**TA Screener 2**

**Heikin Ashi Candle Stick Pattern on Changing Colours**

High Level Design

# **Document Control :**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Heikin Ashi Candle Sticks Pattern On Changing Colours | | |  | | | | | |
| Guided by-  Rahul Tarkunde |  |  |  |  |  |  |  |  |
| Date | Version | Author | Brief Description of Changes | | | | Approver Signature | |
| December 07,2022 | 0.1 | Deshashri Kakade,  Vijaya Gotarane,  Saurav Kumar,  Meghana Birajdar,  Samiksha Dubal |  | | | |  | |
|  |  |  |  | | | |  | |
|  |  |  |  | | | |  | |

**Introduction**

1. **Purpose**

The Purpose of this project is to implement a stock analysis application using heikin

ashi formula that will help user to select stocks by studying past and present data.

The technical analysis method involves examining data generated through market activities, such as volume and prices. Technical Analysis helps identify trading opportunities using actions of Market Participants through charts and it helps traders to pick good stocks to trade.

In the technical analysis we use a Green and Red colours of stock charts to analyze market data, with the help of colours user can easily buy or sell their stocks.

## **1.1 Intended Audience:**

This document is intended to be read by User.

## **1.2 Acronyms/Abbreviations:**

|  |  |
| --- | --- |
| CLIENT | USER |
|  | TO GIVE A CLEAR IDEA TO THE USER RELATED STOCK, WHICH STOCK WILL BE PROFITABLE. |

## **1.3 Project Purpose:**

* The purpose of this document is to display the stocks to the user, which gives a clear understanding to the user related to stock.
* It Recommend user to select stock file based on their past and present data.
* If colour changes from Red to Green then user can Buy that stock.
* If colour changes from Green to Red then user can sell that stock.

## **1.4 Key Project Objectives:**

* User enter into the  application .
* It display the stock file and it helps user to select stock file based on their interest.
* It Recommend user to select stock file based on their past and present data.
* Technical analysis helps identify trading opportunities using actions of market participants through graph.
* If it R-G then user can Buy that stock file.
* If it G-R then user can sell that stock file.
* It generate suggestion report for user to pick good stock .

## **1.5 Project Scope and Limitation:**

* The biggest advantage of technical analysis is that is helps investors and traders predict the future of the market based on past and present data. Up trend, downtrend, and sideways moves of the market are easy to predict, with the help of chart analysis.
* Technical analysis gives early signals and also Generate a graph about the psychology of investors and traders regarding what they are doing.
* Heikin-Ashi Graph provide a lot of information that helps the traders and investors build their positions and take trades.
* Technical Analysis does not give any confirmation this is all about probability.
* This Analysis sometimes give a mixed signals as one indicator shows buy signal and other shows sell signal at the same time.

## **1.6 Functional Overview: -**

* + - 1.6.1 heikin ashi \_01->: Float haclose() :- This function will helps to close the existing open file.
    - 1.6.2 heikin ashi \_02-> void display\_stock() :- This function will display stock from csv files.
    - 1.6.3 heikin ashi 03-> void appendLinkedList() :-This function will add the second Linked List to the first linked list.
    - 1.6.4 heikin ashi \_04-> void printLinkedList() : - This function will print the linked list.
    - 1.6.5 heikin ashi \_05->int LoadFromFile() :- This function will load the stock data from files.
    - 1.6.6 heikin ashi \_06-> int FreeLinkedList ():- This function will free the allocated linked list .
    - 1.6.7 heikin ashi \_07-> void makeheikinashi():- This function will create or make heikinashi graph.
    - 1.6.8 heikin ashi \_08-> void checkColourChange():-

# **2.Design Overview:**

Instant Chatters comprises of the following modules:

|  |  |
| --- | --- |
| Name of the Module | void haclose() , checkColourChange() |
| Handled by | Saurav Kumar |
| Description | This function will helps to close an existing file. |

|  |  |
| --- | --- |
| Name of the Module | void display\_stock(), |
| Handled by | Vijaya Gotarane |
| Description | This function will display stock from csv files.. |

|  |  |
| --- | --- |
| Name of the Module | void printLinkedList(),void makeheikinashi( |
| Handled by | Deshashri kakade |
| Description | This function will print the Linked List. |

|  |  |
| --- | --- |
| Name of the Module | int LoadFromFile() |
| Handled by | Samiksha Dubal |
| Description | This Function will load the stock data from csv files. |

|  |  |
| --- | --- |
| Name of the Module | int FreeLinkedList() |
| Handled by | Meghana Birajdar |
| Description | This function will free the allocated Linked List. |

|  |  |
| --- | --- |
| Name of the Module | void appendLinkedList() |
| Handled by | Deshashri kakade |
| Description | This function will add the second Linked List to the first Linked List. |

**2.1 Design Objectives:**

The main Purpose of this project is to implement a stock analysis application using heikin ashi formula that will help user to select stocks by studying past and present data.With the help of heikin ashi concept we show user to which stock data is better for them to buy or sell.

**2.2 Performance:**

The system will work on the admin terminal. The performance depends on the hardware component of the admin’s system.

**2.3 Maintenance:**

* If maintenance demands consistently keep the Team from completing their Sprint Plan, stop planning for so much. In Sprint Planning, leave some headroom – an allowance for maintenance. Reduce the forecast for new feature work. The size of the allowance may be easy to determine from past Sprints or it may take experimentation. If the Team does not need all of the time budgeted in a Sprint, they can use it for more feature work or payment of technical debt.
* When an allowance is made for maintenance, we can take turns handing it. Team members can rotate in the role of “fixer” from Sprint to Sprint so that no one gets stuck with the cleanup work. The fixer can manage their own time between the maintenance and new work.
* This is the ultimate solution. Use proper development practices – pairing, test-first development, automated acceptance testing, continuous refactoring. Make the code better every day. Wrap the system in automated tests to make bug-hunting easier.

# **3.Environment Description:**

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

**3.2 Language Support: English**

**3.3 User Desktop Requirements:**

* 64-bit processor, 1 GHz or faster
* At least 2 GB free hard drive space
* At least 1 GB RAM

**3.4 Server-Side Requirements:**

* 1.64-bit processor, 1 GHz or faster
* 2.At least 1 GB free hard drive space
* 3.At least 1GB RAM

**3.4.1 Deployment Considerations:**

* Easy setup: no session storage daemon, use tmpfs and memory caching to enhance performance.
* Local storage is used.
* No network latency to consider.
* To scale buys a bigger CPU, more memory, larger hard drive, or additional hardware.

**3.4.2 Application Server Disk Space:**

* No such disk space is required as the program is fully functional on online IDE(s) as well. The Operating System is required and one text file to store the records of processes.

**3.4.3 Database Server Disk Space:**

* No such disk space is required as the program is fully functional on online IDE(s) as well. The Local Operating System is required and one text file to store the records of processes.
  + 1. **Integration Requirements:**
* Language: C
* Tools: Makefile ,Ctags, Valgrind
* Complier: gcc
* Linux Environment

**3.5 Configuration:**

**3.5.1: Operating System:** Linux environment.