|  |
| --- |
| CMSC 495 |
| Stock Back-tester |
| Project Analysis |
|  |
| **Nancy Isaac** |
| **6/28/2015** |

|  |
| --- |
| [Type the abstract of the document here. The abstract is typically a short summary of the contents of the document. Type the abstract of the document here. The abstract is typically a short summary of the contents of the document.] |

Contents

[Project Analysis 2](#_Toc423249509)

[Project Description 2](#_Toc423249510)

[Project Members 2](#_Toc423249511)

[Requirements Gathering/Elicitation 2](#_Toc423249512)

[Requirements Terminology 2](#_Toc423249513)

[Functional Requirements 2](#_Toc423249514)

[Non-Functional Requirements 3](#_Toc423249515)

[Analysis 4](#_Toc423249516)

# Project Analysis

## Project Description

Many organizations systematically trade U.S. equity (common stock) with daily, weekly, and monthly horizons. The trading of stock with a high level of turnover is a costly endeavor due to transaction costs, taxes, etc., and therefore, those that partake in such activity typically perform extensive research on the trading system. Trading commences only after it has been determined, with high statistical confidence, that the rules that define the system are profitable.

This program will read a series of historical equity price and volume data and then test if any pre-programed rules (indicators), or combination of indicators, predicts future returns with a high statistical significance.

## Project Members

* Baker, Pierce
* Casey, Garrett
* Isaac, Nancy
* Kelley, Justin
* Owens, Chris

## Requirements Gathering/Elicitation

This is a self-initiated project. There are many articles available online that describes stock back-testing strategies and we use those to understand the financial aspects involved in this project. We are also using guidelines from other back-testing software available online. We compared the tools available to determine what the most useful fields for inputs and outputs are as well as what can be done in the time frame for the class. The following website has been used as a guideline: <http://finviz.com/screener.ashx?v=111&ft=3>. One of the students in our group works in the financial industry and the project is based on his proposal.

## Requirements Terminology

* Any requirement listed below with a “must” is necessarily in the implementation
* Any requirement listed below with a “should” is a nice to have or can be substituted with another form.

## Functional Requirements

1. The stock back-tester shall use real world data.

We will be using yahoo finance to gather the historical information on a particular ticker symbol. The data that we collect is based on the indicator criteria being triggered. A tester of this program can collect the data from yahoo finance and compare the data to what we collect for this program.

1. The stock back-tester must use well known stock trading indicators.

There are well known indicators that are used in the financial industry. Yahoo finance will give us the OHLCV (open, high, low, close, volume) values. These values can be used in combination with the indicators can be used for back testing.

1. Indicators can be combined for the query.
2. The program must provide a command line interface to the user
3. A configuration file must be used to gather user inputs
4. The following fields must be defined in the configuration file: PORTFOLIO.tickers, STRATEGY.name, and STRATEGY.indicators. All other variables will have default values.
5. User must be able to provide a start date and an end date
6. The user must be able to provide a stock symbol
7. The user must be able to provide a mark-out period of 5, 10 and 20 days
8. All inputs must be logged to STDOUT and to a log file
9. User should have the ability to silence the output to STDOUT
10. The program must output a summary of statistics that can easily be read by humans and machines.
11. The program must output a detailed report of the statistics
12. The program should provide a nice GUI interface to parse the detailed report and provide graphs of the results to the user.
13. The detailed statistics must include forward returns
14. The forward returns should be calculated using the standard formula: log return = ln(end) - ln(start) where ln is the natural log function. *See*  *(Why Log Returns) for info*
15. The detailed statistics should include count, mean and standard deviation
16. The real world data must include the corporate-action adjusted prices.
17. The data source must provide the ability to generate trade-date calendar.
18. The program that evaluates the indicator must return a Boolean value.
19. The program that handles the data must validate the boundaries of the sample period, mark-out periods and maximum indicator look-back period
20. The program responsible for testing the indicators against the historical data must be able to ignore events in overlapping mark-out periods to ensure independent events.

## Non-Functional Requirements

1. The program should be light-weight
2. The program must be able to run on any OS that has python installed.
3. The program must be able to easily reproduce the back-tests with the config files.
4. The program should support multiple independent instantiations of the applications without causing errors.

## Analysis

The requirements above have been considered to provide an outline of the design. Our program will be done in the following hierarchy. The dotted lines indicate either writes or reads. The solid lines indicate sequence of execution.

BackTester

Indicator Library

Data Manager Library

TradeManager

AnalysisManager

ReportManager

The requirements generate the above the code flow and can generate a back-tester program that can peak into the history and generate forward return data. The requirements were sufficient to generate the design needed for this program.