



Tribhuvan University
Faculty of Humanities and Social Science

Stock Trading System

A PROJECT PROPOSAL

Submitted to
Department of Computer Application
Pascal National College

In partial fulfillment of the requirements for the Bachelors in Computer Application

Submitted by
Bikash Shrestha | 6-2-1226-5-2020
August 2024

TABLE OF CONTENTS

LIST OF ABBREVIATIONS	3
LIST OF FIGURES	4
1. INTRODUCTION.....	5
2. PROBLEM STATEMENT	5
3. OBJECTIVES	5
4. METHODOLOGY	6
A. Requirement Identification	6
I. Study of existing system	6
II. Literature Review.....	6
II. Requirement Analysis	7
B. Feasibility Analysis	8
I. Technical	8
II. Operational.....	8
III. Economic	8
C. High Level Design of System.....	9
I. Methodology of proposed system	9
II. System Flow Chart.....	10
III. Working Mechanism	11
IV. Description of Algorithms.....	12
5. GANTT CHART	14
6. EXPECTED OUTCOME.....	14
REFERENCES.....	15

LIST OF ABBREVIATIONS

ATS	Automated Trading System
AOT	Automated Order Triggering
CDS	Central Depository System
CNC	Cash and Carry
GTT	Good Till Triggered
IPO	Initial Public Offering
TMS	Trading Management System
NEPSE	Nepal Stock Exchange
NOTS	NEPSE Online Trading System

LIST OF FIGURES

Figure 2. 1 Use-case diagram of stock trading system	7
Figure 2. 2 Methodology to build stock trading system	9
Figure 2. 3 System flowchart of stock trading system.....	10
Figure 2. 4 Working mechanism of purposed system.....	11
Figure 5. 1 Working schedule of stock trading system.....	14

1. INTRODUCTION

Stock exchange is a marketplace where stocks, bonds and other securities are bought and sold. Popular examples of stock exchange are New York Stock Exchange (NYSE), Bombay Stock Exchange (BSE), Shanghai Stock Exchange (SSE) etc.

Stock exchanges now utilize Automated Trading System (ATS) instead of traditional methods. Nepal Stock Exchange also launched its internet-based fully automated online trading system (NOTS) on November 6, 2018 [1]. When a company issues new securities, it enters the primary market through an IPO. Investors buy these securities, and the company becomes listed on a stock exchange. Subsequently, in the secondary market, investors trade these shares among themselves, facilitated by the stock exchange [2].

Stock exchanges have licensed brokerage firms in which traders must register to participate. The online trading system enables remote stock trading in the secondary market. This project aims to create an automated online trading system with a focus on automated order triggering features.

2. PROBLEM STATEMENT

In Nepal, the existing trading management system, doesn't include any risk management features like target and stoploss triggers. Traders aiming to enter or exit the stock market at a specific price must actively monitor market conditions during trading hours. However, this may not be feasible for all traders. In order to resolve the mentioned issue related to secondary market, this project is primarily going to focus on risk management by implementing Automated Order Triggering (AOT) algorithm.

3. OBJECTIVES

The objectives of this project are follows as:

- To develop a system that provide traders to buy and sell the stocks in real time.
- To develop a system that provides feature of stoploss/target triggers and automates the buy/sell order.

4. METHODOLOGY

A. Requirement Identification

I. Study of existing system

Stock trading systems like TMS, Robinhood and Kite were studied for this project to understand how the overall trading system and automated trigger works. Nepal Stock Exchange's TMS only supports Cash and Carry (CNC) limit and market order [3]. In all studied trading systems, broker server gets the trade and stock information from main exchange server. It depends on the brokerage end to provide order trigger features other than market order and limit order. Robinhood and Kite trading systems have features that traders can enter or exit stock holdings until his/her price condition is met. It includes extra trade trigger orders like stop order, stop limit order, trailing order and One Cancel Other (OCO) orders [4] [5].

II. Literature Review

The literatures related to stock trading system and automated order triggers such as conference papers and tons of website articles were reviewed for this project. As reviewed, for stock trading system to function, it requires broker, CDS, banking, clearing & settlement systems. All trade execution is done by exchange server and those trade information are published to all connected brokers in real time. And the traders communicate with their own brokerage for market orders and trade execution status. It depends on the brokerage system to handle all the automated trade triggers [6]. Automated triggers in stock market can eliminate the need for manual monitoring and constant order adjustments. Trade triggers help traders automate their entry and exit strategies. Once set, they may remain active until triggered, saving traders time and effort in managing trades [7] [8].

II. Requirement Analysis

Functional Requirement:

- The application should be able to register and login valid users.
- The application should be able to show balance and holdings information.
- The application should be able to buy and sell stocks.
- The application should be able to set and place stoploss/target triggers.
- The application should be able to place limit and market orders.

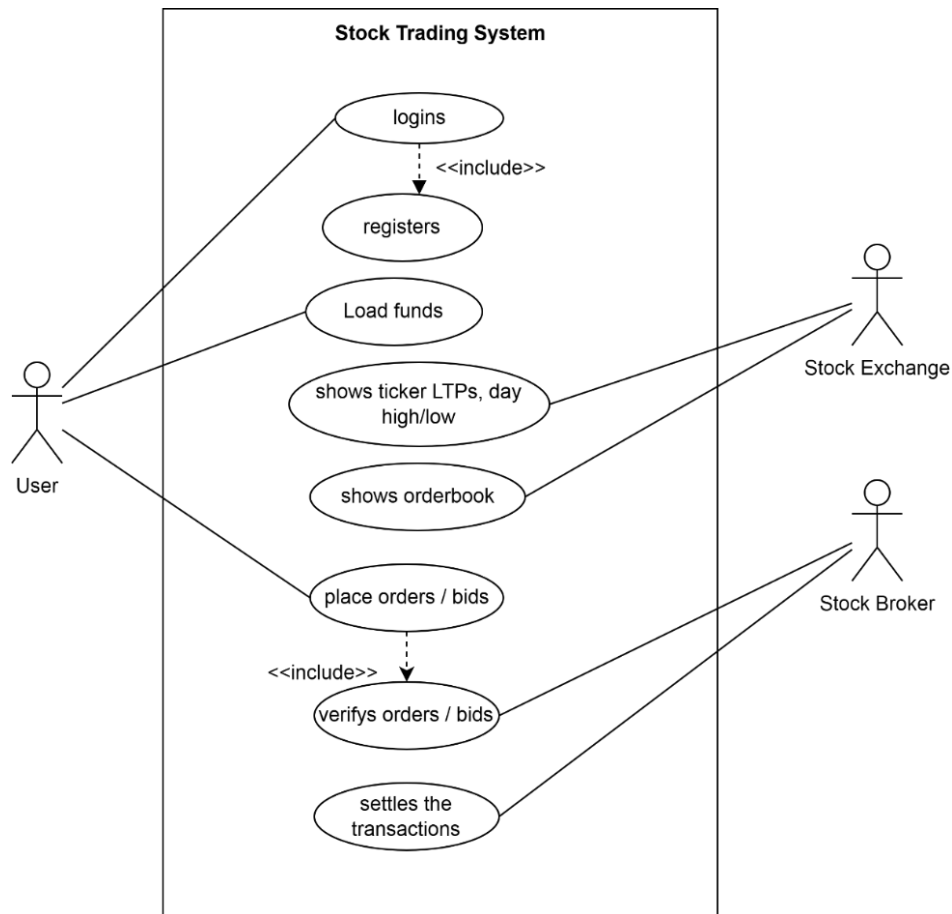


Figure 2. 1 Use-case diagram of stock trading system

Non-Functional Requirement:

- The application should be secured.
- The application should have easy to use and interactive user interface.
- The application should be faster in performance.
- The application should have feature to search for stock information.

B. Feasibility Analysis

I. Technical

This project will only require two different systems for broker and stock exchange. The exchange server requires the message broker to share real-time trade data. The broker server is responsible for verifying the orders before publishing it, placing the automated GTT orders and show the status of placed orders. The stock exchange server is responsible for publishing all valid orders and last transactions of stocks to all subscribers (brokers) as orders are placed by users. All those mentioned functions can be done using existing web technologies and tools so there will be no technical difficulty in developing this project.

II. Operational

This project is simple and intuitive, using this application doesn't require extra knowledge. It is very user-friendly and can be easily adapted by any new user that will use this website. As it uses well-known web technologies, it is easy to maintain.

III. Economic

This project will be very cost-friendly as it uses existing free open-source technologies like RaabitMQ, Spring boot, VS Code and open-source tools. This project won't require money for its development. The only thing is to be done is making an environment for the development with an effective supervision.

C. High Level Design of System

I. Methodology of proposed system

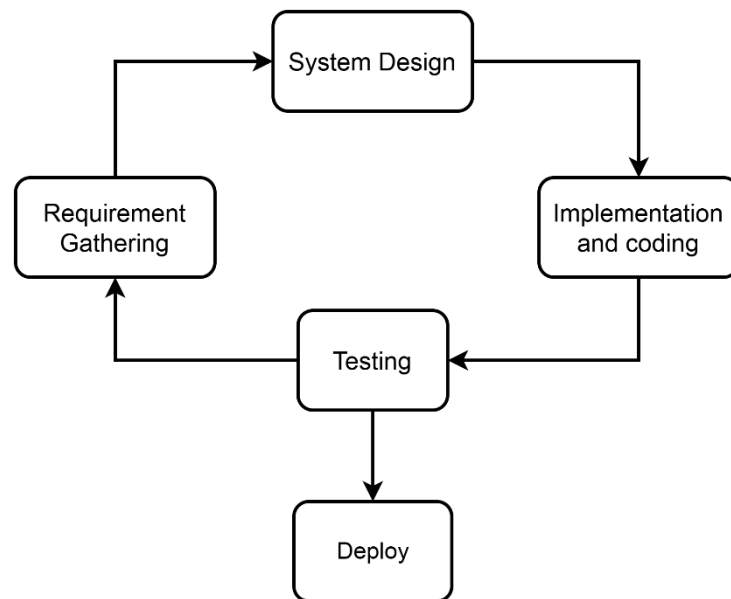


Figure 2. 2 Methodology to build stock trading system

For this project development, the functional requirements will be gathered first. Low level and high level system design such as system flowchart, working mechanism of system, schema, interface, system architecture is designed based on the requirements. After designing all the system necessary modules and components, the system will be implemented based on those designs. Testing of the each modules and functions will be done, if necessary further new requirements will be gathered for more feature before deploying it.

II. System Flow Chart

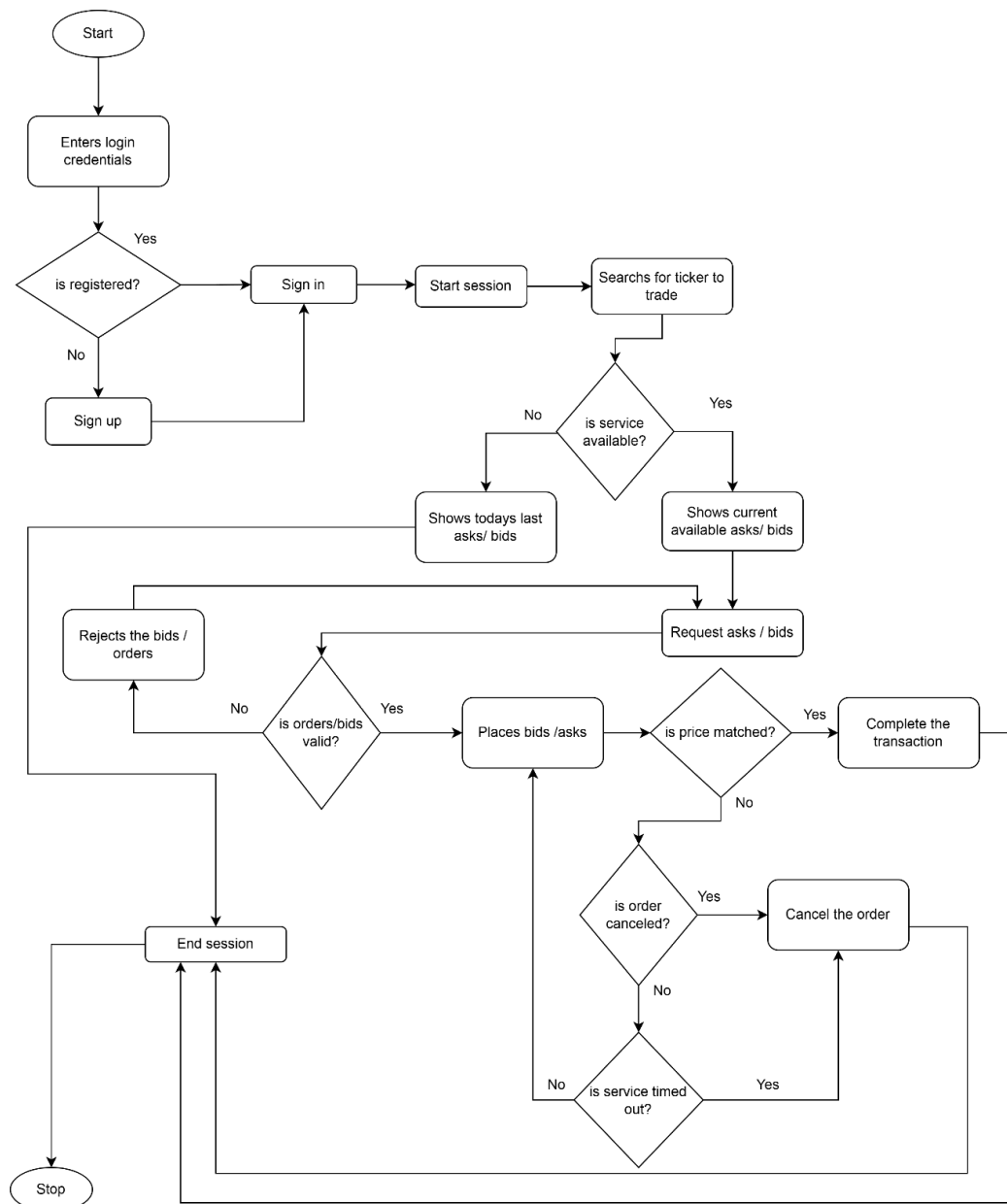


Figure 2. 3 System flowchart of stock trading system

In above flowchart, it shows the user's perspective of interaction with the trading system.

1. User registers and logs in with valid credentials.
2. User searches for stocks to trade, if system trading is available then it shows orderbook which contains all asks and bids of that stock.
3. User requests for placing the order, after verified by system order is placed in on the orderbook.
4. System waits for buy/sell price to matched until trading service available or gets cancelled.
5. User ends the session.

III. Working Mechanism

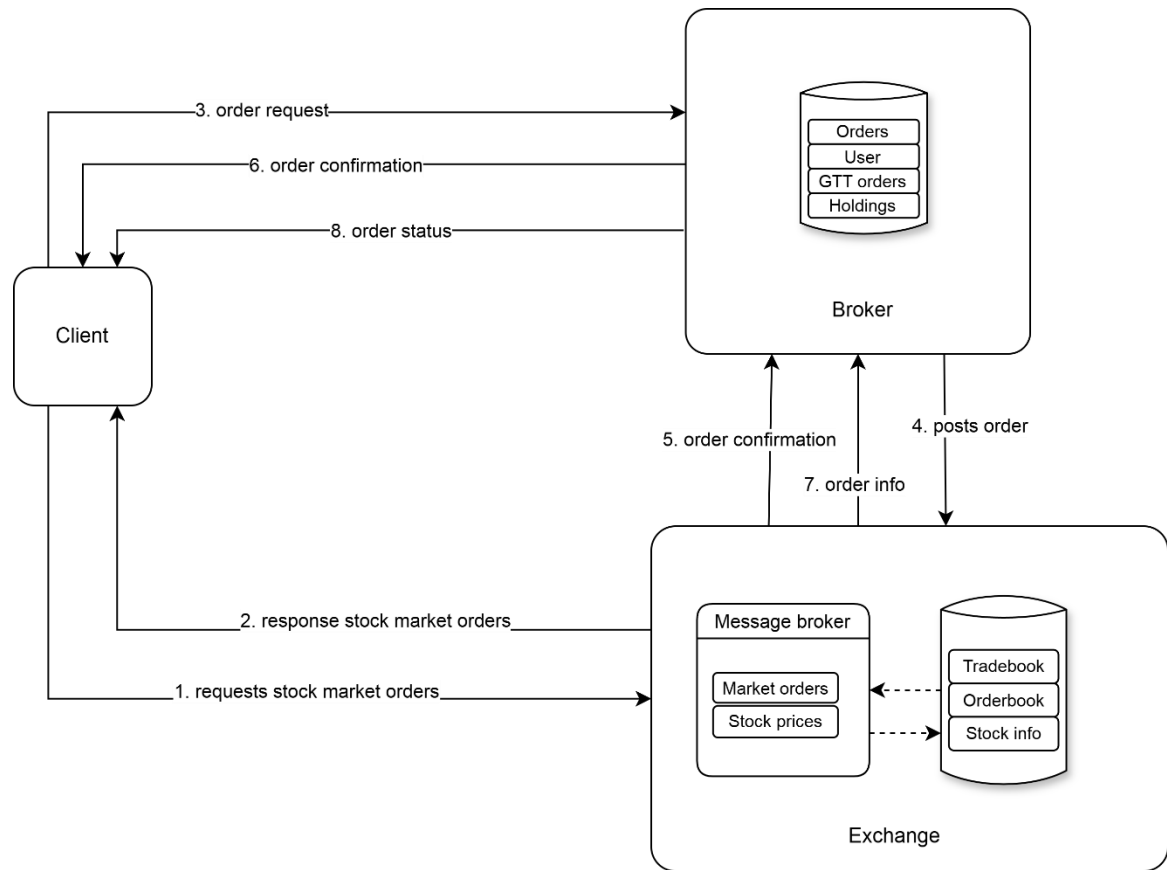


Figure 2. 4 Working mechanism of proposed system

The working mechanism of proposed system are as follows:

1. Client requests for current market bids/asks to exchange server.
2. Exchange responds with current stock market orders.
3. Client requests broker to place order.
4. Broker verifies the order before posting order to exchange.
5. After placing order into orderbook exchange sends confirmation to broker.
6. Broker sends the order confirmation to the client.
7. Exchange provides the update of order status to broker.
8. Broker sends those order status to client.

IV. Description of Algorithms

Automated Order Trigger (AOT) is a feature which allows user to set a trigger price, such that; if user trigger price is hit in a future date, a limit order will be placed on the exchange as per the limit price and preset conditions set by user.

Conditions:

1. AOT feature is only allowed for limit orders.
2. Trigger can only be placed when market is live.
3. A user can have only 50 active AOTs.
4. Target price must be greater than 0 and should be 0.25% away from current market price.
5. Once the trigger price is breached and such limit price selected is outside the circuit limits of the particular scrip then it is rejected.

Algorithm to place valid GTT order

1. Input order type from the user
2. Check if the (order type == buy) or (order type == sell single)
 - 2.1. Input trigger price and target price
 - 2.2. Check trigger price is 0.25% away from current market price (CMP) i.e Absolute $(\text{CMP} - \text{trigger price}) / \text{CMP} * 100 \geq 0.25$
 - 2.2.1. Add order to GTT list and total active trigger ++
3. Check if the order type == sell OCO
 - 3.1. Input target order trigger, target price, stoploss trigger and stoploss price.
 - 3.2. Check if target > stop loss
 - 3.2.1. Check if target and stop loss trigger price is 0.25% away from current market price (CMP) i.e Absolute $(\text{CMP} - \text{trigger price}) / \text{CMP} * 100 \geq 0.25$
 - 3.2.2. Add order to GTT list and total active trigger ++
4. Else return "Invalid trigger request"
5. End

Algorithm to place the GTT order:

1. While stock market is live:
 - 1.1. For each unique stock in gtt order list
 - 1.2. Fetch the last traded price of that stock every 50 ms
 - 1.3. If LTP is updated then:
 - 1.3.1. Fetch all single_leg orders having (trigger_price <= LTP)
 - 1.3.2. For each order in single orders_list
 - 1.3.2.1. Place buy/sell limit order
 - 1.3.2.2. Set order status from "active" to "triggered"
 - 1.3.3. End for
 - 1.3.4. Fetch all double_leg orders having (stoploss_trigger >= LTP or target_trigger <= LTP)
 - 1.3.5. For each order in double orders_list
 - 1.3.5.1. Place sell limit order
 - 1.3.5.2. Set order status from "active" to "triggered"
 - 1.3.6. End for
 - 1.4. End If
 - 1.5. End For
2. End While
3. End

5. GANTT CHART

	Week 1	Week 2	Week 3	Week 4	Week 5	Week 6	Week 7	Week 8	Week 9
Requirement Gathering									
System Design									
Development Phase									
Testing									
Documentation									
Deployment									

Figure 5. 1 Working schedule of stock trading system

The working schedule of the proposed system includes the first 2 weeks of requirement gathering and next 4 weeks of designing the system and 4 weeks of developing and writing code and 4 weeks of testing as well. All system design, development and testing will be done interchangeably. Documentation of the project will be starting from system design until the testing ends. The deployment of the project will be on the last week after all the documentation is finished.

6. EXPECTED OUTCOME

This project will provide trading platform which works in real-time exchange of stocks with automated order triggering features, reducing risk for traders.

REFERENCES

- [1] The Kathmandu Post, "Nepse Online Trading System goes live," The Kathmandu Post, 7 November 2018. [Online]. Available: <https://kathmandupost.com/money/2018/11/07/nepse-online-trading-system-goes-live>. [Accessed 08 June 2024].
- [2] Corporate Finance Institute, "Stock Exchange," Corporate Finance Institute, [Online]. Available: <https://corporatefinanceinstitute.com/resources/equities/stock-exchange/>. [Accessed 22 July 2024].
- [3] Sharesansar, "Frequently Asked Questions (FAQ)," [Online]. Available: <https://www.sharesansar.com/faq>. [Accessed 26 July 2024].
- [4] Zerodha Kite, "Introducing GTT – Good Till Triggered orders," 22 July 2019. [Online]. Available: <https://zerodha.com/z-connect/kite/introducing-gtt-good-till-triggered-orders>. [Accessed 22 July 2024].
- [5] Robinhood, "Investing with stocks: The basics," [Online]. Available: <https://robinhood.com/us/en/support/articles/stop-limit-order/>. [Accessed 25 July 2024].
- [6] B. Dhakal and M. K. Gupta, "A System Model of Online Trading System for Nepal," in *2014 International Conference on Computing for Sustainable Global Development (INDIACom)*, 2014.
- [7] Srishti Mathur, "What is GTT Order? Learn Meaning of Good Till Triggered Orders & How to Execute them," 3 April 2024. [Online]. Available: <https://www.smallcase.com/learn/what-is-gtt-good-till-triggered-order/#why-use-gtt>. [Accessed 29 June 2024].
- [8] A. Ganti, "Trade Trigger: Meaning, Overview, Pros and Cons," Investopedia, 21 January 2022. [Online]. Available: <https://www.investopedia.com/terms/t/trade-trigger.asp>. [Accessed 21 July 2024].