9 (High)

7.6.2   Stimulus/Response Sequences

Stimulus: Portfolio System finishes selecting stocks based on users' questions
Response: System uses Markowitz Mean Variance and the Modern Portfolio Theory to form a portfolio.

7.6.3   Functional Requirements

Basis.Allocator: This system shall utilize Markowitz Mean Variance and the Modern Portfolio Theory for portfolio allocation to minimize risk.

7.7   Portfolio

7.7.1   Description and Priority

The users shall receive recommendation in the form of a portfolio of stocks based on their answers to the questions given by the system.
Priority: 6 (Medium)

7.7.2   Stimulus/Response Sequences

Stimulus: User asks for stocks with least risk.
Response: The system sends the portfolio of least risky stocks.
Stimulus: User asks for growth stocks.

Response: The system sends the user a portfolio containing stocks with high growth
potential.
Stimulus: User asks for value stocks.
Response: The system sends the user a portfolio of stocks with high value according to the
value metrics.

7.7.3   Functional Requirements

Basis.Portfolio: This system shall store portfolios and provide the user with the portfolio of their choice.
Basis.Portfolio.get: This system shall send the entire set of portfolios to the UI.
Basis.Portfolio.risk: This system shall send the portfolio with the least risk to the UI.
Basis.Portfolio.growth: This system shall send the portfolio with the highest growth to the UI.
Basis.Portfolio.value: This system shall send the portfolio with the best value to the UI.

8. External Interface Requirements

8.1 User Interfaces
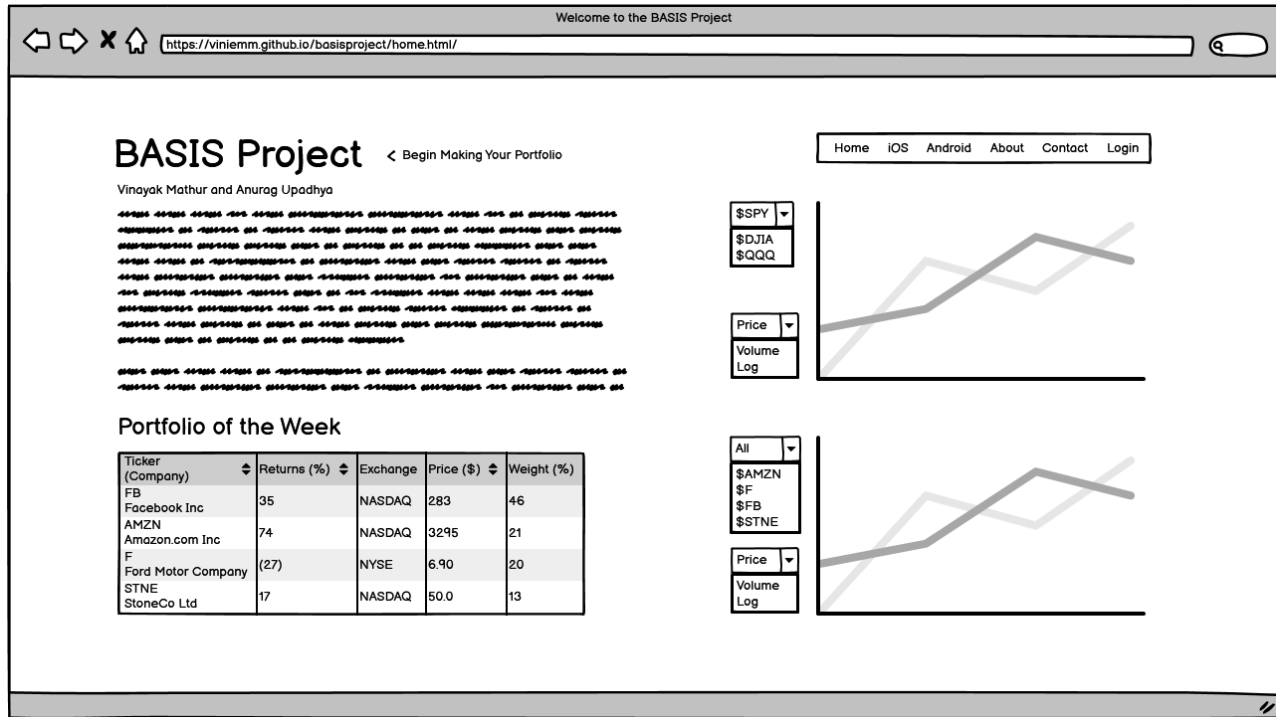
(Figure 2.1 shows a wireframe UI of how BASIS will look to a user on a desktop browser.)

UI-1: The system shall have instructions on each of the HTML pages which will explain how users should interact with it.
UI-2: The user shall be able to navigate the web pages through the mouse and keyboard.

(Figure 2.2 shows a wireframe UI of how BASIS will look to a user on a mobile browser.)
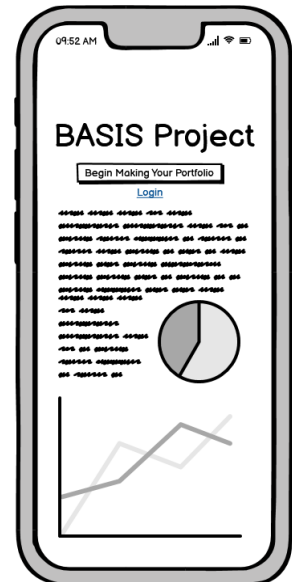
8.2 Hardware Interfaces

No hardware interfaces have been identified.

8.3 Software Interfaces

SI-1: Stock Selector
SI-1.1: Stock Selector shall import the data from the csv file of the SSelector and Fetcher through a programmable interface.
SI-1.2: When the user adds the stock to the portfolio list, the Stock Selector shall remove the stock from the list.

8.4   Communications Interfaces

CI-1: The system shall send a message (prompt) to the user confirming a valid answer to the
question.
CI-2: The system shall send a message to the user with a list of portfolios.

9.   Other Nonfunctional Requirements

9.1   Performance Requirements

PE-1: The API calls shall be limited to 60 per minute and 30 calls per second.
PE-3: Confirmation messages shall be displayed to the user within 5 seconds of the
submission of information.
PE-2: Responses to users' queries shall be displayed within 1 minute.

9.2   Safety Requirements

No safety requirements have been identified.

9.3   Security Requirements

No security requirements have been identified.

9.4   Software Quality Attributes

Availability-1: BASIS Automated Stock Investment Software shall be available to users on
the internet between 4:00am Eastern Time and 8:00pm Eastern Time.

Appendix A: Inspection Report

Date	Comments	Status	Inspector
9/18/2020	Add System Features Fetcher, SSlector and Markowitz Portfolio Allocator.	Completed by Vinayak Mathur	Anurag Upadhya
9/20/2020	Add Communication Interfaces, User Interfaces and Communication Interfaces.	Completed by Vinayak Mathur	Anurag Upadhya
9/21/2020	Add System Requirements to System Feature DTable	Completed by Vinayak Mathur	Anurag Upadhya
9/21/2020	Add Functional Requirements to System Feature SSelector	Completed by Vinayak Mathur	Vinayak Mathur