****

**TECHNICAL ANALYSIS SCREENER\_1 MOVING AVERAGES**

Software Requirement Specification (SRS) Document

Sprint Implementation

Project Timeline: 06.11.2022 to 12.11.2022

**ABSTRACT**

A general issue with moving average trading is the assumption that all buy/sell signals result in a trading action. The argument that such trading rules are representative of trading practice is highly questionable. This thesis proposes two new moving average trading rules designed to capture trading practice. The first trading rule is the trade reduction rule and is based on the idea of allowing a trade to run. The second trading rule is the positive autocorrelation rule and is based on the idea of only trading if it is believed to be profitable to do so. The trading rules are tied to moving average trading via the buy/sell signal generating mechanism and alter the way the price crossover rule responds to the buy/sell signals. Simulations of portfolios of UK equities find that the trading rules uncover information that is missed by the price crossover rule and there is evidence that this information is financially exploitable. This motivates the argument that the information needed for trading to be economically viable is observable in the price. The trading rules also establish a link with the market microstructure literature. The trading rules uncover issues of informed trading (asymmetric information), liquidity, adverse selection and price impact. The strongest interpretation that can be applied to the trading rules in this context is that they are examples of informed trading. Compared to the price crossover rule, the trading rules are better able to extract meaning from or are better able to understand the same price information.

**INDEX**

1.Introduction

1.1 Background  
 1.2 Purpose  
 1.3 Key Project Objectives  
 1.4 Functional Overview  
 1.4.1 Header Files

2. Design Overview  
  
3. Moving Averages

3.1 What is a Moving Average (MA)

3.2 TYPES OF MOVING AVERAGES  
 3.2.1 Simple Moving Average  
 3.2.2 Exponential Moving Average

4. Detailed System Design  
 4.1 Data Flow Diagram  
 4.2 Data Overview  
  
5. Environment Description  
 5.1 Time Zone Support  
 5.2 Language Support  
 5.3 User Desktop Requirements  
 5.3.1 Integration Requirements  
 5.3.2 Network  
 5.3.3 Configuration  
  
6. Conclusion

7. References

**1.INTRODUCTION**

A stock screener is a set of tools that allow investors to quickly sort through the myriad of available stocks and increasing Exchange-traded funds according to the investor’s own criteria. Stock screeners are most typically available on brokerage trading platforms (usually free), but there are also some independent subscription-based stock screeners available. Stock screeners allow investors to employ their own methodology about what makes a stock or ETF valuable (longer-term traders) or spot a potential trading opportunity (shorter-term traders).

**1.1BACKGROUND**

Technical analysis is an approach to predicting future price movements based on identifying patterns in prices, volume and other market statistics. Technical analysis usually proceeds by recording market activity in graphical form and then deducing the probable future trend from the pictured history. The premise is that prices exhibit various geometric regularities, which, once identified, inform the trader what is likely to happen next. This in turn allows the trader to run a profitable trading strategy. Technical analysis is prevalent in financial markets and is readily accessible in practitioner texts such as pring (2002), in the form of tools provided by online brokers such as Barclays Stockbrokers (www.stockbrokers.barclays.co.uk) as well as in the form of commentary in the financial and investment press.

**1.2 PURPOSE**

Technical stock screeners allow you to filter stocks according to many of the same price-dependent technical indicators that you would use on a stock chart. So, technical screeners can be extremely valuable for traders who normally trade momentum, reversal, or other short- and medium-term strategies.

**1.3 KEY PROJECT OBJECTIVES**

* To supply capital - To achieve this task, ownership in a private corporation is sold to the public in the form of shares of stock. Funds received from the sale of stock contribute to the firm’s capital formation.
* Toinspiresavings - This inspires people to save their income by making a profit. Continuous purchase and sale of securities on a stock exchange lead to the evaluation of their prices.
* Todevelop economy - It helps economic development by supplying the capital to the industries.
* To protect fraudulently**-**It is also to ensure that no fraudulence occurs in a transaction.
* To do long-term financing - Commercial banks generally disburse the short-term loan. So, supplying long-term finance is an objective of the stock exchange.

**1.4 FUNCTIONAL OVERVIEW**

**1.4.1 HEADER FILES**

* stdio.h
* stdlib.h
* string.h

**2.DESIGN OVERVIEW**

|  |  |
| --- | --- |
| Name of the Module | Extraction of file and add stock to file |
| Handled by |  |
| Description | Declared and created files in the valid format and extension and researched about create and read operations |

|  |  |
| --- | --- |
| Name of the Module | Remove and update stock to file |
| Handled by | Abhinaya Vaitla |
| Description | Developed code on update and delete operations and implemented error detection and exceptional handling |

|  |  |
| --- | --- |
| Name of the Module | Addition and modification functionality of stocks |
| Handled by | Oleti Lalita Sowmya |
| Description | Researched and developed all the conditions and functions on update and delete operations |

|  |  |
| --- | --- |
| Name of the Module | Deletion functionality of stocks and error detection |
| Handled by | Mansi Singh |
| Description | Developed code in creating sales record and designed dataflow diagrams and flow charts |

|  |  |
| --- | --- |
| Name of the Module | Config.ini or settings.ini file maintenance |
| Handled by | Abhinaya Vaitla |
| Description | Developed code on update and delete operations and implemented error detection and exceptional handling |

|  |  |
| --- | --- |
| Name of the Module | Design of algorithm |
| Handled by | Abhinaya Vaitla |
| Description | Developed code on update and delete operations and implemented error detection and exceptional handling |

|  |  |
| --- | --- |
| Name of the Module | Implementation of algorithm for buy or sell recommendation |
| Handled by | Abhinaya Vaitla |
| Description | Developed code on update and delete operations and implemented error detection and exceptional handling |

**3.MOVING AVERAGES**

## 3.1What Is a Moving Average (MA)?

A moving average (MA) is a stock indicator commonly used in technical analysis. The reason for calculating the moving average of a stock is to help smooth out the price data by creating a constantly updated average price

By calculating the moving average, the impacts of random, short-term fluctuations on the price of a stock over a specified time frame are mitigated.

**3.2 TYPES OF MOVING AVERAGES**

### **3.2.1 SIMPLE MOVING AVERAGE**

A Simple Moving Average (SMA), is calculated by taking the arithmetic mean of a given set of values over a specified period. A set of numbers, or prices of stocks, are added together and then divided by the number of prices in the set. The formula for calculating the simple moving average of a security is as follows:

​*SMA*=nA1​+*A*2​+…+*An* **/**n

where:

A=Average in period n

n=Number of time periods​

Charting stock prices over 50 days using a simple moving average may look like this:

Chart, histogram

Description automatically generated

**3.2.2 EXPONENTIAL MOVING AVERAGE** The exponential moving average gives more weight to recent prices in an attempt to make them more responsive to new information. To calculate an SMA the simple moving average (SMA) over a particular period is calculated first.

EMAt​=[*Vt*​×(s /1+d)]+EMAy​×[1−(s /1+d​)]

where:

EMAt​=EMA today

Vt​=Value today

EMAy​=EMA yesterday

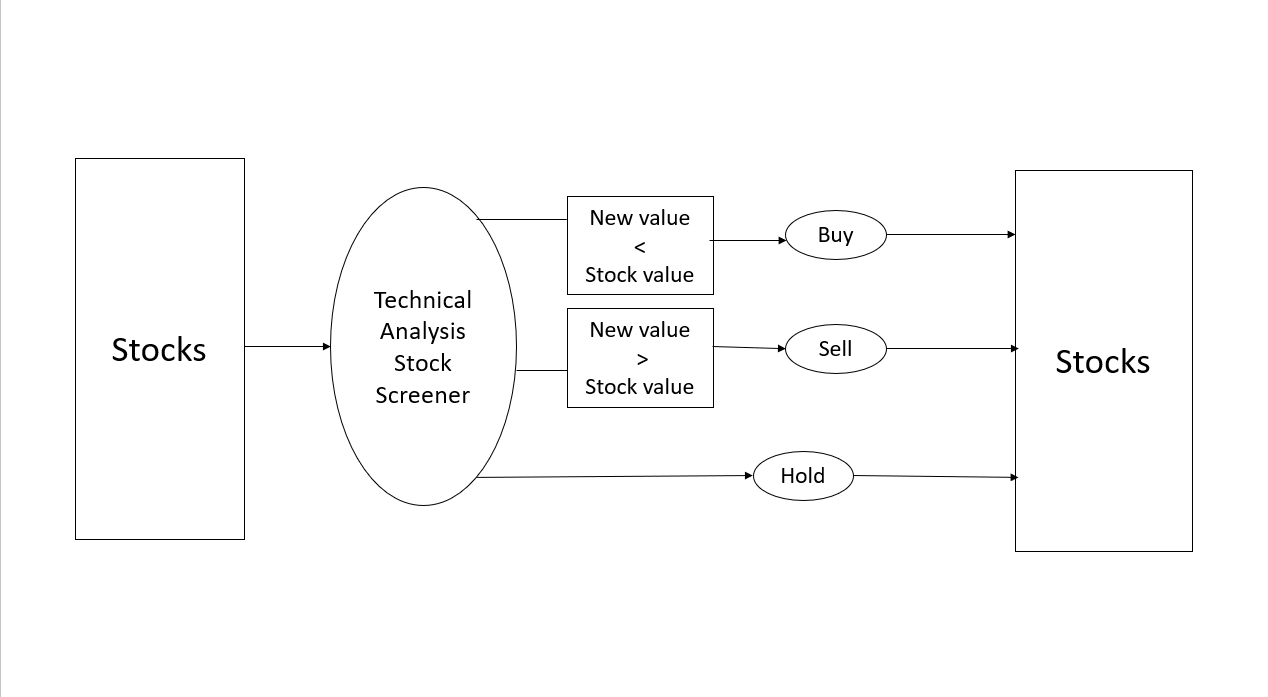
s=Smoothing

d=Number of days​

**4.DETAILED SYSTEM DESIGN**

**4.1 DATA FLOW DIAGRAM**

****

****

**4.2 DATA OVERVIEW**

****

​​

**5.ENVIRONMENT DESCRIPTION**

**5.1Time Zone Support:** IST- Kolkata

**5.2Language Support:** English

**5.3User Desktop Requirements**

* + 64-bit processor, 1.50 GHz or faster
  + At least 10 GB free hard drive space
  + At least 1 GB RAM Server

**5.3.1** **Integration Requirements**

* + - Language: C
    - Tools: Valgrind, ctags
    - Complier: gcc
    - Linux Environment

**5.3.2Network:** End to End

**5.3.3 Configuration:**

Operating System: Linux environment

**6.CONCLUSION**

Moving averages estimates the technical analysis used to help smooth out price data by creating a constantly updated average price. A rising moving average indicates the security is an uptrend, while a declining moving average indicates a downtrend. Based upon the averages in uptrend is recommended and the averages in the downtrend is not recommended.

**7.REFERENCES**

The references are:

<https://corporatefinanceinstitute.com/resources/wealth-management/stock-screener/>

<https://www.investopedia.com/terms/m/movingaverage.asp>

<https://www.nirmalbang.com/knowledge-center/use-moving-average-for-buying-stocks.html>

<https://www.nseindia.com/market-data/live-equity-market>

<https://www.researchgate.net/publication/251335107_Market_Timing_With_Moving_Averages>