STOCK INDICATOR SCANNER CUSTOMIZATION TOOL USING DEEP REINFORCEMENT LEARNING

BY

DESMOND CHEONG YONGHONG

A REPORT

SUBMITTED TO

Universiti Tunku Abdul Rahman

in partial fulfillment of the requirements

for the degree of

BACHELOR OF COMPUTER SCIENCE (HONOURS)

Faculty of Information and Communication Technology
(Kampar Campus)

JAN 2022

STOCK INDICATOR SCANNER CUSTOMIZATION TOOL USING DEEP REINFORCEMENT LEARNING

BY

DESMOND CHEONG YONGHONG

A REPORT

SUBMITTED TO

Universiti Tunku Abdul Rahman

in partial fulfillment of the requirements

for the degree of

BACHELOR OF COMPUTER SCIENCE (HONOURS)

Faculty of Information and Communication Technology
(Kampar Campus)

JAN 2022

UNIVERSITI TUNKU ABDUL RAHMAN

REPORT STATUS DECLARATION FORM

Title: __STOCK INDICATOR SCANNER ____________________________
__CUSTOMIZATION TOOL USING DEEP______________________
__REINFORCEMENT LEARNING____________________________

Academic Session: ___JAN 2022____

I _______DESMOND CHEONG YONGHONG______________
(CAPITAL LETTER)

declare that I allow this Final Year Project Report to be kept in
Universiti Tunku Abdul Rahman Library subject to the regulations as follows:

1. The dissertation is a property of the Library.
2. The Library is allowed to make copies of this dissertation for academic purposes.

Verified by,

(Author's signature)

(Supervisor's signature)

Address:
__19, Lorong SP/15, _________
__Taman Suria Permai, _______
__34000, Taiping, Perak. _____

Ku Chin Soon
Supervisor's name

Date: ___21 APRIL 2022______

Date: 22/04/2022

Universiti Tunku Abdul Rahman			
Form Title : **Sample of Submission Sheet for FYP/Dissertation/Thesis**			
Form Number: **FM-IAD-004**	Rev No.: **0**	Effective Date: **21 JUNE 2011**	Page No.: **1 of 1**

FACULTY OF INFORMATION AND COMMUNICATION TECHNOLOGY

UNIVERSITI TUNKU ABDUL RAHMAN

Date: ___21 APRIL 2022_____

SUBMISSION OF FINAL YEAR PROJECT /DISSERTATION/THESIS

It is hereby certified that ______***Desmond Cheong Yonghong***_______ (ID No: __***18ACB04824***__) has completed this final year project entitled "___***Stock Indicator Scanner Customization Tool Using Deep Reinforcement Learning***_" under the supervision of ___ Ts Dr Ku Chin Soon ___ (Supervisor) from the Department of ____ Computer Science ___, Faculty of ___ Information and Communication Technology ____ .

I understand that University will upload softcopy of my final year project in pdf format into UTAR Institutional Repository, which may be made accessible to UTAR community and public.

Yours truly,

____
(DESMOND CHEONG YONGHONG)

DECLARATION OF ORIGINALITY

I declare that this report entitled "**STOCK INDICATOR SCANNER CUSTOMIZATION TOOL USING DEEP REINFORCEMENT LEARNING**" is my own work except as cited in the references. The report has not been accepted for any degree and is not being submitted concurrently in candidature for any degree or other award.

Signature : __________________________

Name : DESMOND CHEONG YONGHONG

Date : 21 APRIL 2022

ACKNOWLEDGEMENTS

I would like to express my sincere thanks and appreciation to my supervisors, Ts Dr. Ku Chin Soon, for willing to undertake me as his supervisee. I am glad to be given the bright opportunity to engage in a machine learning project. It is my first step to establish a career in machine learning field. A million thanks to you.

Moreover, I am grateful for the helps and supports I received from my friends and course mates at UTAR. Finally, I must say thanks to my parents and my family for their love, support, and continuous encouragement throughout the course.

ABSTRACT

Nowadays, there have some applications provide predictive model for user to predict the stock trend, however user cannot customize the type of input data used in the predictive models. User cannot use the indicator that they prefer to make the prediction. Other than that, many current stock indicator scanners only allow user to specify some simple conditions to scan the stocks and do not harness the advancement of machine learning. This project will deliver a web application with dynamic stock prediction model based on deep reinforcement learning or more particularly, Deep Q-Network (DQN) algorithm which enable input customization. In this system, user able to create their own indicator by choose a combination of some well-known fundamental indicators and technical indicators that are provided in the application. This indicator can then be used as the input of the predictive model. The stock indicators selected by user will be the input of DQN algorithm and act as state while the actions allowed for the DQN agent will be buy and sell. For simplicity, return of investment (ROI) will be used as the reward of RL agent.

Since stock data is sequential data and Recurrent Neural Networks (RNN) works better on sequential data compared to classical feedforward DNN, thus the feedforward DNN used in classical DQN have been replaced by a specialized version of RNN called LSTM. By using LSTM instead of RNN, the short-term memory problem of RNN which caused by vanishing gradient problem can be overcome. To address the problem of overfit, dropout regularization technique will be used. Moreover, ADAM optimization technique will be applied to adjust the parameters of the network in the DQN and ReLU activation function will be used since these techniques have shown promising achievement in some literature reviews. The mean squared error (MSE) loss function used in classical DQN will be replaced by Huber loss to improve the stability of the model training.

TABLE OF CONTENTS

LIST OF TABLES

Table Number **Title** **Page**

LIST OF FIGURES

Figure Number **Title** **Page**

LIST OF SYMBOLS

S	Set of states
A	Set of actions
a	action taken
t	Time
s_t	State at time t
s_{t+1}	State at time $t+1$
$P_a(s_t, s_{t+1})$	probability of action a in s_t lead to s_{t+1}
$R_a(s_t, s_{t+1})$	expected reward received after changing from s_t to s_{t+1} by doing action a
π	Policy of reinforcement learning

LIST OF ABBREVIATIONS

IPO	Initial Public Offering
ETF	Exchange-Traded Fund
DJIA	Dow Jones Industrial Average
S&P 500	Standard & Poor’s 500
EPS	Earnings Per Share
P/E	Price-to-Earnings
P/B	Price-to-Book
PEG	Price/Earnings-to-Growth
SMA	Simple Moving Averages
EMA	Exponential Moving Averages
MACD	Moving Average Convergence Divergence
RSI	Relative Strength Index
OBV	On-Balance Volume
RL	Reinforcement Learning
MDP	Markov Decision Process
DL	Deep Learning
NLP	Natural Language Processing
DNN	Deep Neural Network
WT	Wavelet Transforms
SAEs	Stacked Autoencoders
LSTM	Long Short-Term Memory
RNN	Recurrent Neural Network
DR	Direct Reinforcement
RRL	Recurrent Reinforcement Learning
ReLU	Rectified Linear Unit
FDDR	Fuzzy Deep Direct Reinforcement
FRDNN	Fuzzy Recurrent Deep Neural Network
BPTT	Backpropagation Through Time
DRL	Deep Reinforcement Learning
SCOT	Sparse Coding-Inspired Optimal Training

DDR	Deep Direct Reinforcement
MODRL	Multi-Objective Deep Reinforcement Learning
ADAM	Adaptive Moment Estimation
GDQN	Gated Deep Q-learning
GDPG	Gated Deterministic Policy Gradient
GRU	Gated Recurrent Unit
MA	Moving Averages
VR	Volatility Volume Ratio
OBV	On Balance Volume
TDQN	Trading Deep Q-Network
OHLCV	Open-High-Low-Close-Volume
CNN	Convolutional Neural Network
ROI	Return on Investment
DAX	Deutscher Aktien Index
UI	User Interface
VM	Virtual Machine
IDE	Integrated Development Environment
HTML	HyperText Markup Language
CSS	Cascading Style Sheets
API	Application Programming Interface
DBMS	Database Management System

CHAPTER 1: INTRODUCTION

1.1 Project Inspiration

In stock market there have a lot of opportunity to gain profit. However, just like many other investments, it often is a challenge to grasp that opportunity. Since the stock market exists, stock investors always try to predict the trend of the stock and hope to improve their profit in stock trading. However, there have a lot of uncertainty in the stock market and the price of a stock is influenced by many factors. In order to have a good profit in the stock market, investor need to spend a lot of time to analyze the stock and the market. The analysis is very time consuming and tedious. According to Drakopoulou [1], the stock valuation techniques normally used by stock investor can divided into 2 categories, that is Fundamental Analysis and Technical Analysis.

As artificial intelligence and machine learning rise nowadays, people have the chances to apply this new technology in stock market. Machine learning made the replacement of tedious and time-consuming stock analysis and more accurate prediction compared to traditional methods possible. Many people have tried to utilize this new approach to forecast the trend of stock. The result of prediction can then facilitate investor in decision making when trading stock.

1.2 Problem Statements

To forecast the trend of stock effectively and efficiently, many of the machine learning approaches have been proposed by researcher. Although the existing approach have shown promising results in stock market, there have still some limitations for those approaches. The main problem is **user cannot customize the type of input data using existing predictive models**. The type of input data used by the predictive model is fixed and user cannot use the indicator that they prefer to make the prediction. Investor forced to deal with the fixed input data types that they less familiar and have less confidence. On the other hand, many current stock indicator scanners **only allow user to specify some simple conditions and do not harness the advancement of machine learning**. User needs to provide some conditions such as EPS greater than RM3.00 and ROE (Return on Equity) greater than 15% to filter the stocks. To solve this problem, this project has been proposed.

1.3 Project Objectives

Objective 1: **To deliver a web application with dynamic stock prediction model based on deep reinforcement learning which enable input customization**.

User able to choose a combination of some well-known fundamental indicators and technical indicators that are provided in the application to create their own indicator. The used of indicator instead of raw data of market is aimed to reduce the training cost of the model without compromise the performance of the model.

Objective 2: **To assist investor in decision making when trading a stock by provide the stock trend forecast**.

User can gain insight from the prediction of model by using the indicators they frequently applied in stock market instead of unfamiliar indicators used in other models. User can completely follow the prediction result in stock trading or treat the prediction result as a reference just like the advices from an expert. The transaction cost, investment risk, and profit will be considered to deliver a practical model which have good performance in real stock market.

1.4 Project Scope

The scope of this project is to deliver a dynamic model to predict the trend of stock market. User will be allowed to **customize the input of the predictive model** on the stock. The types of customizable inputs can be used by users is indicators which is a structured data. Some well-known fundamental indicators and technical indicators which will be available for user to create their own indicator. User can do so by choosing a combination of fundamental indicators and technical indicators. The purpose of this functionality is to let investor to use the indicator they prefer during the stock trend forecast. This functionality also enable investor to test the capability of new indicator they come out without using real money. Even though many research have shown that unstructured input like news and social media able to increase the accuracy of predictive model, but this input type will not be included in this project due to the unstructured data is more difficult to handle.

Besides, the proposed system will provide **stock trend prediction** based on deep reinforcement learning. This feature enable investor to forecast the stock trend more effective and efficient compared to traditional methods. The prediction result also is expected to assist investor in decision making when trading stock and thus improve the profit of stock trading. The trend of the stock (buy or sell) will be the output of the model. Then, the proposed system will **evaluate the performance of the model generated**. The results of the model evaluation enable user to know and compare the performance of the models created. Furthermore, user will be allowed to **save the indicators** they created and **save the models** they customized. This makes user able to use the same indicator created or modify indicator created for other models. Next time, user also can directly apply the model generated without waiting for model training. To limit the scope of the project, the model will only focus on top 100 stocks in Malaysia. Other financial instruments and other markets will not be considered.

1.5 Project Impact and Contribution

The contribution of this project is the delivery of a **novel web application with stock prediction model which contain customizable input**. User can choose and create the indicators they prefer to use for the training of the model. The trend of the stock will be the output of the model. Investor can make trading decision based on the forecast stock trend and even improve the profit gained in the stock trading. Investor also can use this system to examine the performance of the new indicator they come out with using real money in stock market.

1.6 Chapter Summary

The structure of this paper is arranged as: Chapter 2 reviews about existing stock analysis techniques, current machine learning approaches for stock trend forecast, and some existing stock forecasting system. Chapter 3 describes the methodology of the proposed system. Chapter 4 illustrate the designs of the system. Chapter 5 describe the testing performed on the system and evaluate on the outcome of the testing. Chapter 6 discuss about system limitation and some future works. Finally, last chapter makes a conclusion to the system and project.

CHAPTER 2: LITERATURE REVIEW

2.1 Overview of Malaysia Stock Market

Stock market is a regulated public market which let people to issue, buy, and sell stock that found in a stock exchange [2]. Stocks exchange also known as securities exchange or bourse is a marketplace to facilitate investor in the trading of financial instruments. Malaysia stock exchange called Bursa Malaysia and it located in the capital city of Kuala Lumpur. Financial instruments can be found in bourse including stocks, currencies, commodities, futures, bonds, options, and so on. Stocks also called share or equity, act as fractional ownership of a company. A company can raise capital to expand their businesses by issue shares instead of borrowing from bank. This let the company avoids from the increase of debt and reduce the spending on the interest of debt. When a company choose to become a public company at the first time, their need to approaches an investment bank and let the investment bank deal with the initial public offering (IPO). The IPO market also called as primary market where investor directly buy the share from the company. All trade occur after primary market is called secondary market where the stocks are trade between investors [2].

Other than investor and investment bank, there have many other stock market participants. Stockbroker is a licensed professional who buys and sells the securities on behalf of investors. Stockbroker act as an intermediary between investor and stock exchange. Portfolio manager is a professional who invest portfolio for client. Portfolio is a collection of securities. Company like mutual fund, hedge fund, exchange-traded fund (ETF), and pension plan will hire portfolio manager to manage the investment of the company. Due to portfolio manager will buy and sell large quantities of stocks, the stock price may be affected significantly by their trading decision [2].

For company looking for listing in Malaysia, Bursa Malaysia offers 3 choice of markets, that is Main Market, ACE Market, and LEAP Market. Main Market is a prime market for established companies that must met the standards in terms of quality, size, and operations [3]. ACE Market is a sponsor-driven market for companies with growth prospects [3]. Both Main Market and ACE Market are easier to perceive by the capital market. This characteristic makes the listing companies able to receive funds from various investors. On the other hand, LEAP Market is an adviser-driven market which provides greater fund-raising access to newly developing companies like small-and-medium enterprises (SME). However, unlike Main Market and ACE Market, companies list in LEAP Market only able to raise funds from sophisticated investors as defined in Capital Markets and Services Act 2007 [3]. To accommodate the needs of Islamic finance growing, Bursa Malaysia also has established an Islamic Market that provides end-to-end Islamic securities exchange platform and act as a marketplace for Shariah-compliant instruments trading [4].

In a stock market, there have an index to show the overall performance of the market. Stock market indexes consists of a set of selected stocks which have different weight on the indexes [5]. Some well-known stock market indexes including Dow Jones Industrial Average (DJIA), Standard & Poor's 500 Index (S&P 500), NASDAQ Composite Index, and Hang Seng Index. In Malaysia, the stock market index is FTSE Bursa Malaysia KLCI (Kuala Lumpur Composite Index), which is composed of the top 30 companies on Bursa Malaysia. Every index has different way to calculate. Stock market indexes themselves can be traded in the exchanges too. The main income of stock exchange come from the transaction fee charged for every trade carried out on its platform. Other than that, listing fee charged to a company during IPO, regulatory fee, and registration fee charged to broker are other income source of stock exchange [2].

2.2 Fundamental Analysis

Fundamental Analysis believe that each stock has a "intrinsic value". Fundamental Analysis assume that the current market value of a stock does not reflect its intrinsic value, but in long term, the trend of the stock will follow its intrinsic value [6]. Fundamental Analyst will buy the stock when it is undervalued and sell the stock when it is overvalued by the market. Since share represent fractional ownership of a company, fundamental analyst believes that the intrinsic value and the prospect of a stock can be determined by focus on the underlying factors that affect the business. The Fundamental Analysis can further divide into qualitative analysis and quantitative analysis. Qualitative analysis analyzes non-quantifiable information such as political scandals, accounting frauds, inflation, consumer behavior, and threat of war while quantitative analysis focus on quantifiable information like revenue, expenses, assets, liabilities, and also other information which can retrieve from financial statement of a company [6]. Some popular indicator used in fundamental analysis include earnings per share (EPS), price-to-earnings (P/E) ratio, price-to-book (P/B) ratio and Price/Earnings-to-Growth (PEG) ratio.

2.2.1 Earnings Per Share (EPS)

EPS of a company can be obtained from divide the net profit of the company with the number of shares outstanding. EPS shown how much a company can earn per share, in another words, the profitability of the company [7].

2.2.2 Price-to-Earnings (P/E) Ratio

P/E ratio is the result of market value per share divided by EPS as the formula shown below.

$$P/E\ ratio = \frac{market\ value\ per\ share}{EPS}$$

Generally, P/E ratio is applied by investor to determine the relative value of a stock. Therefore, P/E ratio can be applied in the comparison of a stock to its own historical performance and also comparison with other stocks in the same industry. High P/E ratio might indicate that the stock is overvalued and vice versa [8]. For companies that do not have earning or losing, their EPS value will be null, thus value of P/E ratio will be empty too.

There have 2 types of P/E ratio, that is forward P/E ratio and trailing P/E ratio. Forward P/E ratio also called estimated P/E ratio, is P/E ratio which uses future earnings in calculation. This P/E ratio can be used to compare current earnings to future earnings. However, just like other estimations, underestimate and overestimate of forward P/E ratio might happen. Different analysts and investor might also come out different forward P/E ratio and cause confusion. While trailing P/E ratio is P/E ratio which calculated based on the past performance of the stock. Normally, EPS over the past 12 months will be used to calculate trailing P/E ratio. Although trailing P/E ratio do not have the same problems as forward P/E ratio, however, trailing P/E ratio which rely on company past performance cannot provides enough insight to the future performance of company. Less reflective to the fluctuation of stock prices is another problem of trailing P/E ratio, this is due to the company earnings released each quarter while stocks traded daily [8].

2.2.3 Price-to-Book (P/B) Ratio

By divide the market price per share with book value per share, we can get P/B ratio of a stock as the equation shown below.

$$P/B\ ratio = \frac{market\ value\ per\ share}{book\ value\ per\ share}$$

Book value is the difference between total assets and total liabilities of a company. P/B ratio let investor to compare the stock price with the assets of the company if the company bankrupt instantly [9]. When a company have positive book value but negative earnings, P/B ratio still can be used while P/E ratio will become useless. Since number of companies with negative book values is less then companies with negative earnings, P/B ratio also has been widely used by investor. However, P/B ratio will become less useful for companies with little tangible assets like companies in service and information technology sector. Although P/B ratio can provide some insight to identity does a stock is undervalued or overvalued, but there is not a specific value of P/B ratio to tell that. Different industries will have different good P/B ratio and other indicators can also be considered to have more accurate interpretation to a P/B ratio [9].

2.2.4 Price/Earnings-to-Growth (PEG) Ratio

PEG ratio is the enhanced version of P/E ratio by including the expected earnings growth to the calculation. PEG ratio can be calculated by divide the P/E ratio of a stock with the earnings growth rate for a specified period as formula shown below.

$$PEG\ ratio = \frac{P/E\ ratio}{EPS\ Growth}$$

Lower PEG ratio indicated that the stock likely be undervalued based on its expected future earnings. A well-known investor Peter Lynch thinks that PEG ratio of 1.0 denoted a fairly valued company. Thus, stock with PEG ratio more than 1.0 is considered overvalued while PEG ratio lower than 1.0 is considered undervalued [10]. By using different growth in the calculation, PEG ratio can be divided into forward PEG which uses future growth and trailing PEG which uses historical growth. The common period of growth rate used are 1 year, 3 years, and 5 years.

2.3 Technical Analysis

On the other hands, instead using the intrinsic value, Technical Analysis use the statistics generated by market activity to analyze the stock [11]. Technical Analyst believe that the market price reflects all information that impact the market, therefore they do not analyze the economic growth, news, company development and other information like fundamental analyst. There has various method can be used by a technical trader to predict the price or trend of a stock for instance chart patterns and technical indicators. All the technique used in technical analysis is based on 3 assumptions, that is, (1) the market discounts everything, (2) prices move in trends, and (3) history tends to repeat itself [11]. Some renowned technical indicators include simple moving averages (SMA), exponential moving averages (EMA), moving average convergence divergence (MACD), relative strength index (RSI), on-balance volume (OBV) and much more.

2.3.1 Moving Averages (MA)

Moving averages (MA) is calculated to extract the price data by creating a series of average price. MA enables investor to analyze the price data of a stock easier since the impacts of random and temporary fluctuations of the stock price over a specified period are mitigated [12]. MA usually been used to identify the stock trend direction or determine the level of support and resistance of a stock. The common period used to calculate MA are 15, 20, 30, 50, 100, and 200 days. MA will be more sensitive to price change if shorter period used and vice versa. Therefore, shorter MA is more suitable for short-term trading, while longer MA more proper for long-term trading [12].

MA can be divided into 2 types, that is simple moving averages (SMA) and exponential moving averages (EMA). SMA is calculated by taking mean of a set of prices over specific number of days while EMA is a weighted average which give more priority to the recent prices. Below is the formula of SMA where A_n is the price of an asset at period *n* and *n* is the period.

$$SMA = \frac{A_1 + A_2 + \cdots + A_n}{n}$$

Below is the formula of EMA where EMA_t is EMA today, V_t is value today, EMA_y is EMA yesterday, *s* is smoothing, and *d* is number of days. The value of smoothing can be calculated by using formula 2 / (period + 1).

$$EMA_t = \left[V_t \times \left(\frac{s}{1+d}\right)\right] + EMA_y \times \left[1 - \left(\frac{s}{1+d}\right)\right]$$

2.3.2 Moving Average Convergence Divergence (MACD)

MACD is calculated by minus 12-day EMA with 26-day EMA of a stock as shown below.

$$MACD = 12\ day\ EMA - 26\ day\ EMA$$

Then, a signal line which consist of 9-day EMA is stack on top of MACD line. The typical way to interpret both signal line and MACD line are crossover, divergence and rapid rises or falls. It is a buy signal when MACD line crosses above the signal line and sell signal when MACD line crosses below the signal line. It is bullish divergence when MACD form 2 rising lows while the price form 2 falling lows and bearish divergence when MACD form 2 falling highs while the price form 2 rising highs. When MACD rises rapidly, it indicated that overbought is happen. Likewise, when MACD falls rapidly, it indicated that oversold is happen [13].

2.3.3 Relative Strength Index (RSI)

RSI is a momentum oscillator which can have value from 0 to 100. RSI is used to measure the degree of change in price to show overbought or oversold in the market [14, p.13]. When the number and size of positive closes increase, value of RSI will rise while when the number and size of losses increase, value of RSI will drop. Generally, a stock is considered overbought when its RSI is above 70% and oversold when below 30% [15].

2.3.4 On-Balance Volume (OBV)

OBV is a momentum indicator (oscillator) which use the change of volume to forecast the price of a financial instrument. OBV can be used as a measure of crowd sentiment which can used to forecast whether a security will be bullish or bearish. The theory of OBV is based on the difference between smart money like institutional investors and less sophisticated retail investors. When institutional investors start buying a stock that retail investors are selling, the price will remain at a relative level even though the volume will increase. Eventually, the price will rise due to the rise of the volume. Institutional investors begin to sell at this moment, while smaller investors begin to buy [16]. Below is the formular of OBV.

$$OBV = OBV_{prev} + \begin{cases} volume, & if\ close\ > close_{prev} \\ 0, & if\ close\ = close_{prev} \\ -volume, & if\ close\ < close_{prev} \end{cases}$$

OBV is the cumulative sum of the stock volume which add the previous OBV with current volume if current close price is larger than previous close price. If current close price is less than previous close price, the current OBV is previous OBV minus current volume. If the current close price same as previous close price, then value of OBV remains same.

2.4 Overview of Reinforcement Learning: Q-Learning

In Reinforcement learning (RL), an agent is trained by interact with an environment. In this environment, the agent will come across different situation called states s_t. Under a state, the agent can perform an action a. This action will bring rewards r_t to the agent and cause the agent to come to new state s_{t+1}. The reward can be positive or negative and the objective of the agent is to maximize the total reward it gets across an episode. An episode is everything happened from the first state to the terminal state. At the end, the agent will learn the best action given a state or also known as strategy or policy π.

RL methods can divided into 3 types, actor-only, critic-only, and actor-critic method. Actor-only method directly learns a policy from data so that maximum reward can received from every action performed in every state. While critic-only method learns a policy by optimize a value function or expected future returns. Actor-critic method is the combination of actor-only and critic-only method where actor derive the policy and critic evaluate the action performed to maximize the final rewards [17].

Except first state, every state is the result of its previous state. However, it is infeasible to store the information for all states. To solve this problem, we assume what the agent do in the environment is Markov Decision Process (MDP) which every state follows Markov property that is every state depends merely on the previous state and the transition of the state [18].

The Q-Learning consists of a Q-table that contains the expected future reward or also known as Q-value $Q(s_t, a_t)$ for each state-action pair. With this "cheat sheet", the agent able to take the best action for every state and generate maximum total rewards. At the beginning, the Q-value in the Q-table will be initialized randomly. The agent can choose to "explore" the environment by taking random action or “exploit” the Q-table by perform the action with highest Q-value for the given state s_t. Once that agent receives the reward r as the outcome of perform action a and change from state s_t to new state s_{t+1}, the Q-value $Q(s_t, a_t)$ will be updated using the equation below where α is learning rate and is γ discount factor [19].

$$Q^{new}(s_t, a_t) = Q(s_t, a_t) \; + \; \alpha \cdot (r_t \; + \; \gamma \cdot \max_{a} Q(s_{t+1}, a) - \; Q(s_t, a_t))$$

The discount factor γ is used to measure how important about future reward compared to immediate reward. Discount factor γ will have value between 0 and 1. Since the main objective is to get maximum total reward, long term rewards is more important than short term rewards, therefore discount factor γ will be set to large.

2.5 Overview of Deep Reinforcement Learning: Deep Q-Learning

Although the capability of Q-Learning is great, however, it is less feasible to deal with complex environment with many possible states and actions. To save and update the Q-table under large number of states, large number of memories are required. Besides, the agent needs more time for the environment exploration since the number of states is large [18]. To overcome these problems, Deep Q-Learning or Deep Q-Network (DQN) has been created. Instead of gets the Q-value from Q-table and update the Q-table in Q-Learning, Deep Q-Learning use neural network (Deep Learning) to estimate the Q-value and update the neural network. The comparison between Q-Learning and Deep Q-Learning is illustrated in Figure 2.5.

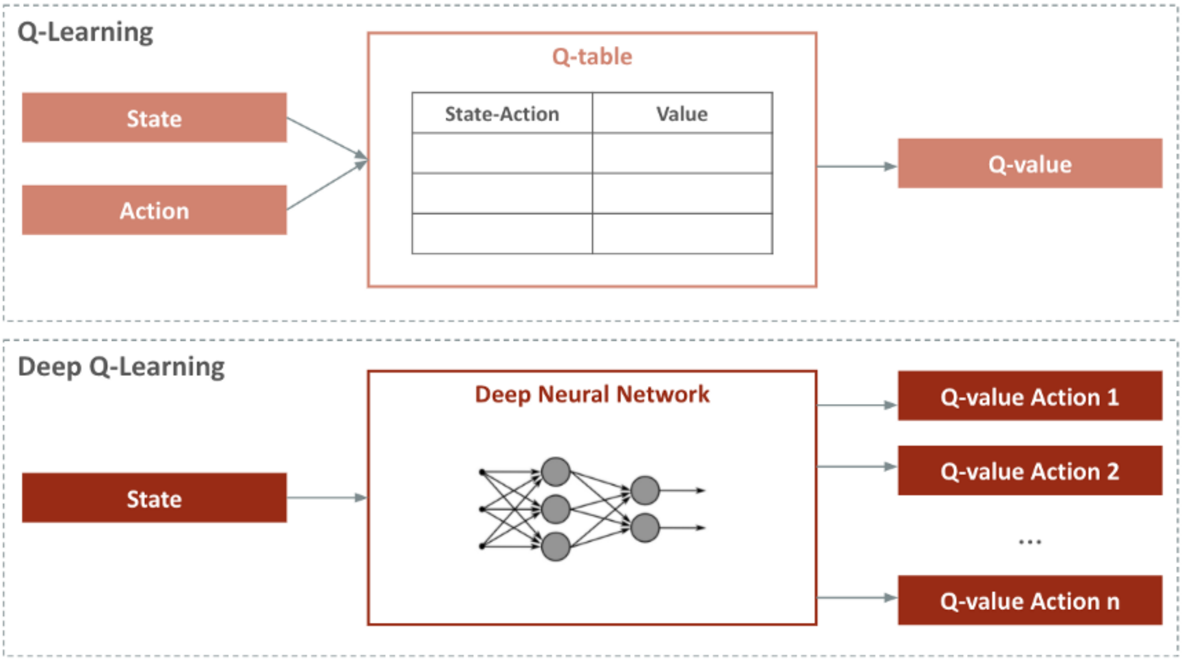


***Figure 2.5** Comparison between Q-Learning and Deep Q-Learning. [19]*

The loss function used in the deep neural network is Mean Squared Error (MSE) of predicted Q-value and the target Q-value as show in equation below where $r_t + \gamma \cdot \max_a Q(s_{t+1}, a)$ is target Q-value and $Q(s_t, a_t)$ is predicted Q-value [19].

$$loss = (r_t + \gamma \cdot \max_a Q(s_{t+1}, a) - Q(s_t, a_t))^2$$

From the equation above, we can notice that the value of the target Q-value is unstable. This mean the network is learning a constantly changing value and this will cause the training very unstable. To solve this problem, 2 neural networks is used instead of 1. One is the Prediction Network which responsible for predicted Q-value and another is Target Network which responsible for target Q-value. Both networks will have same structure and weights at the beginning, but during training, the weights of Target Network are fixed while the weights of Prediction Network been updated. The weights of Prediction Network will be copied to Target Network every *n* episodes specified. Due to the target Q-value is relatively stable, the training is more stable too [18].

Moreover, experience replay is often used in DQN. Rather than update or train the network using the information get in single step, the information *(s_t, a_t, r_t, s_{t+1})* get from several steps are stored in replay memory. Then, a minibatch obtained from random sampling the replay memory is used to update the network. Experience replay enables the algorithm to "remember" past experiences [19], prevent local optimal problem [17] and makes the training more stable.

Deep Reinforcement Learning (DRL) have been applied in several area in real life. For instance, in the development of self-driving cars, Wayve.ai has applied DRL in lane following task to train a car how to drive. In addition, DRL also can be used in industry automation. One example for this is Deepmind has been used to control the temperature of Google Data Centers which bring 40% less in energy consumption. In the area of Natural Language Processing (NLP), DRL also have been applied. For instance, in dialogue generation, conversations are simulated between 2 virtual agents, then sequences which contain crucial conversation characteristics like coherence, informativity, and ease of response will receive more reward. Moreover, DRL also has been used in robotics manipulation to train robots to grasp different objects. The trained robots can then be deployed in assembly line building products. On the other hand, DRL also have been used in trading and finance. In stock market, compared to other time series model which predict the stock prices, DRL able to determine which action should perform for the given stock price [20].

2.6 Deep Learning Approach

The used of deep learning (DL) in the stock market has been widely studied. For example, Arévalo et al. [21] used deep neural networks (DNN) to perform high-frequency trading strategy. The stock prediction strategy proposed used current time, pseudo-log-returns of last *n*, prices standard deviations of last *n* and created trend indicators of last *n*, where *n* is the window size for the input of DNN. Different with other approach reviewed by et al. [21], time is included as an input in the proposed approach and the results show that time is useful to exploit repetitive patterns in the stock market. The strategy proposed will predict next 1-minute average price. Buy, sell or do nothing will be decided after compare the predicted price will current price. The proposed strategy has a pretty good performance which shown 66% directional accuracy and 81.77% successful trades. Although the strategy proposed have good performance in the experiment, but it might not very practical in real stock market because the transaction cost does not been considered. Since the system might buy or sell every minute, the profit is not optimum and the investor need to pay a lot of transaction fee.

Other than that, Bao, Yue & Rao [22] proposed a deep learning framework to forecast the close price of 6 famous market indexes where each 2 represent developing market, relatively developed market and developed market. As shown in Figure 2.5.1, in the proposed framework, the input data are preprocessed using wavelet transforms (WT) to eliminate noise, then used stacked autoencoders (SAEs) for features extraction and lastly predict the closing price of stock by Long Short-Term Memory (LSTM). Input variables used in the proposed deep learning framework include daily trading data, technical indicators, and macroeconomic variables. The proposed WSAEs-LSTM is compared with WLSTM, LSTM, and recurrent neural network (RNN) in predictive accuracy test and together with Buy-and-hold strategy in profitability test. The results show that the performance of proposed framework is more excellent compare with other due to WSAEs-LSTM able to obtain more abstract and invariant features.

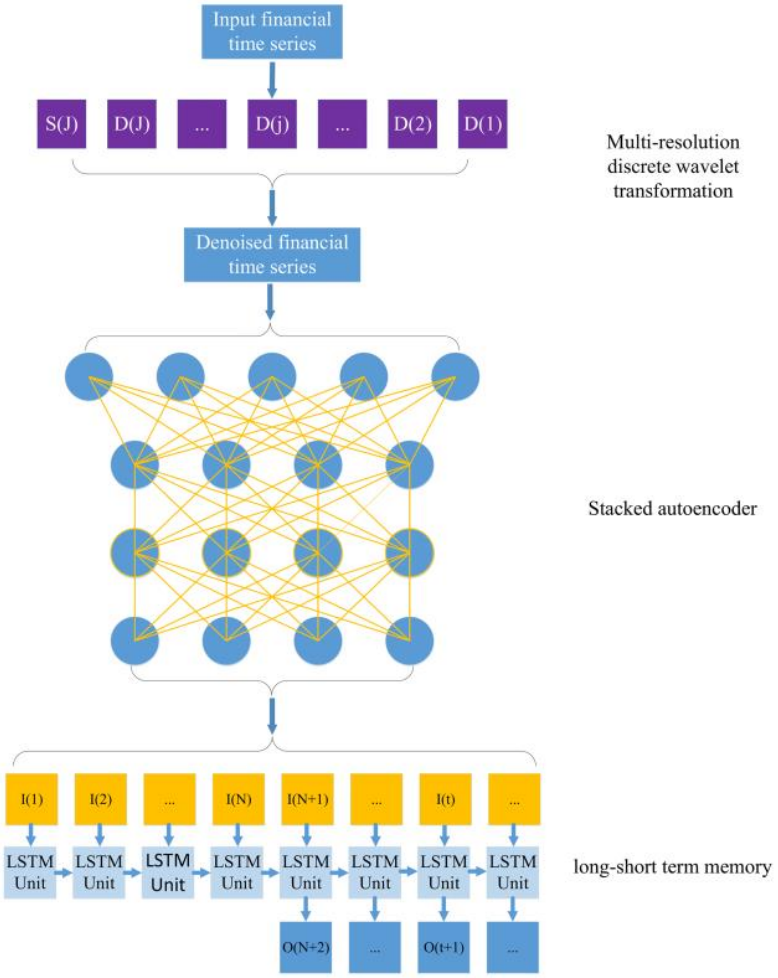


Figure 2.5.1 *Deep Learning Framework for financial time series. [22]*

2.7 Reinforcement Learning Approach

Stock trend prediction using reinforcement learning (RL) also has been tried by many other researchers and astonishing results have been shown. Moody & Saffell [23] proposed an implementation of trading system based on direct reinforcement (DR) approach using recurrent reinforcement learning (RRL). As shown in Figure 2.6.1, the performance function *U(θ)* is used to optimize the parameter *θ* of trading system directly. The paper show that the used of RRL enables to represent the problem in simpler form and prevent Bellman's curse of dimensionality. Instead of optimize profit, differential Sharpe ratio and downside deviation ratio which rewards high average positive returns and penalizes risky returns are introduced to optimize risk-adjusted return. According to the experiment results, compare with Q-Learning that need to learn a value function, DR which can directly derive a policy without having to learn a value function has better performance, interpretability. and computational efficiency.

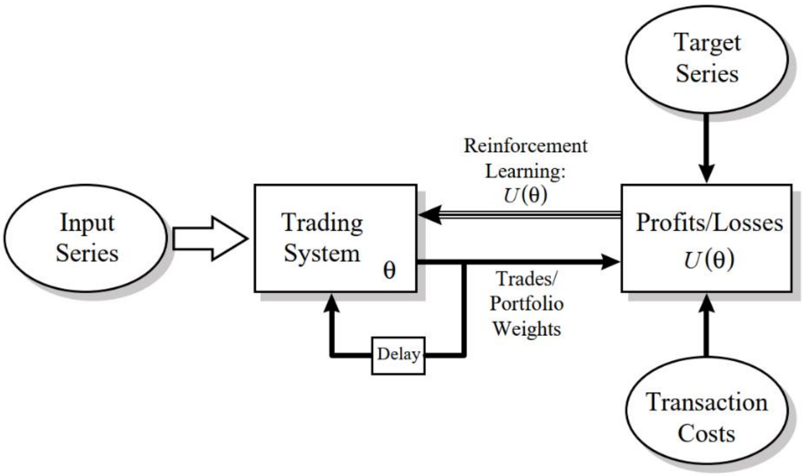

Target Series
Reinforcement Learning: $U(\theta)$
Input Series
Trading System θ
Profits/Losses $U(\theta)$
Trades/ Portfolio Weights
Delay
Transaction Costs

Figure 2.6.1 *Trading System based on Direct Reinforcement [23]*

Besides stock market, application of RL in forex market also has been studied. Carapuço, Neves & Horta [24] has proposed a short-term speculation in forex trading by adapt the rectified linear unit (ReLU) and gradient descent with adaptive learning rate to the Q-learning. In the proposed framework, historical tick data which contain largest intraday data is used as the input of prediction model. They discovered that the reward signal which only based on profit obtained will have consistent behavioral flaws. Instead of choosing Sharpe ratio which consider both upside and downside risk as new reward signal, modification of Sharpe ratio that is Sortino ratio which only consider downside risk is chosen. Although the proposed framework shown satisfactory result with average yearly profit 16.3% to 19.1%, but the framework is train and test in a simulated market, thus the performance and reliability in real forex market is questionable. Besides, the proposed approach only focuses on forex market and not necessary can works well in other market like stock market.

2.8 Deep Reinforcement Learning Approach

The promising results of deep learning and reinforcement learning in the stock market have inspired researcher to combine both methods to have better performance. Deng et al. [25] proposed Fuzzy Deep Direct Reinforcement (FDDR) which used DL for feature learning and RL for the deep representations to predict stock price. The fuzzy recurrent deep neural network (FRDNN) structure which introduced in the paper consists of fuzzy learning to reduce the uncertainty of the input data, DNN for feature learning while recurrent neural network (RNN) responsible to produce trading policy. Figure 2.7.1 illustrated the overview of FRDNN where *w* is weight and *u* is reward. Purple panel in the figure is fuzzy representation part while the blue panel is deep transformation part. The green nodes in the figure represent the input vector of the model. For the ease of model explanations, author chose total profit as the reward. However, the comparison of total profit and Sharpe ratio as reward shown that Sharpe ratio as reward has more reliable performance. Due to the complicated structure of FRDNN proposed, task-aware backpropagation through time (BPTT) has been introduced for fine tuning the model. The FDDR proposed is compare with Buy-and-Hold strategy, deep reinforcement learning (DRL), sparse coding-inspired optimal training (SCOT) system, and deep direct reinforcement (DDR) on total profit and Sharpe ratio. FDDR shown impressive performance in the experiment. The model not only can used on stock indexes, but it also works well for commodity future contracts trading. Due to the model is technical-indicator-free, it is easier for user to choose the stocks from the market. However, the verification of robustness show that the training cost of the proposed method is high since large layer number and nodes number per layer is needed for good performance. To solve the high training cost, fundamental indicator and technical indicator which summarize the market information can be used as inputs instead of raw price tick data.

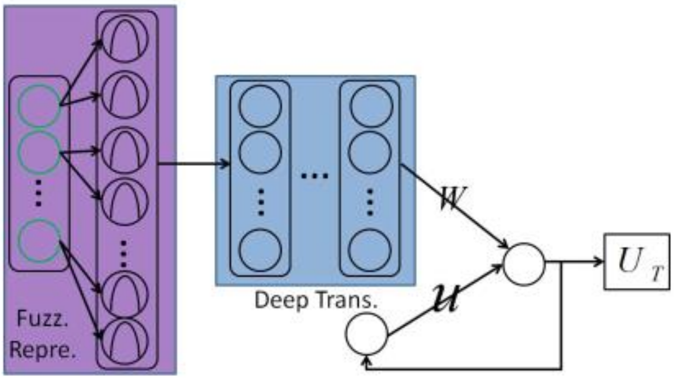


Figure 2.7.1 *Overview of FRDNN [25]*

Different kind of RNN have also been applied by other researchers in stock market. Si et al. [26] introduce a multi-objective deep reinforcement learning (MODRL) approach for intraday trading of stock index future. In MODRL, multiple fully connected hidden layers are used for hidden features learning and reduce the uncertainty of input data. Then, a special kind of RNN that is Long Short-Term Memory (LSTM) is used for decision making in the trading. Figure 2.7.2 shown the overview of MODRL. A multi-objective structure has been introduced to balance the profit and risk of investment. The goal of RL agent (reward) is maximize the multi-objective structure which subtract average daily profits with volatility which also known as risk. MODRL accept raw data of market as input and will output trading signal that is buy, sell, and hold. To prevent overfitting problem in the MODRL, dropout technique is applied on the input layer and 4 fully connected layers. To fine-tune the parameters of the network and the learning rate automatically, adaptive moment estimation (ADAM) optimization technique is used. The proposed MODRL has compared with RRL and Buying and Holding Strategy on index-based future contracts in China. Sharpe Ratio, average profits and total trading number of each approach is measured during the evaluation and profit & loss curve is plotted. The evaluation result show that MODRL has the best performance among them. Although the total trading number of MODRL is the highest, but the profit of MODRL is the highest too. This indicated that MODRL is able to adjust its trading positions at a proper time and reduce the trading cost. However, MODRL only tested on stock index future in China market, the performance of MODRL for other financial instrument in other market is still arguable.

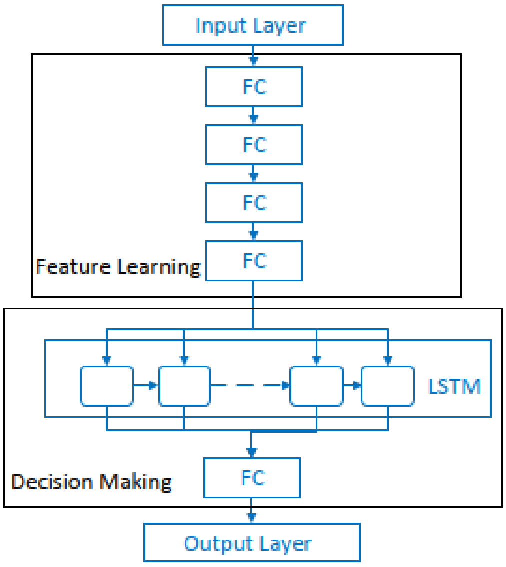


Figure 2.7.2 *Overview of MODRL [26]*

Gated Deep Q-learning trading strategy (GDQN) with critic-only framework and Gated Deterministic Policy Gradient trading strategy (GDPG) with actor-critic framework is proposed by Wu et al. [17] as adaptive stock trading strategies with deep reinforcement learning (DRL). Gated Recurrent Unit (GRU) is applied in the model to extract financial features of a stock. Dropout regularization technique is used to solve the overfitting problem that cause by the sampling noise generated by GRU. Due to the raw data of the market is full of noise and highly complicated, a variety of technical indicators are used to extract market information from various perspectives. Moving averages (MA), EMA, MACD, BIAS, volatility volume ratio (VR) and on balance volume (OBV) are some of the technical indicators deployed as input. As the GDQN framework illustrated in Figure 2.7.3, an action a_t will be returned from the Q network after input the current state s_t to the network. The agent will perform a_t and observe r_t and s_{t+1}. To solve the problem of infinite memory needed due to infinite number of states exist, an experience replay memory has been added. Usually in the traditional DQN, as the number of action increase, the estimation error of the DQN will increase too, thus, 2 neural networks that is Q network and target Q network have been deployed. Q network used to select a_{t+1} to maximize the Q value while target Q network calculates the a_{t+1}. On the other hand, as shown in Figure 2.7.4, GDPG framework consists of an actor module and a critic module. Actor module is responsible to update the trading strategy while critic module responsible to control the actor module for the update. Each module contains 2 neural networks with same structure but different parameters. Instead of using Sharpe ratio which compare the investment with risk-free investment, Sortino ratio which more focus on downside risk and more suitable for high-volatility investment evaluation is used as the reward of the model. The proposed GDQN and GDPG has then been compared with Turtle trading strategy and tested on multiple stocks in different stock market using Sortino ratio and return rate as performance indicator. The results indicated that GDQN and GDPG have better performance compared to Turtle trading strategy as both have better Sortino ratio and return rate compared to Turtle trading strategy. The results also indicated that actor-critic strategy of GDPG has more stable return compared to GDQN which used critic-only strategy.

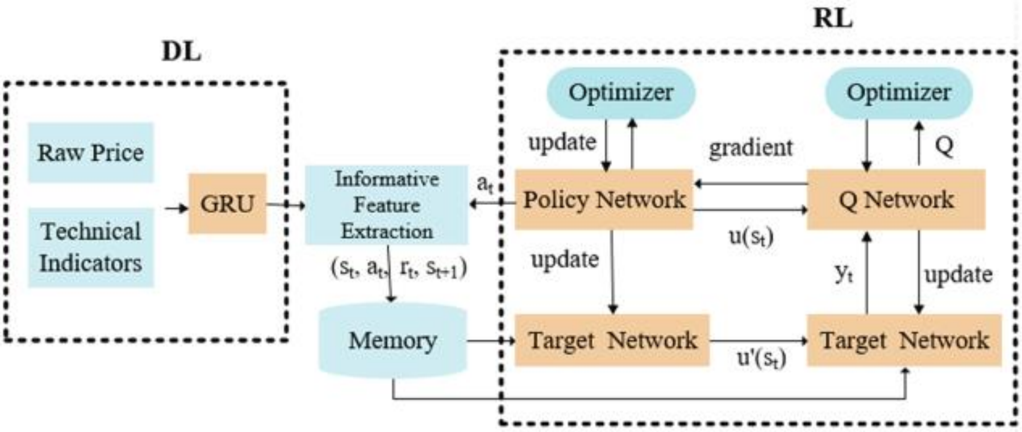


Figure 2.7.3 The framework of GDQN [17]

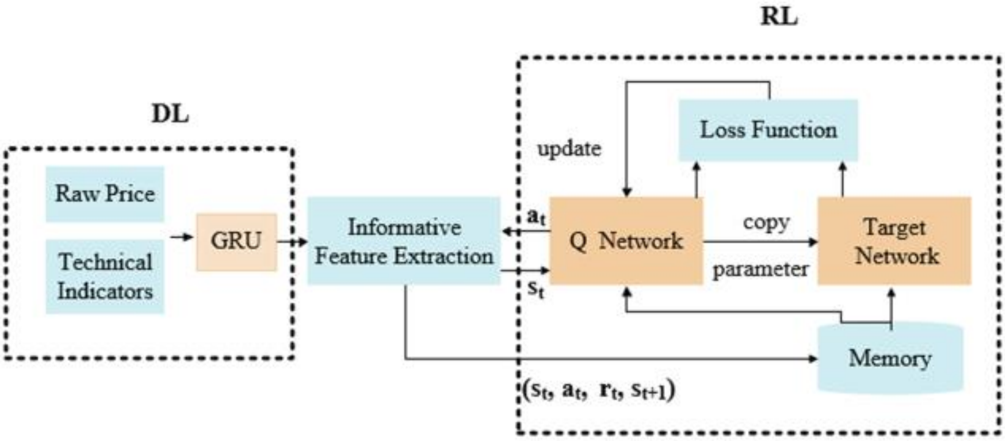


Figure 2.7.4 The framework of GDPG [17]

Théate & Ernst [27] has presents Trading Deep Q-Network algorithm (TDQN) to use in algorithmic trading. Algorithmic trading is a trading approach which trading decisions is made based on a set of mathematical rules computed by a machine. The proposed TDQN is used to determine the optimal trading position at trading time. In the proposed method, state information of RL agent and Open-High-Low-Close-Volume (OHLCV) of stock is used as inputs. Compared to some paper above, daily returns have been chosen as RL reward to prevent sparse reward setup instead of Sharpe ratio or Sortino ratio which consider the risks of investment. In TDQN, a feedforward DNN with leaky ReLU activation functions is used instead of convolutional neural network (CNN). Other than that, double DQN, ADAM optimizer, Huber loss, Xavier initialization and so on have been applied to increase the performance of the model. Three regularization techniques that is dropout, L2 regularization and Early Stopping is used to handle the overfit problem of DRL. TDQN is compared with Buy-and-hold strategy, Sell-and-hold strategy, trend following with moving averages, and mean reversion with moving averages on 30 stocks from different sector and region. Certain performance indicators like Sharpe ratio, profit & loss, annualized return, annualized volatility, Sortino ratio, maximum drawdown, maximum drawdown duration, and the rest is used to measure the performance of each strategy. During the performance assessment of Apple stock, the results show that TDQN able to achieves promising results and diverse trading costs by reduces its trading frequency. However, during the performance assessment of Tesla stock, high max drawdown percentage, long maximum drawdown duration, and low profit show that TDQN does not perform well on high volatility stock and does not handle investment risk properly compared to other trading strategy. To resolve this problem, Sharpe ratio or Sortino ratio can be used as RL reward for better address to the investment risk. Other than OHLCV of stock, some fundamental indicators and technical indicators can be added as input of the model to improve the performance.

An ensemble of reinforcement learning approach using multiple DQN is proposed by Carta et al. [28]. In the proposed multi-DQN approach, 3 types of action can be performed by the RL agent are long action which buy the stock and sell it before market closes, short action that used uncovered sale, sell the stock and buy it before market closes and opt-out action which do nothing. Return on Investment (ROI) ratio which divide the dollar return with dollar invested is used as reward of RL agent in the proposed multi-DQN. As shown in Figure 2.7.5, decisions of multiple agents trained at different epochs which represent strategies of investors with different experiences is combined to address uncertainties of stock markets. There has a log file for each agent to record the decisions made by them meant for easy trace in the future. A threshold will be used to select the final action. Furthermore, multi-resolution method is proposed to summarize the input information provided to RL agent using different time resolution and history depths. The multiple DQN with different strategy (only-long, only-short, long+short, and full) are used to compare with Buy-and-Hold strategy in market indexes

S&P500 and Deutscher Aktien Index (DAX). The performance of each strategy is evaluated using accuracy, maximum drawdown, coverage, ratio and profit & loss. The experiment shows that the multi-DQN able to provide convincing results compared to Buy-and-Hold strategy. However, the experiment is done on market indexes only and no individual stocks were tested. The proposed method also does not validate on other financial instrument like commodities and bonds. Therefore, the reliability of multi-DQN on those markets still ambiguous. On the other hand, multi-DQN only used past prices data to train the model and do not consider other information like fundamental indicators, technical indicators and news. The proposed multi-DQN also does not evaluate on trading frequency and address the transaction cost of investment. If the trading frequency is high, investor will lose a lot of profit due to the transaction cost. To address transaction cost and have better control on investment risk, reward other than ROI ratio can be used.

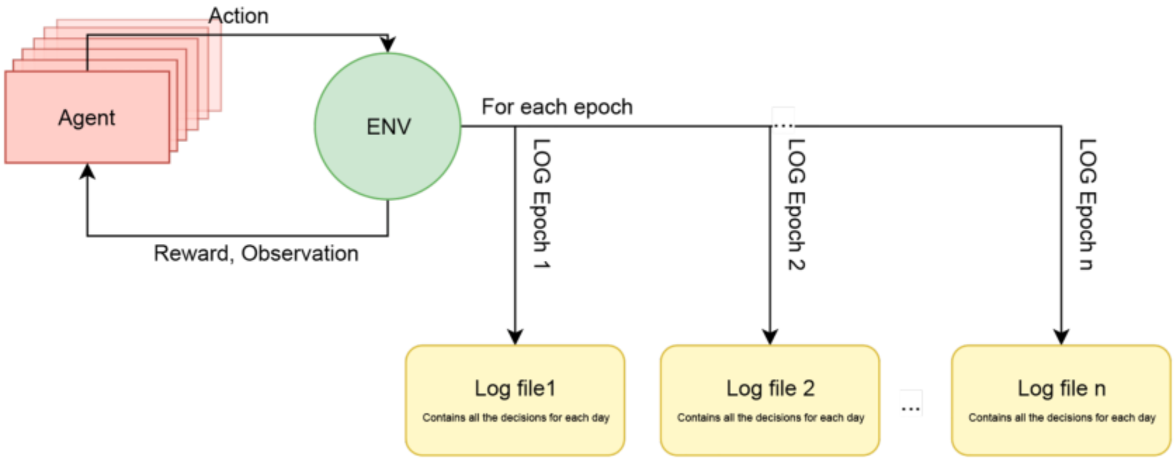
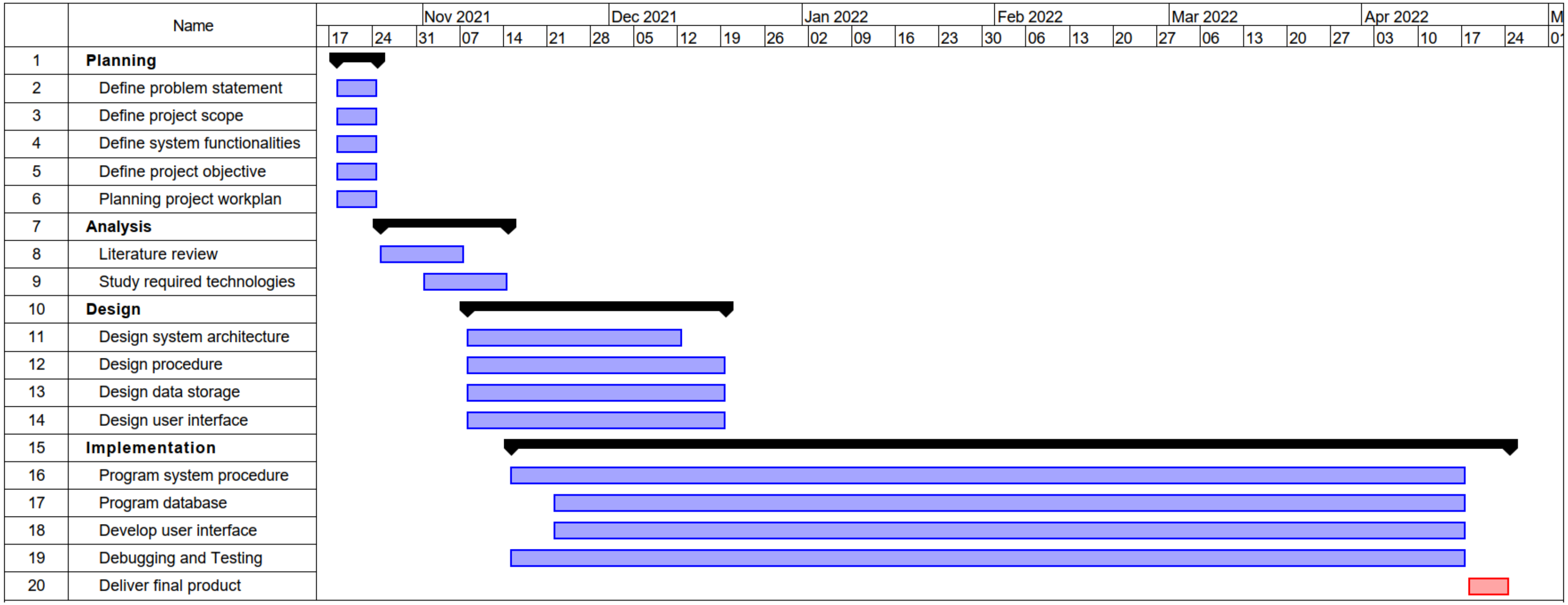

Action
Agent
ENV
For each epoch
Reward, Observation
LOG Epoch 1
LOG Epoch 2
LOG Epoch n
Log file1
Contains all the decisions for each day
Log file 2
Contains all the decisions for each day
...
Log file n
Contains all the decisions for each day

Figure 2.7.5 *Overview of multi-DQN [28]*

CHAPTER 3: SYSTEM METHODOLOGY

3.1 Project Development



	Name
1	**Planning**
2	Define problem statement
3	Define project scope
4	Define system functionalities
5	Define project objective
6	Planning project workplan
7	**Analysis**
8	Literature review
9	Study required technologies
10	**Design**
11	Design system architecture
12	Design procedure
13	Design data storage
14	Design user interface
15	**Implementation**
16	Program system procedure
17	Program database
18	Develop user interface
19	Debugging and Testing
20	Deliver final product

Figure 3.1 *Gantt chart of the project*

As the Gantt chart shown in Figure 3.1, the problem statement of this project is defined first. Then, determine the scope of the project and the functionalities that need to include in the system. At the same time, the objectives that wish to achieve in the project is defined. After that, a project workplan is developed.

In analysis phase, literature review had been carried out to have better understanding on existing approaches used to forecast stock trend. The strengths and weaknesses of existing approaches will be studied. To deliver better approach, possible solution for the weaknesses will be studied. The tools and technologies required in this project will be identified and investigated. The features, strengths, and weaknesses of the tools and technologies will be studied to ensure them suitable for this project.

In design phase, the overall system architecture will be designed first. Then, using activity diagram to design the procedures required to accomplish each system functionality. The data storage design of the system is decided in this phase too. After that, the user interface which determine how user interact with the system will be designed.

During implementation phase, programming will be carried out to realize all designs in the design phase. The required development environment will be setup for the development of system procedure, database, and user interface. During the whole development process, debugging and testing of the system will be conducted from time to time in order to make sure the system performs as intended. Lastly, after finished all the implementation, the final system will be delivered.

3.2 Information Gathering

The data required for the predictive model is the **stock information** in the Malaysia stock market. For simplicity, only 100 stocks in the stock market will be consider. These stocks are Bursa Malaysia Top 100 Index Constituents as show in Appendix A. The Open-High-Low-Close-Volume (OHLCV) of the stock and **company financial information** like revenue, expense, total assets, and total liabilities is needed to retrieve for the calculation of the value of fundamental indicators, and technical indicators. The indicators provide for user are as show in table below.

Fundamental Indicator	**Description**
EPS	Earnings Per Share
P/E Ratio	Price-to-Earnings Ratio
P/B Ratio	Price-to-Book Ratio
PEG Ratio	Price/Earnings-to-Growth Ratio
Technical Indicator	**Description**
Open	Open price of the stock
High	The highest price of the stock traded
Low	The lowest price of the stock traded
Close	Close price of the stock
Volume	Stock volume traded in the period
SMA	Simple Moving Averages
EMA	Exponential Moving Averages
MACD	Moving Average Convergence Divergence
RSI	Relative Strength Index
OBV	On-Balance Volume

Table 3.2 *List of indicators provide to user*

The indicator values get from the calculation will then use in the training of the predictive model. On the other hand, the OHLCV of the stock also will be used in the visualization of the performance of the predictive model by plotting a graph. When new stock market data available, the new data will be used to update the predictive model and keep the prediction accurate. The information of stocks will be retrieved from Yahoo Finance using Yahoo Finance API.

3.3 System Functionalities for Stock Indicator Scanner Customization Tool

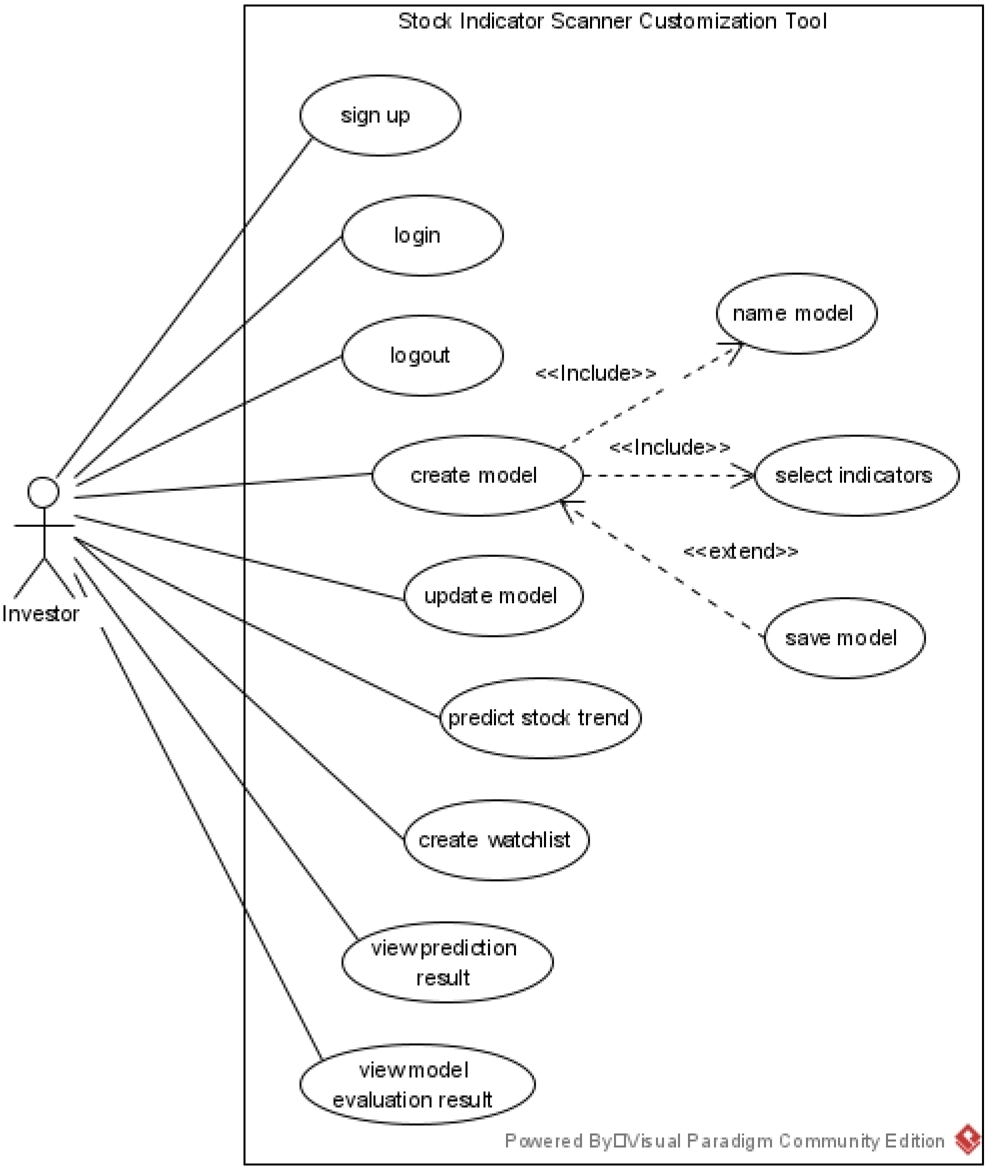


Figure 3.3 *System functionalities of Stock Indicator Scanner Customization Tool*

In the Stock Indicator Scanner Customization Tool, user can sign up and login to their account, and logout. Then, investor can create a prediction model by provide a name for the model and select indicators used in the model training. The indicators available for user to select is show in Table 3.2. If the model is not up to date, investor can choose to update that model. The latest stock data will be retrieved and trained to update the model. Furthermore, investor also can choose a model to forecast the trend of the stocks. The selected model will be used in the prediction and investor can view the prediction result.

Besides, investor also can view the model evaluation result to know the capability and performance of the model created. Then investor can decide to save or abort the model created. To save the model, investor need to login to their account. Each investor will be limited to save at most 5 model in this system due to the limited storage. If investor save the model, investor can select and use that model to predict the stock trend next time. Other than that, investor can create watchlists by select some stocks as the member in the watchlist. Investor can decide to forecast the trend of all stocks or only the trend of stocks in the selected watchlist.

3.3.1 Design of stock trend prediction procedure

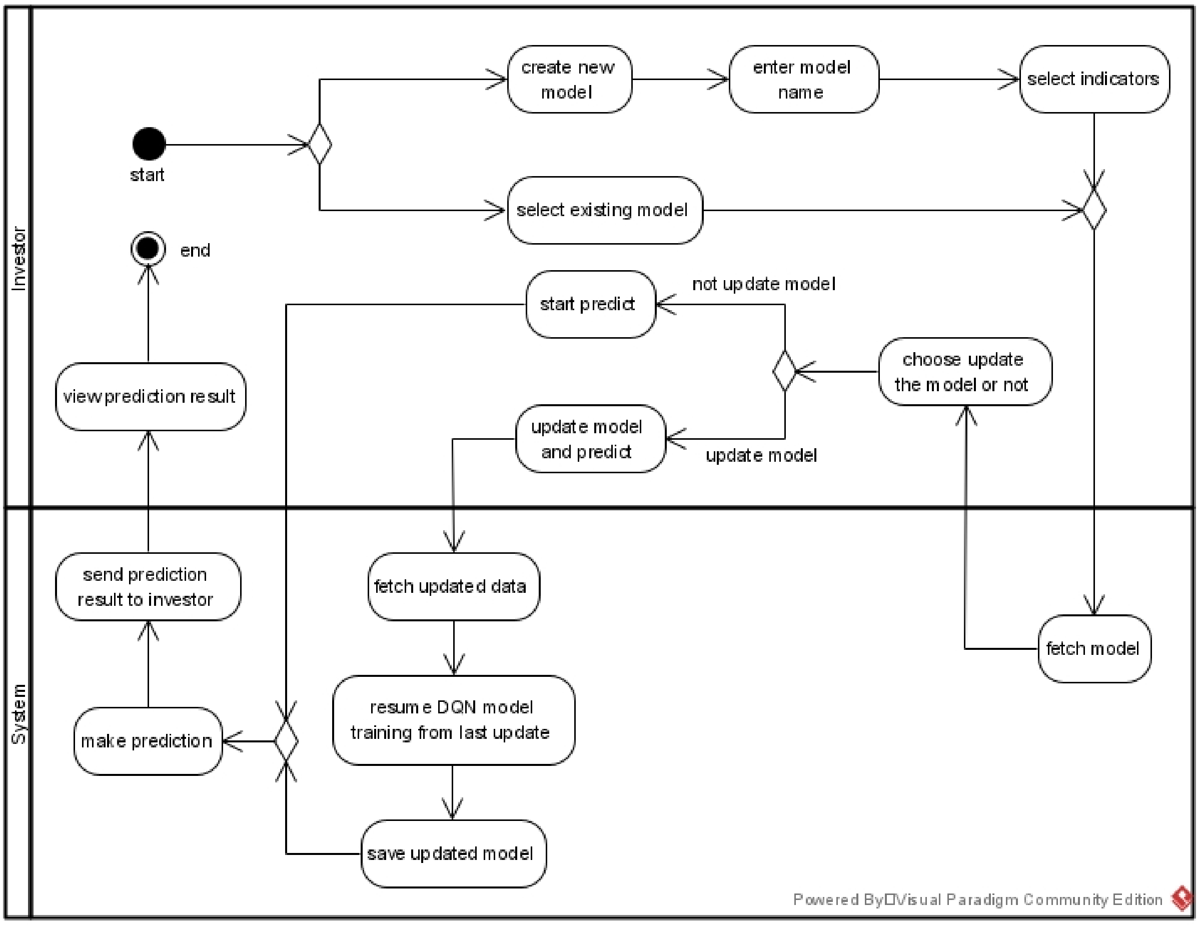


Figure 3.3.1 *The process of stock trend prediction procedure*

In the stock trend prediction procedure, investor can create new model. Investor needs to enter a name for the model and select indicators to be used by the prediction model during model creation. Investor also can select existing model to predict the stock trend. After that, the system will fetch the model from the server file system and return the model to investor.

Due to the model training from scratch required a long time, to reduce the waiting time of user, the model will be pretrained and stored in the server file system. When new stock data is available, the model can be retrieved from the server file system and resume the training from last update. Since the model is not trained from the scratch but just train on the new data, the user waiting time for the model can be reduced.

Investor can choose to update the model or not if the model is not up to date. If investor choose to not update the model, system will use that model to make the stock trend prediction. If investor choose to update the model, system will fetch updated stock data and use this data to resume DQN model training from last update. After the training is done, system will save the updated model to the server file system. Then, system will use the updated model to make prediction on stock trend. After the predict is done, system will return the prediction result to investor and investor can view the prediction result.

3.3.2 Design of DQN

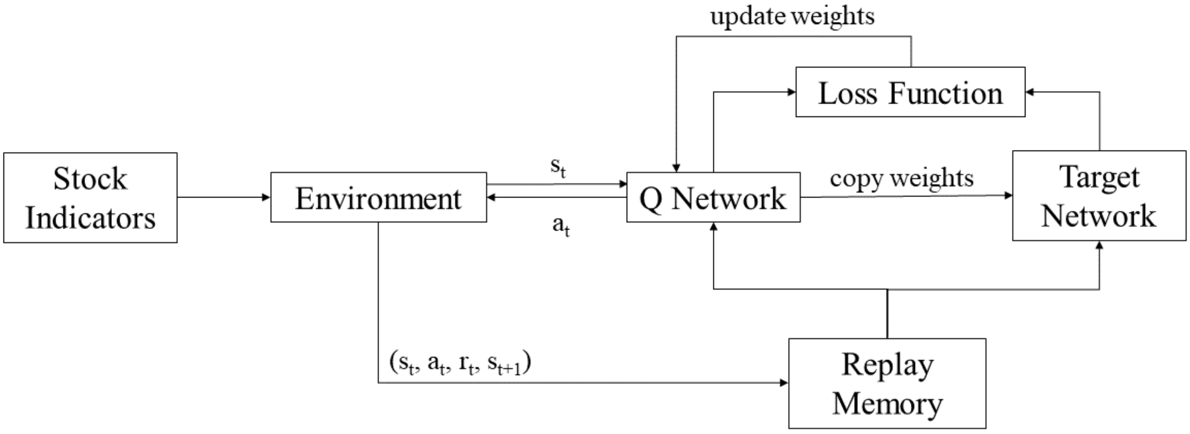


Figure 3.3.2 *Overview of the DQN*

The objectives of this project will be achieved by using Deep Q-Network (DQN) algorithm to develop the dynamic stock prediction model. Instead of tick data, the used of indicators as input is expected to reduce the training cost of the model. For simplicity, return of investment (ROI) will be used as the reward of DQN agent.

The environment will use the current state s_t to get action a_t from the Q Network. The state in this context is the stock indicators selected by user and the actions allowed for the DQN agent will be buy and sell. For a given state, the Q Network will return the predicted Q value for each action. The action with highest Q value will be selected and performed by the agent.

After the agent perform that action in the environment, the agent will receive reward r_t, and come to new state s_{t+1}. Information generated after every step (s_t, a_t, r_t, s_{t+1}) will be added to replay memory. The replay memory has a queue structure and will have minimum size of 1,000 and maximum size of 50,000. We will get a minibatch of random samples from the replay memory and use in the training of the network. The use of replay memory able to prevent local optimal problem [17] and makes the training more stable.

As show in Figure 3.3.2, 2 neural networks: Q Network and Target Q Network have been used in DQN. The use of 2 instead of 1 neural network in DQN is due to 1 neural network in DQN tends to overestimate the Q values and rise the estimation error as the steps taken increases [17]. By using Q Network to select action a_t based on current state s_t while using Target Q Network to retrieved maximum future Q value and updates the Q value of given state s_t, the overestimation problem can be overcome [17]. The Target Q Network is same with Q Network, however, unlike the weights of Q Network frequently get updated, the weights of Q Network will be copied to Target Q Network every 5 episodes. The structure of the neural network is illustrated in Figure 3.3.3 and discussed more in next section.

To ensure the agent get enough exploration of the environment before exploit the model, epsilon greedy strategy have been applied. Enough exploration able to prevent local optimal problem of the model. In every step, a random number between 0 and 1 will be produced. This random number will be compared to a number called epsilon. If the random number larger than epsilon, then the agent will take an action based on the predicted Q values get from model, else, the agent will take random action to explore the environment. After done an episode and before another episode start, the value of epsilon will be decayed.

Below, an example of how DQN working will be discussed. First, user need to provide stock symbol and indicators as input. For instance, user provide “1155.KL” as symbol and “SMA, RSI, OBV” as indicators. Then, this information will be used to retrieve stock data with the specified indicators. The stock data is a Pandas DataFrame with date as index and stock indicators as other columns. Let said the stock data start from 2019-04-25 and end with 2022-04-12, while value of SMA will be column 1, RSI as column 2, and OBV as column 3. Next, initialize the environment using the stock data. A DQN agent also been initialized, and epsilon is set to 1. For every episode, the episode reward is set to 0, episode step count is set to 1, episode done status is set to false, and current state of environment is initialized to initial state. The current state is the stock indicators values in a given window size. The window size is set to 21, thus the initial state will be the first 21 stock indicators value start with date 2019-04-25 to 2019-05-28. An array with 21 items, 1 item represent the value of SMA, RSI, and OBV in a day.

When episode not done, a random number between 0 and 1 will be compared to the epsilon. If the random number is larger than epsilon, agent will perform an action based on the predicted Q values get from model. Example of Q values is [0.02524731, 0.00415763, 0.00041585, 0.03465225, 0.02165215]. First position represent action "Strongly Sell", second position represent action "Sell", forth for "Hold", fifth for "Buy", and sixth for "Strongly Buy". Action with largest Q value will be selected, thus "Buy" action will be selected in this example. Agent will pass in the current state as input to the Q Network to get predicted Q values as output. Otherwise, if the random number is less than or equal to epsilon, then, a random action will be performed. After performed the action, we will get new state, reward, and new done status. The reward will be added to the episode reward while current state, action, reward, new state, and done status is appended to the replay memory. Next, train the Q Network if the size of replay memory is larger than or equal to the minimum size of replay memory. Otherwise, if the size of replay memory is smaller than the minimum size of replay memory, skip the train of the network. The minimum size of replay memory is set to 1,000. This means the first 1,000 steps will not be used to train the network but to collect enough sample in replay memory.

For the training of the network, a minibatch of random samples with size of 64 will be retrieved from replay memory. Then, the current states and new states will be extracted from the minibatch. Prepare the Q values of the current state (current Qs list) will be retrieved by predict using the Q Network while the Q values of the new state (future Qs list) will be retrieved by predict using the Target Q Network. For each item in minibatch, if the done state is false, get the maximum future Q value and *new Q value = reward + DISCOUNT * maximum future Q value*. If done state is true, then *new Q value = reward*. Then, update the Q value in current Q values which has position represent by the action to the new Q value. Append current state to X array and append current Q values to y array. After done for all item in minibatch, fit or train the Q Network using X and y array. Increase target update counter, if target update counter is 5, update the weight of Target Q Network by copy the weight of Q Network, then reset target update counter to 0.

After done the train of the network, assign the new state to current state and increase episode step by 1. Repeat the processes until episode done status is true. Then, decay the epsilon if epsilon is greater than the minimum epsilon, 0.01. The new epsilon can be retrieved by multiply current epsilon by epsilon decay of 0.999. Then, another episode will be started, and the process illustrated above will be repeated until finish all episode.

To predict using the model, the stock data will be loaded and used to initialize the environment. A DQN agent trained previously will also be loaded. Then, the current state of environment is initialized to initial state. When not done, the current state will be used to get Q values predicted by the trained Q Network and action with largest Q value will be performed. After performed the action, we will retrieve new state, reward, and new done status. Repeat the processes until done status is true, then, the action returned will be the recommended action.

3.3.3 Design of DQN network structure

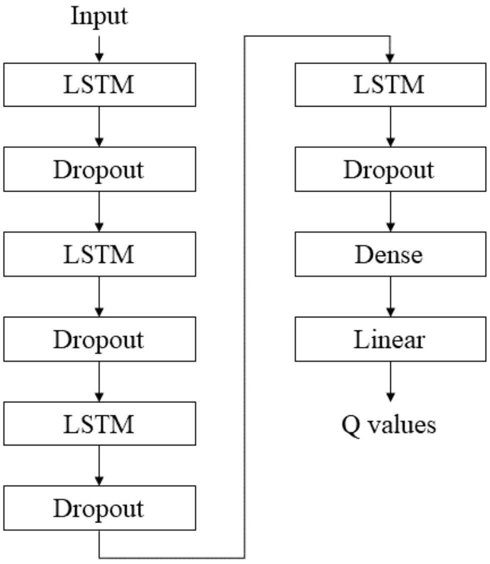


Figure 3.3.3 *The network structure of DQN*

Since stock data is sequential data and Recurrent Neural Networks (RNN) works better on sequential data compared to classical feedforward DNN, thus the feedforward DNN used in classical DQN have been replaced. However, RNN have short-term memory problem cause by vanishing gradient problem if the sequence is long enough. During the backpropagation in training, the gradient is calculated and used to update weights of the network. The small gradient of the previous layer will cause the gradient of current layer smaller. Small gradient brings small effect to the weight update and this cause the network not learning from earlier inputs [29]. To solve this problem, 2 specialized version of RNN have been created, they are

Gated Recurrent Unit (GRU) and Long Short-Term Memory (LSTM). Due to GRU have been used in the GDQN proposed by Wu et al. [17], therefore, in this project, LSTM will be used to replace the feedforward DNN used in classical DQN.

LSTM consists of cell state, forget gate, input gate, and output gate. Cell state act as the memory of the network, able to bring related information over the processing of the time series. Forget gate will be responsible to decide what information should keep from previous steps. Forget gate done this by passing previous hidden state and current input to a sigmoid function which transform a value to be between 0 and 1. The value closer to 0 tend to be forgotten while value closer to 1 tend to be kept. The input gate determines what information from the current input should be added. The next hidden state will be decided by output gate [30]. Both input gate and output gate consist of sigmoid function and tanh function which transform a value to be between -1 and 1.

This network will take the current state s_t as input and output the Q values $Q(s_t, a_t)$ as prediction. To address the problem of overfit, dropout regularization technique will be used. After each LSTM layer, a dropout of 20% is added. At the dense layer of the network, ReLU activation function will be used since [24] and [27] have shown the reliability of this activation function. Moreover, as [26] and [27] have shown promising achievement by using ADAM optimizer, thus ADAM optimization technique will be applied to adjust the parameters of the network in the DQN. Since Théate & Ernst [27] suggested that the use of Huber loss instead of mean squared error (MSE) loss in classical DQN able to improve the stability of the model training, therefore Huber loss will be used in the network as the loss function.

3.4 Hardware and Software Requirements

For software requirement, the development will perform under Windows 10 operating system. Visual Studio Code will be used as the IDE in the development of the system. The development of the web application will used HTML, CSS, and JavaScript. For better looking user interface, frontend library like Bootstrap or React will be used. Python will be used as the programming language to develop the predictive model. When develop the predictive model, TensorFlow an open source machine learning platform will be utilized due to it has a wide range of tools, libraries, and community resources. Then, the predictive model will integrate with the web application using TensorFlow.js. MySQL will be used as the DBMS to manage the database of the system.

For hardware requirement, due to Microsoft Azure will be used to host the web application, thus no hardware needed.

CHAPTER 4: SYSTEM DESIGN

4.1 Graphical User Interface Design

As shown in Figure 4.1.1 and Figure 4.1.2, user can create a username and password in the sign up page to register an account and use these information to login their account in login page.

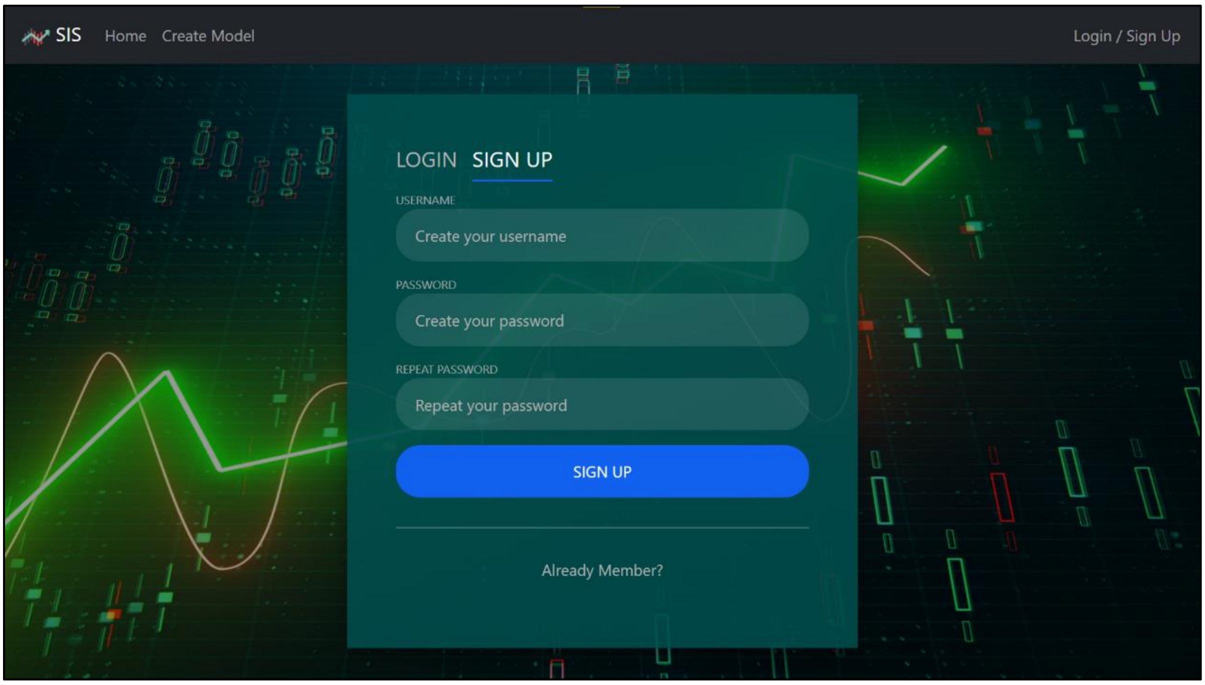


Figure 4.1.1 *Sign up page*

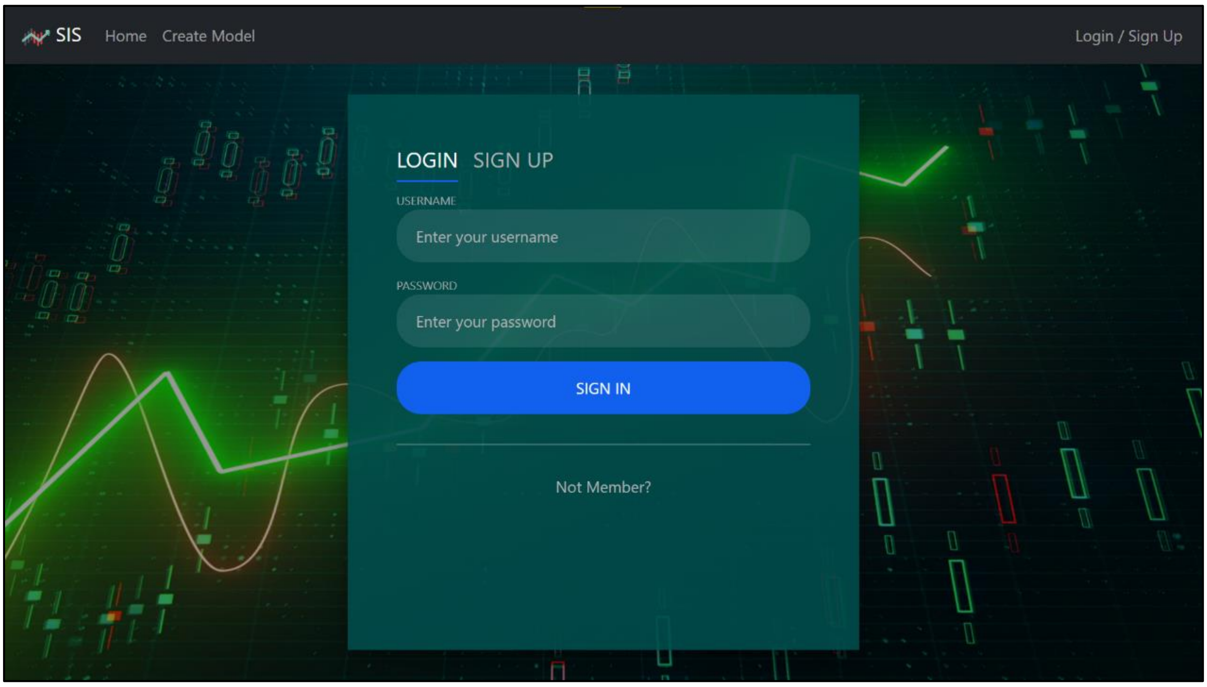
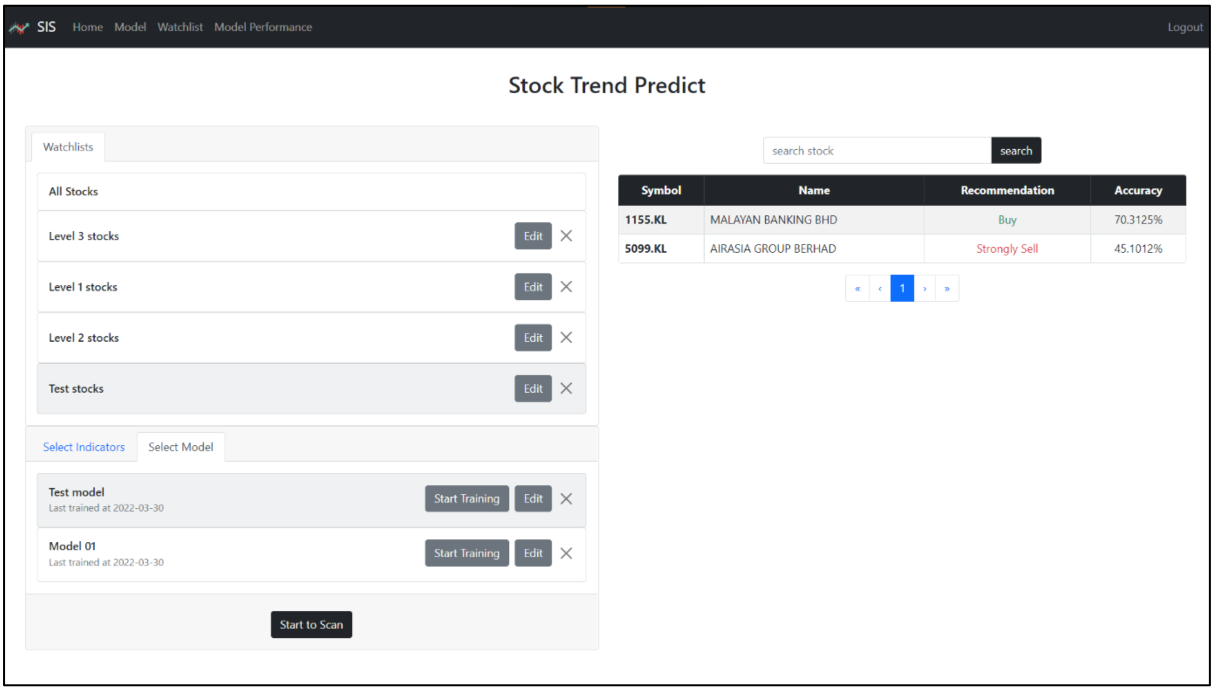


Figure 4.1.2 *Login page*

In the “Home” page, there will show the watchlists that user created previously. The “All Stocks” stock list which contain all 100 stocks will be selected as default. At the bottom, there have 2 tabs, that is “Select Indicators” tab and “Select Model” tab. User can go to “Select Model” tab and select the model they created previously to start making prediction. Other than that, user also can directly choose the indicators they would like to use to make the prediction without created a model upfront. At the right, the stocks consist in the watchlist will be display in a table. The table will display maximum 10 stocks at a time. To view other stocks, user can use the pagination function to navigate to other page. User also can use the search function to search the desired stock easily without going through all pages.

As illustrated in Figure 4.1.3, to make prediction, user first need to select the watchlist which consist of stocks user interested with. Then, user need to select indicators or select model which user would like to use in the prediction. After user click the “Start to Scan” button, prediction will start. After a while the prediction will done, the stock trend (strongly buy, buy, hold, sell, strongly sell) for each stock will be shown in the “Recommendation” column in the table and the accuracy of the prediction will be shown in the “Accuracy” column.



Stock Trend Predict

All Stocks

Level 3 stocks

Level 1 stocks

Level 2 stocks

Test stocks

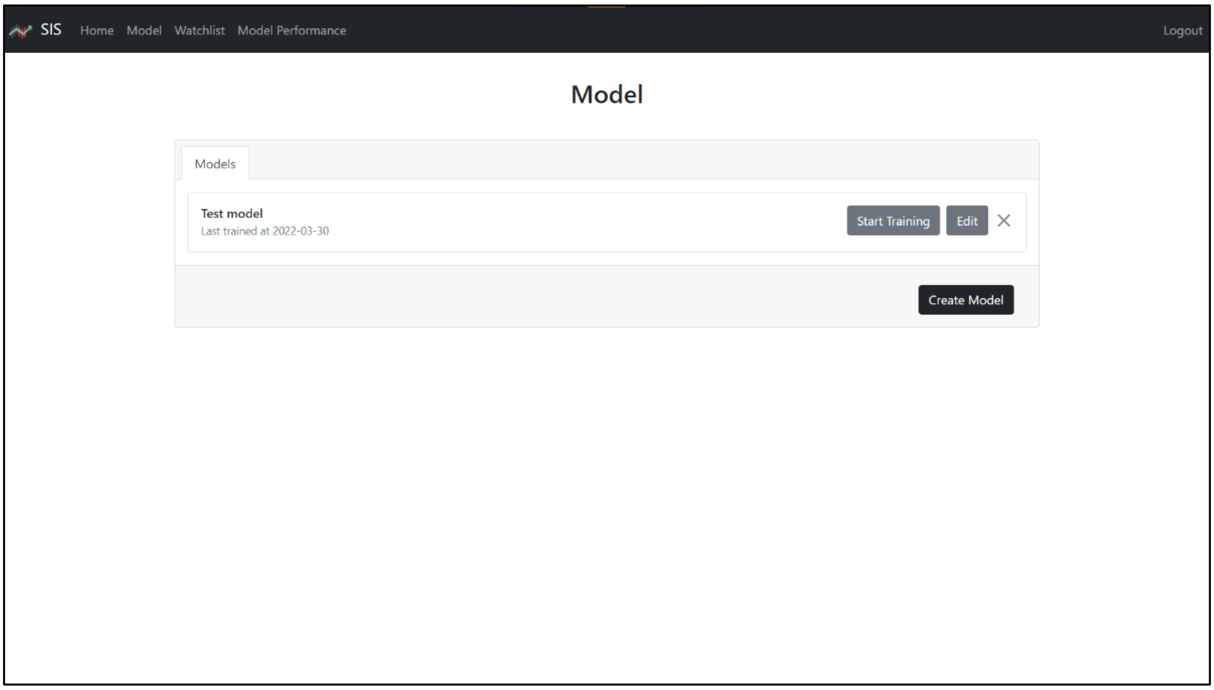
Test model
Last trained at 2022-03-30

Model 01
Last trained at 2022-03-30

Symbol	Name	Recommendation	Accuracy
1155.KL	MALAYAN BANKING BHD	Buy	70.3125%
5099.KL	AIRASIA GROUP BERHAD	Strongly Sell	45.1012%

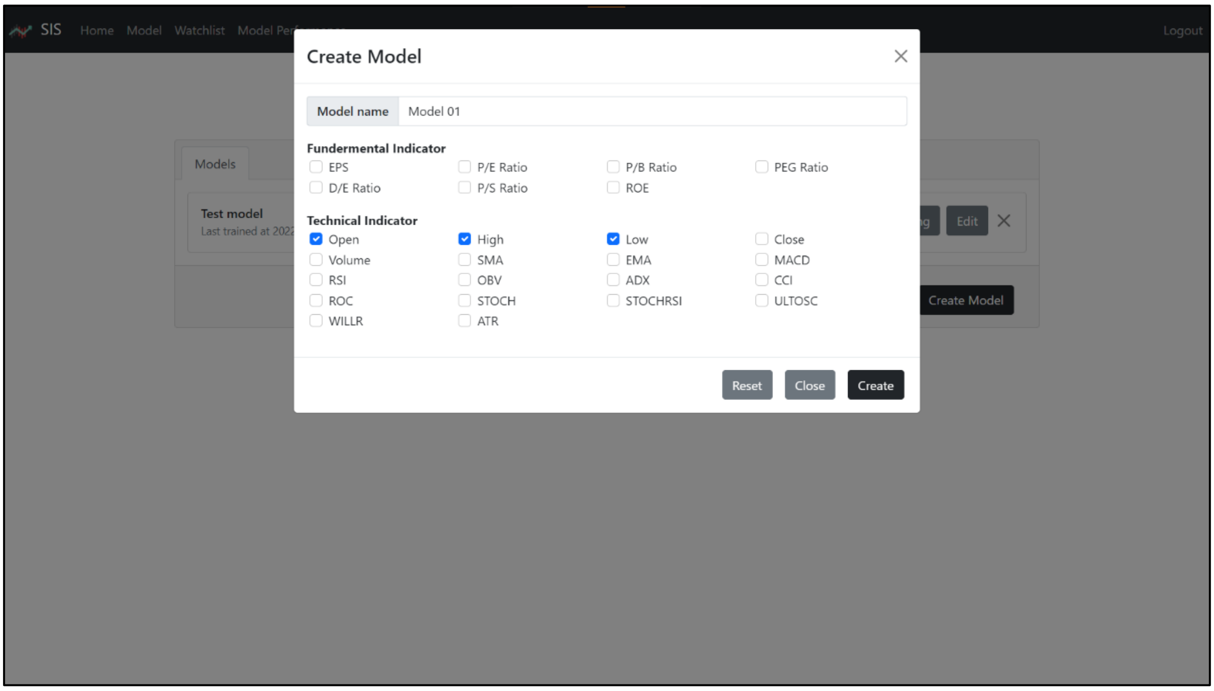
Figure 4.1.3 *Home page*

In the “Model” page, the models user created previously will be shown as illustrated in Figure 4.1.4. If user have not created any model, “No Model Found” will be shown. User can create a model by click the “Create Model” button. Then, as shown in Figure 4.1.5, a modal will occur for user to provide a name for the model and select the indicators used in the model. The indicators are divided into fundamental indicator and technical indicator. After user click "Create" button and the model will be created. Besides, for the models that are not trained using latest stock data (not up to date), user can click the "Start Training" button to train (update) the model using latest stock data. Besides, user can edit the existing model by click the "Edit" button beside the model listed. As shown in Figure 4.1.6, a modal will appear and user can change the model name and indicator consist in the model. User also can remove the model by click the cross button beside the model.



Test model
Last trained at 2022-03-30

Figure 4.1.4 *Model page (Part 1)*



Create Model

Model name: Model 01

Fundermental Indicator

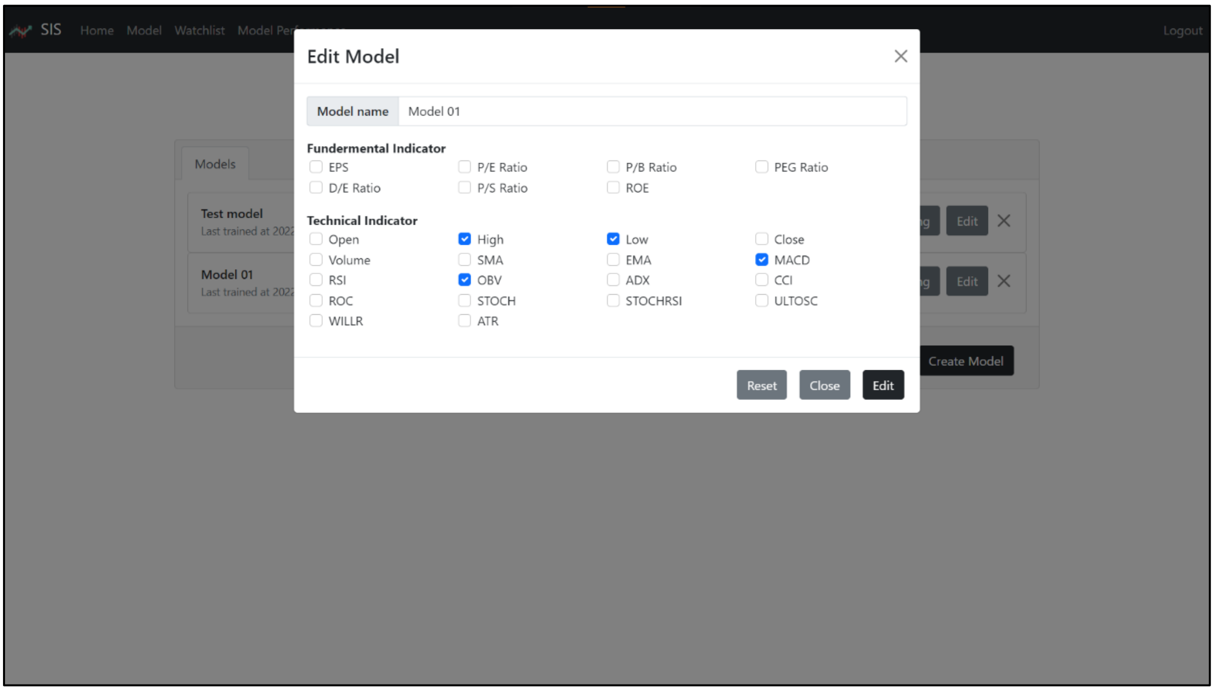
☐ EPS ☐ P/E Ratio ☐ P/B Ratio ☐ PEG Ratio
☐ D/E Ratio ☐ P/S Ratio ☐ ROE

Technical Indicator

☑ Open ☑ High ☑ Low ☐ Close
☐ Volume ☐ SMA ☐ EMA ☐ MACD
☐ RSI ☐ OBV ☐ ADX ☐ CCI
☐ ROC ☐ STOCH ☐ STOCHRSI ☐ ULTOSC
☐ WILLR ☐ ATR

Reset Close Create

***Figure 4.1.5** Model page (Part 2)*



Edit Model

Model name: Model 01

Fundermental Indicator

☐ EPS ☐ P/E Ratio ☐ P/B Ratio ☐ PEG Ratio
☐ D/E Ratio ☐ P/S Ratio ☐ ROE

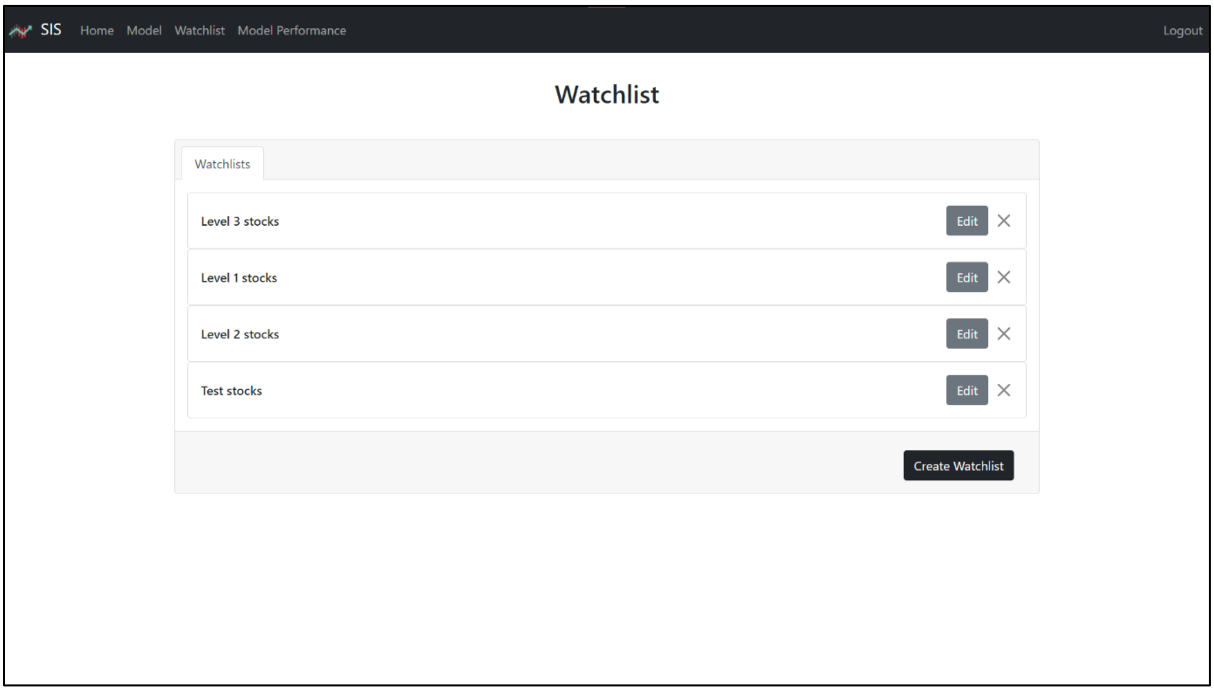
Technical Indicator

☐ Open ☑ High ☑ Low ☐ Close
☐ Volume ☐ SMA ☐ EMA ☑ MACD
☐ RSI ☑ OBV ☐ ADX ☐ CCI
☐ ROC ☐ STOCH ☐ STOCHRSI ☐ ULTOSC
☐ WILLR ☐ ATR

Reset Close Edit

***Figure 4.1.6** Model page (Part 3)*

In the “Watchlist” page, the watchlists user created previously will be displayed. If user did not create any watchlist before, “No Watchlist Found” will be displayed. To create a watchlist, user can click the “Create Watchlist” button. A modal will be displayed for user to input the watchlist name and select the stocks they would like to add into the watchlist. Other than that, user also can click the "Edit" button to edit the watchlist created. User can change the watchlist name and the stocks consist in the watchlist. To remove a watchlist, user can click the cross button beside the watchlist.



Watchlist

Watchlists

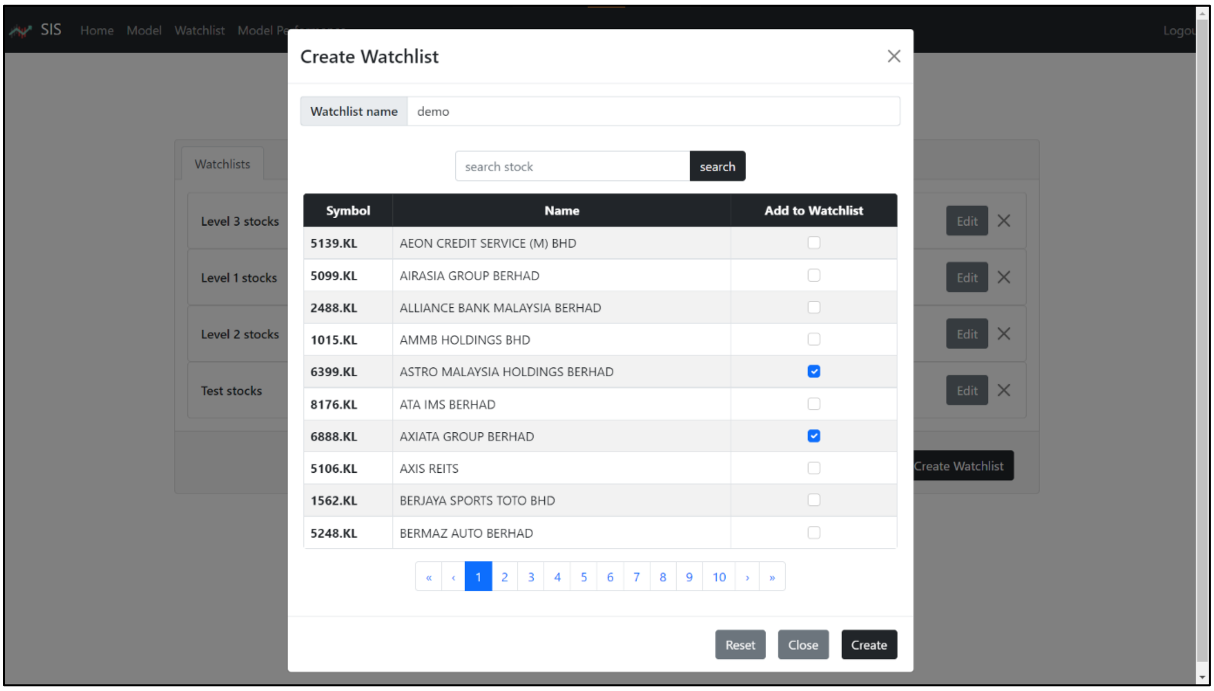
Level 3 stocks Edit ×

Level 1 stocks Edit ×

Level 2 stocks Edit ×

Test stocks Edit ×

Figure 4.1.7 *Watchlist page (Part 1)*



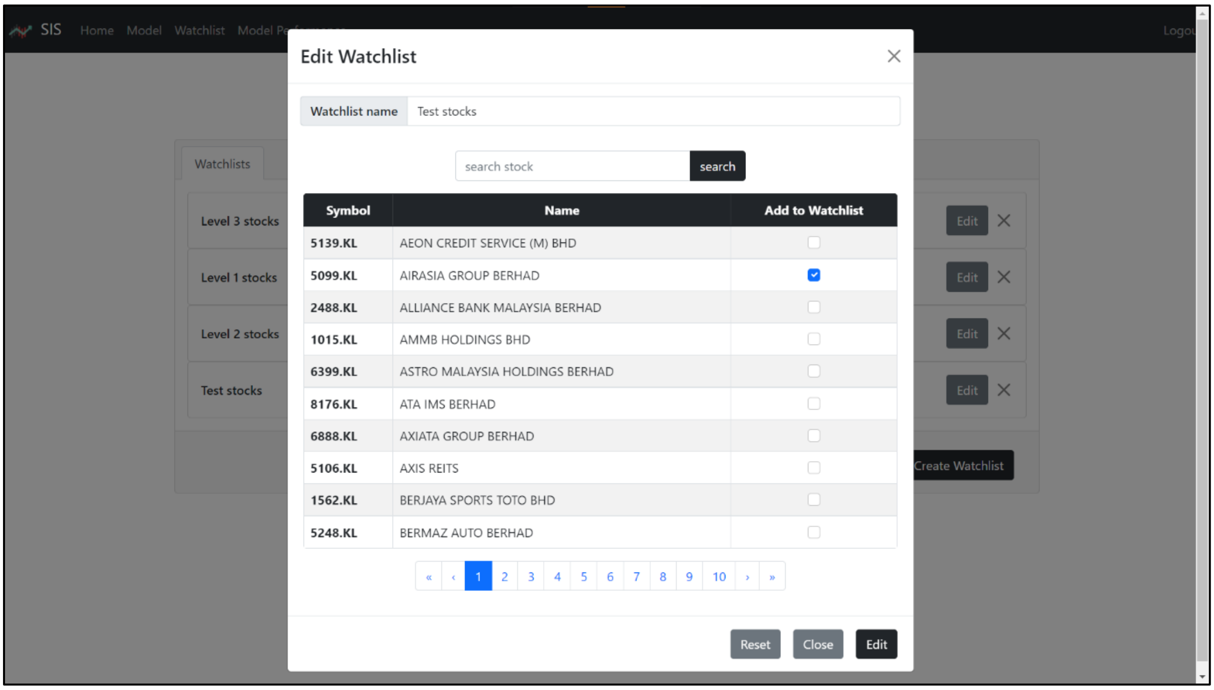
Create Watchlist ×

Watchlist name: demo

search stock | search

Symbol	Name	Add to Watchlist
5139.KL	AEON CREDIT SERVICE (M) BHD	□
5099.KL	AIRASIA GROUP BERHAD	□
2488.KL	ALLIANCE BANK MALAYSIA BERHAD	□
1015.KL	AMMB HOLDINGS BHD	□
6399.KL	ASTRO MALAYSIA HOLDINGS BERHAD	☑
8176.KL	ATA IMS BERHAD	□
6888.KL	AXIATA GROUP BERHAD	☑
5106.KL	AXIS REITS	□
1562.KL	BERJAYA SPORTS TOTO BHD	□
5248.KL	BERMAZ AUTO BERHAD	□

Figure 4.1.8 *Watchlist page (Part 2)*



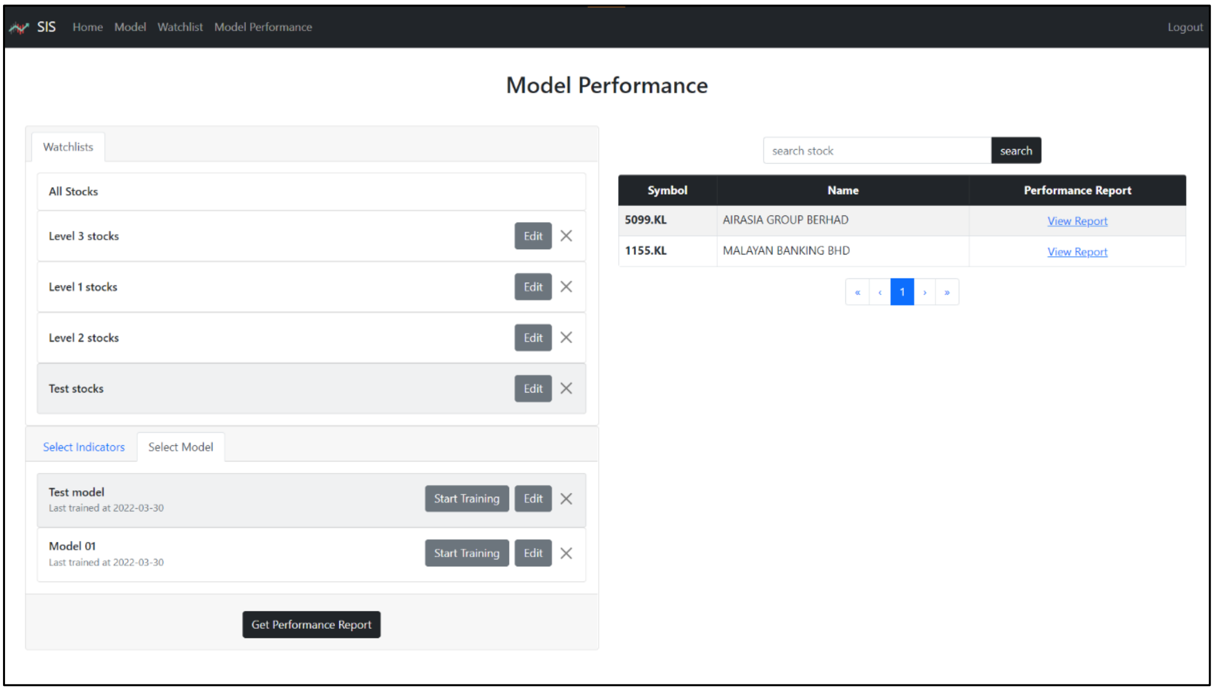
Watchlist name: Test stocks

search stock

Symbol	Name	Add to Watchlist
5139.KL	AEON CREDIT SERVICE (M) BHD	□
5099.KL	AIRASIA GROUP BERHAD	☑
2488.KL	ALLIANCE BANK MALAYSIA BERHAD	□
1015.KL	AMMB HOLDINGS BHD	□
6399.KL	ASTRO MALAYSIA HOLDINGS BERHAD	□
8176.KL	ATA IMS BERHAD	□
6888.KL	AXIATA GROUP BERHAD	□
5106.KL	AXIS REITS	□
1562.KL	BERJAYA SPORTS TOTO BHD	□
5248.KL	BERMAZ AUTO BERHAD	□

***Figure 4.1.9** Watchlist page (Part 3)*

In the “Model Performance” page, the design is same as “Home” page except the “Recommendation” column and “Accuracy” column is replaced by “Performance Report” column while the “Start to Scan” button is replaced by “Get Performance Report” button. After user select a watchlist, select indicators or model and click “Get Performance Report” button, a report will be generated for each stock. User can access the report by click the “View Report” button at the “Performance Report” column after the report generation is done as shown in Figure 4.1.10. At the left of the report generated, some chart like cumulative returns, EOY returns, monthly returns, daily returns rolling volatility, and so on will be display. At the right of the report, some key performance metrics will be shown. The report generated is shown in Figure 4.1.11 to Figure 4.4.13.



Model Performance

All Stocks

Level 3 stocks

Level 1 stocks

Level 2 stocks

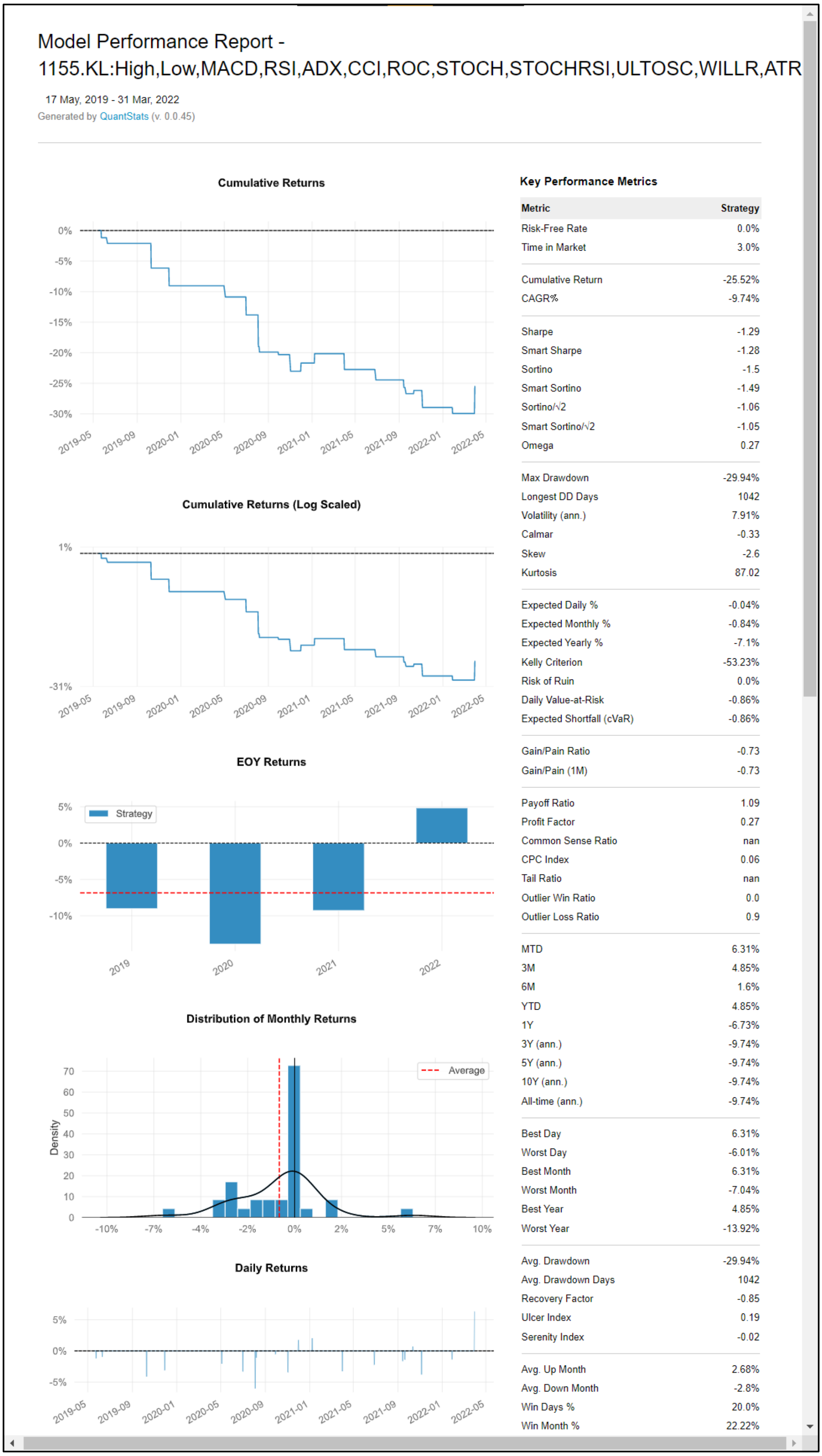
Test stocks

Test model
Last trained at 2022-03-30

Model 01
Last trained at 2022-03-30

Symbol	Name	Performance Report
5099.KL	AIRASIA GROUP BERHAD	View Report
1155.KL	MALAYAN BANKING BHD	View Report

Figure 4.1.10 *Model Performance page*



Model Performance Report - 1155.KL:High,Low,MACD,RSI,ADX,CCI,ROC,STOCH,STOCHRSI,ULTOSC,WILLR,ATR

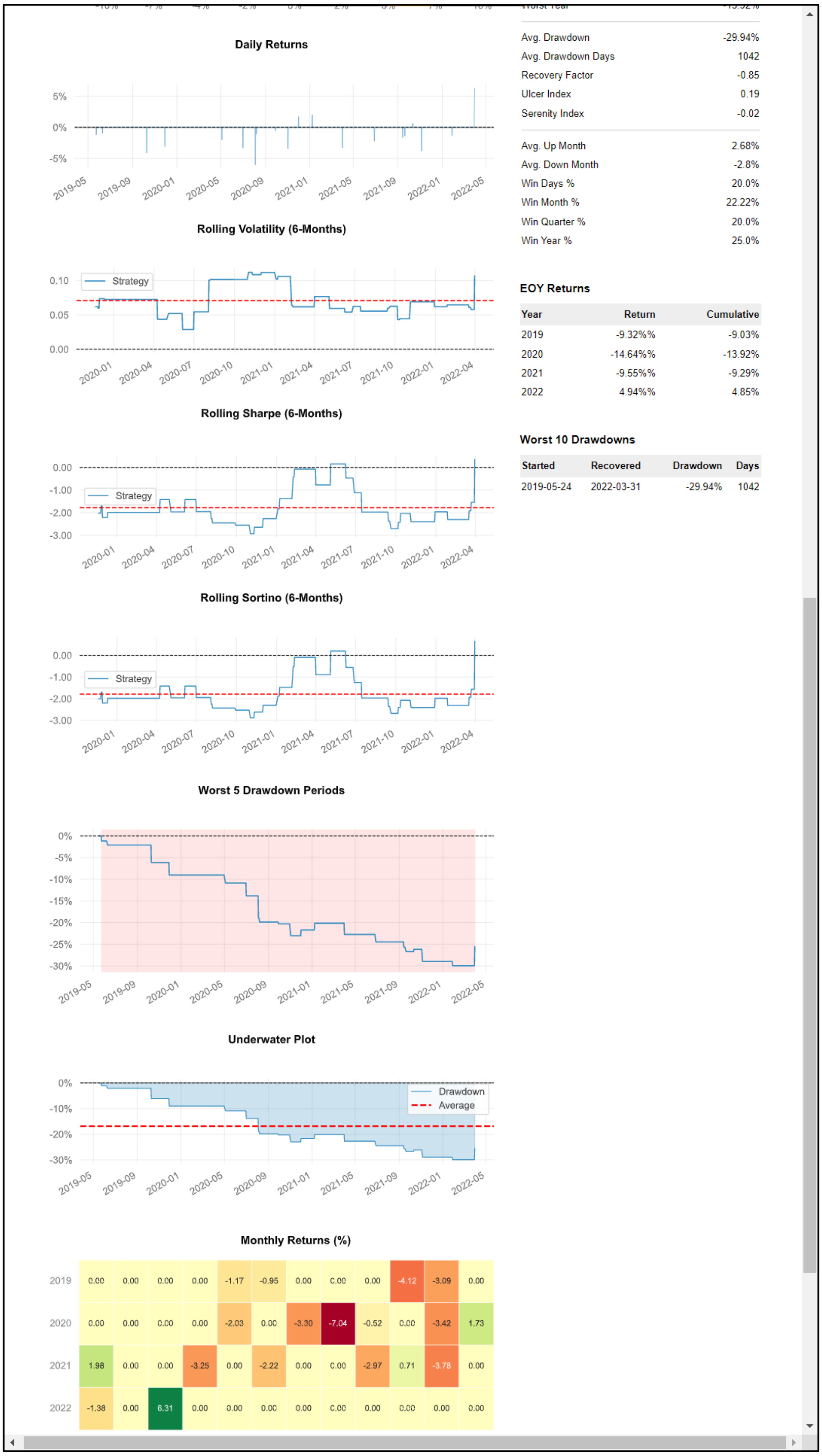
17 May, 2019 - 31 Mar, 2022

Generated by QuantStats (v. 0.0.45)

Key Performance Metrics

Metric	Strategy
Risk-Free Rate	0.0%
Time in Market	3.0%
Cumulative Return	-25.52%
CAGR﹪	-9.74%
Sharpe	-1.29
Smart Sharpe	-1.28
Sortino	-1.5
Smart Sortino	-1.49
Sortino/√2	-1.06
Smart Sortino/√2	-1.05
Omega	0.27
Max Drawdown	-29.94%
Longest DD Days	1042
Volatility (ann.)	7.91%
Calmar	-0.33
Skew	-2.6
Kurtosis	87.02
Expected Daily %	-0.04%
Expected Monthly %	-0.84%
Expected Yearly %	-7.1%
Kelly Criterion	-53.23%
Risk of Ruin	0.0%
Daily Value-at-Risk	-0.86%
Expected Shortfall (cVaR)	-0.86%
Gain/Pain Ratio	-0.73
Gain/Pain (1M)	-0.73
Payoff Ratio	1.09
Profit Factor	0.27
Common Sense Ratio	nan
CPC Index	0.06
Tail Ratio	nan
Outlier Win Ratio	0.0
Outlier Loss Ratio	0.9
MTD	6.31%
3M	4.85%
6M	1.6%
YTD	4.85%
1Y	-6.73%
3Y (ann.)	-9.74%
5Y (ann.)	-9.74%
10Y (ann.)	-9.74%
All-time (ann.)	-9.74%
Best Day	6.31%
Worst Day	-6.01%
Best Month	6.31%
Worst Month	-7.04%
Best Year	4.85%
Worst Year	-13.92%
Avg. Drawdown	-29.94%
Avg. Drawdown Days	1042
Recovery Factor	-0.85
Ulcer Index	0.19
Serenity Index	-0.02
Avg. Up Month	2.68%
Avg. Down Month	-2.8%
Win Days %	20.0%
Win Month %	22.22%

Figure 4.1.11 *Model Performance Report (Part 1)*



Daily Returns

Rolling Volatility (6-Months)

Rolling Sharpe (6-Months)

Rolling Sortino (6-Months)

Worst 5 Drawdown Periods

Underwater Plot

Monthly Returns (%)

2019	0.00	0.00	0.00	0.00	-1.17	-0.95	0.00	0.00	0.00	-4.12	-3.09	0.00
2020	0.00	0.00	0.00	0.00	-2.03	0.00	-3.30	-7.04	-0.52	0.00	-3.42	1.73
2021	1.98	0.00	0.00	-3.25	0.00	-2.22	0.00	0.00	-2.97	0.71	-3.78	0.00
2022	-1.38	0.00	6.31	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00

Avg. Drawdown	-29.94%
Avg. Drawdown Days	1042
Recovery Factor	-0.85
Ulcer Index	0.19
Serenity Index	-0.02
Avg. Up Month	2.68%
Avg. Down Month	-2.8%
Win Days %	20.0%
Win Month %	22.22%
Win Quarter %	20.0%
Win Year %	25.0%

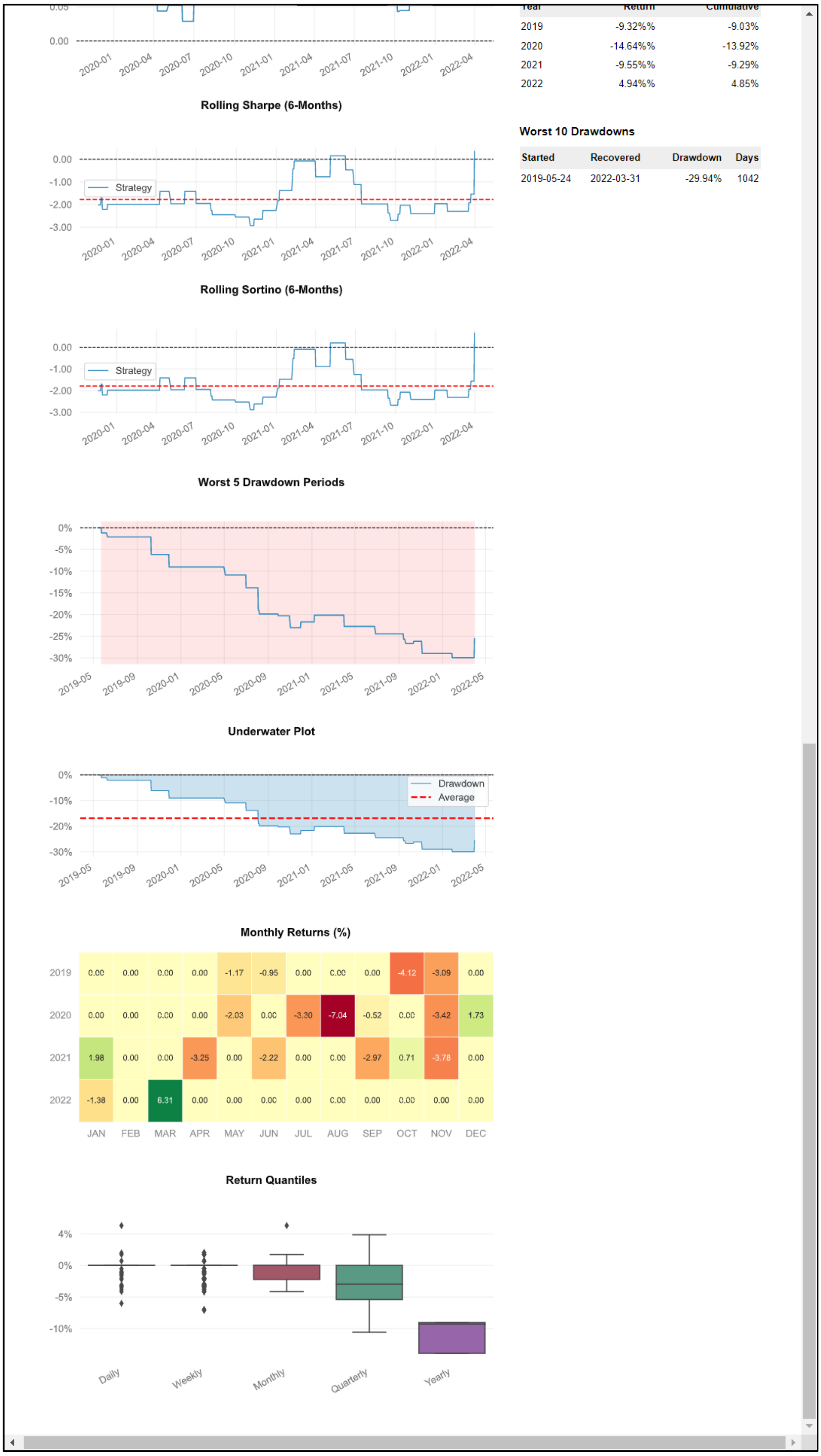
EOY Returns

Year	Return	Cumulative
2019	-9.32%%	-9.03%
2020	-14.64%%	-13.92%
2021	-9.55%%	-9.29%
2022	4.94%%	4.85%

Worst 10 Drawdowns

Started	Recovered	Drawdown	Days
2019-05-24	2022-03-31	-29.94%	1042

***Figure 4.1.12** Model Performance Report (Part 2)*



Year	Return	Cumulative
2019	-9.32%%	-9.03%
2020	-14.64%%	-13.92%
2021	-9.55%%	-9.29%
2022	4.94%%	4.85%

Worst 10 Drawdowns

Started	Recovered	Drawdown	Days
2019-05-24	2022-03-31	-29.94%	1042

Monthly Returns (%)

	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	SEP	OCT	NOV	DEC
2019	0.00	0.00	0.00	0.00	-1.17	-0.95	0.00	0.00	0.00	-4.12	-3.09	0.00
2020	0.00	0.00	0.00	0.00	-2.03	0.00	-3.30	-7.04	-0.52	0.00	-3.42	1.73
2021	1.98	0.00	0.00	-3.25	0.00	-2.22	0.00	0.00	-2.97	0.71	-3.78	0.00
2022	-1.38	0.00	6.31	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00

Figure 4.1.13 *Model Performance Report (Part 3)*

4.2 Data Storage Design

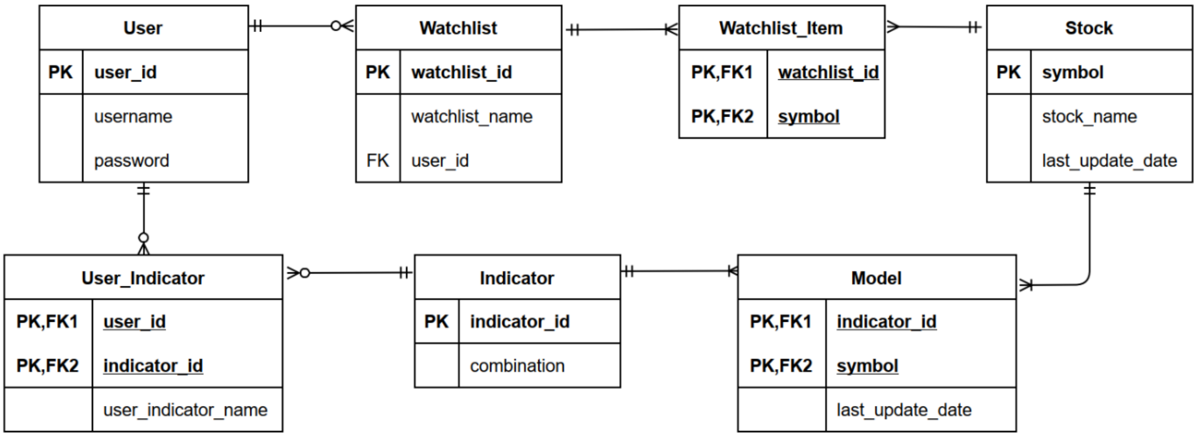


Figure 4.2 *ERD of Stock Indicator Scanner Customization Tool*

In this system, main information required to store including user, indicator, stock, model, and watchlist. In the database, user table will be used to store username and hashed password to authenticate the user. The primary key of user table is user_id which will auto generate by the DBMS. The model's name given by investor during model creation will be stored as user_indicator_name under user_indicator table while the combination of indicators investor select during model creation will be stored as combination under indicator table. Every combination of indicators will be identified by indicator_id.

In stock table, symbol of the stock will be the primary key. Name of the stock, and last update date of the stock data will store under stock table too. By using indicator_id and symbol as composite key for model table, DQN model for different stock using different combination of indicators can be identified. Model table will also store last update date of the model. Then, the watchlist table is used to store the watchlist user created. The watchlist_id which act as unique identifier will be auto generated by the DBMS while watchlist_name is the name given by user for the watchlist created. User_id in watchlist table act as foreign key to identify the owner of the watchlist. The watchlist_item table will store the stocks consist in the watchlist by using composite key of watchlist_id and symbol.

In the server file system, the model data will be stored in a directory named by the stock symbol. This directory will locate under directory named by indicator_id. For example, the DQN model for stock 5099.KL predict using MACD, RSI, OBV with 5 as indicator_id will have path 5/ 5099.KL. The directory and file consist in this directory is show in the table below.

Name	Type	Description
model	Directory	Store the model (Q Network) of the DQN
target_model	Directory	Store the target model (Target Network) of the DQN
agent_data	File	Store the information of DQN agent
env_data	File	Store the environment information of the DQN
stats_data	File	Store statistic data of the DQN

***Table 4.2** Directory and file consist in model data directory*

4.3 System Architecture Design

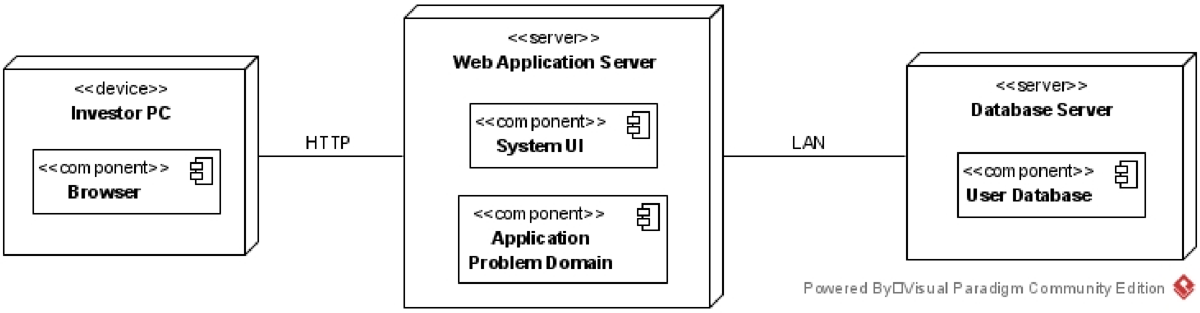


***Figure 4.3** Architecture of Stock Indicator Scanner Customization Tool*

Stock Indicator Scanner Customization Tool will be implemented as a web application because it brings a lot of convenience. As a web application, no install required for user and the system is able to work in various devices and platforms. The system will apply server-based architecture where all the software components such as data storage, data access logic, application logic, and presentation logic will be handled by the server. User required a browser and internet to access the server and use the service provided by the system.

Microsoft Azure will be used as web server to host the web application. The web application server will handle the UI of the system and the application logic. To avoid computing resource of client been used intensively during model training, Azure VM will be setup for model training. After the model training done in the web server, the results will be sent to client. On the other hand, the storage service provide by Microsoft Azure will be used as a cloud storage to store the user information (username and password), indicators user created, models user saved, stock trend prediction, and model evaluation result.

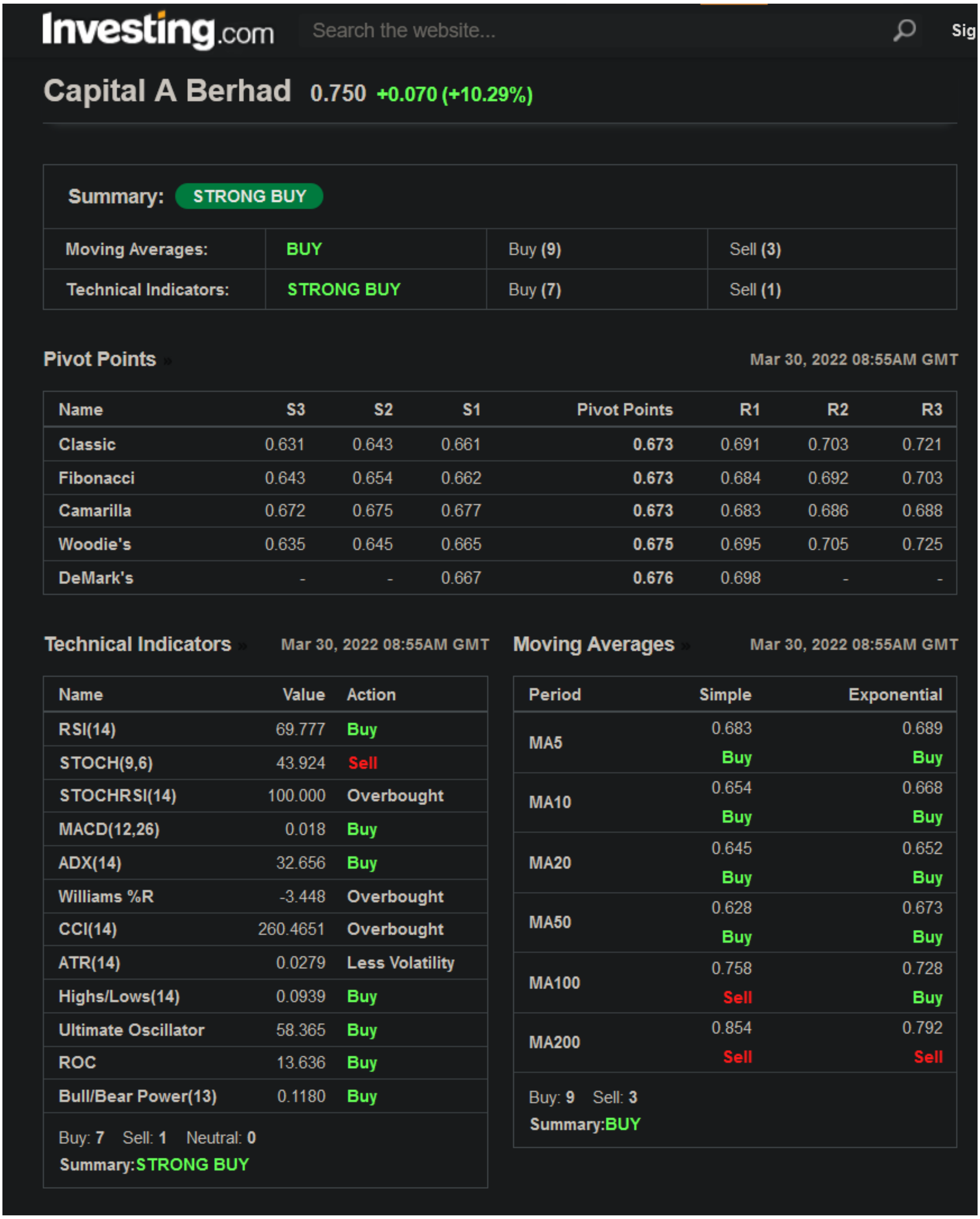
CHAPTER 5: SYSTEM TESTING

5.1 Project Verification Plans

Objective 1 of this project will be verify using experiment. In the experiment, Buy-and-Hold strategy will be compared with the proposed model on 2 stocks in the real stock market. Instead of live trading, paper trading, a simulated trading which record all trades will be used to conduct the experiment. This method allows us to evaluate the performance of different trading strategies without risking real money. At the beginning of the experiment, each trading strategy will have initial capital of RM10,000 and the transaction fee will be calculated too. The paper trading will be performed using pass 2 years of stock data. After paper trade, each trading strategies will be evaluated based on maximum drawdown, maximum drawdown duration, Sortino ratio and profit & loss. If these performance metrics showing good value, then this illustrate that the system enable user to gain better profit and risk management in the investment.

For objective 2, the prediction results able to assist investor in decision making during trading also can be verify by compare the prediction made the system and the prediction made by Investing.com based on some technical indicators. If most predictions are same or similar between both systems, then this will indicate that the prediction result of the system able to help investor to make decision when trading just like the prediction provided by Investing.com. To ensure the prediction result between both system is similar, other than the Bull/Bear Power indicator which is not supported, the proposed system will apply the indicators used by Investing.com to create the model.

Figure 5.1 shows the technical page of stock “Capital A Berhad” in Investing.com. From the figure, we can know how Investing.com use the technical indicators and moving averages to make prediction. Investing.com first calculate the value of a technical indicator or moving average, then suggest an action based on the value calculated. The action suggested by most of the indicators will become the final prediction result for technical indicators part. If the number of indicators that suggest “Buy” is far more than the number of “Sell”, then “Strong Buy” will be the final prediction result and vice versa for “Sell”. It is same for the part of moving averages. Investing.com will then combine both technical indicators part and moving averages part to come out the final suggestion. Due to the proposed system does not support moving average with various period, therefore only the prediction result from technical indicators part will be used to compared with the forecast from the proposed system.



Capital A Berhad 0.750 +0.070 (+10.29%)

Summary: STRONG BUY

Moving Averages:	BUY	Buy (9)	Sell (3)
Technical Indicators:	STRONG BUY	Buy (7)	Sell (1)

Pivot Points » Mar 30, 2022 08:55AM GMT

Name	S3	S2	S1	Pivot Points	R1	R2	R3
Classic	0.631	0.643	0.661	**0.673**	0.691	0.703	0.721
Fibonacci	0.643	0.654	0.662	**0.673**	0.684	0.692	0.703
Camarilla	0.672	0.675	0.677	**0.673**	0.683	0.686	0.688
Woodie's	0.635	0.645	0.665	**0.675**	0.695	0.705	0.725
DeMark's	-	-	0.667	**0.676**	0.698	-	-

Technical Indicators » Mar 30, 2022 08:55AM GMT

Name	Value	Action
RSI(14)	69.777	Buy
STOCH(9,6)	43.924	Sell
STOCHRSI(14)	100.000	Overbought
MACD(12,26)	0.018	Buy
ADX(14)	32.656	Buy
Williams %R	-3.448	Overbought
CCI(14)	260.4651	Overbought
ATR(14)	0.0279	Less Volatility
Highs/Lows(14)	0.0939	Buy
Ultimate Oscillator	58.365	Buy
ROC	13.636	Buy
Bull/Bear Power(13)	0.1180	Buy

Buy: 7 Sell: 1 Neutral: 0
Summary:STRONG BUY

Moving Averages » Mar 30, 2022 08:55AM GMT

Period	Simple	Exponential
MA5	0.683 Buy	0.689 Buy
MA10	0.654 Buy	0.668 Buy
MA20	0.645 Buy	0.652 Buy
MA50	0.628 Buy	0.673 Buy
MA100	0.758 Sell	0.728 Buy
MA200	0.854 Sell	0.792 Sell

Buy: 9 Sell: 3
Summary:BUY

Figure 5.1 *Technical page of stock "Capital A Berhad" in Investing.com in 30 Mar 2022*

5.2 Test Results

Below are the results of experiment conducted using stock data between 10 Jun 2019 to 21 April 2020 on symbol 1155.KL (Malayan Banking Bhd) and symbol 5099.KL (Capital A Bhd) using Buy-and-Hold strategy and the proposed model. Table 5.2.1 compared the performance metrics between both strategy on symbol 1155.KL while Table 5.2.2 on symbol 5099.KL.

From Table 5.2.1, we can observe that the maximum drawdown of the proposed model is far lower than the Buy-and-Hold strategy on symbol 1155.KL. This indicate that the investment losses of proposed model are lesser than the Buy-and-Hold strategy. However, the maximum drawdown duration of the proposed model is higher than the Buy-and-Hold strategy. Although the profit & loss metric shown that the profit of proposed model is lesser than Buy-and-Hold strategy, but the Sortino ratio of the proposed model is higher than Buy-and-Hold strategy. This shown that the proposed model is able to earn returns more efficient compared to Buy-and-Hold strategy.

Performance Metrics	**Strategy**	
	Buy-and-Hold	**Proposed model**
maximum drawdown (%)	-19.94	-7.26
maximum drawdown duration (days)	395	644
Sortino ratio	0.69	1.51
profit & loss (%)	21.4	16.47

Table 5.2.1 *Performance metrics of strategies on symbol 1155.KL*

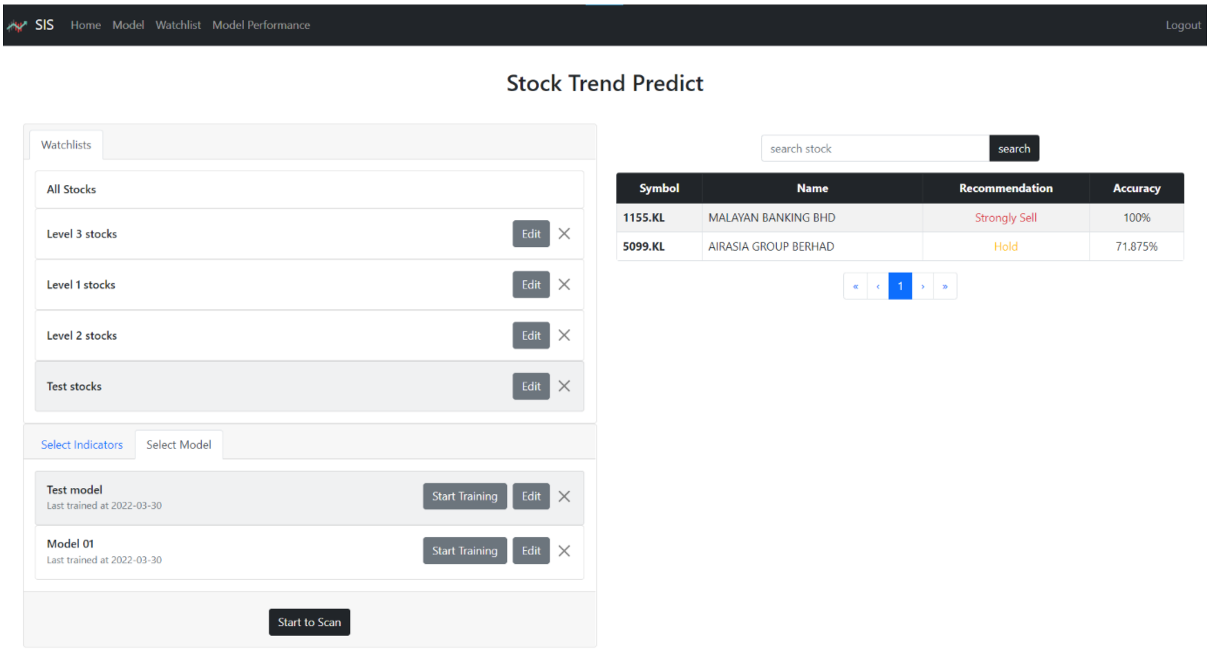
For symbol 5099.KL, we can notice that the maximum drawdown of the proposed model is lower than the Buy-and-Hold strategy from Table 5.2.2. This indicate that the losses of investment cause by the proposed model are lesser compared to Buy-and-Hold strategy. Moreover, the maximum drawdown duration of the proposed model is lesser than the Buy-and-Hold strategy too. Although the profit & loss metric shown that both strategies were losing, but the loss of the proposed model not only lesser than the Buy-and-Hold strategy, the Sortino ratio of the proposed model is also higher than Buy-and-Hold strategy. This indicated that the proposed model can earn more returns for every unit of negative risk compared to Buy-and-Hold strategy. For more detail about the performance of the strategies on both symbols, refer to Appendix B.

Performance Metrics	**Strategy**	
	Buy-and-Hold	**Proposed model**
maximum drawdown (%)	-73.47	-58.53
maximum drawdown duration (days)	994	877
Sortino ratio	0.24	0.35
profit & loss (%)	-35.49	-15.31

Table 5.2.2 *Performance metrics of strategies on symbol 5099.KL*

On the other hand, as shown in Figure 5.2.1, the proposed model recommended "Strongly Sell" on symbol 1155.KL (Malayan Banking Bhd) which is same with the prediction Investing.com made based on the technical indicators as shown in Figure 5.2.3. For symbol 5099.KL (Capital A Bhd), the proposed model recommended "Hold" while Investing.com suggested "Sell" as shown in Figure 5.2.1 and Figure 5.2.4. Although the prediction result for symbol 5099.KL is different between 2 system, however, the differ between both prediction result is not huge. From Figure 5.2.4 we can notice that there have 5 indicators suggest "Sell", and 3 indicators suggest "Buy". If change the suggestion of an indicator from "Sell" to "Buy", then the number of both "Sell" and "Buy" will be the same and the final suggestion will be "Neutral".

The prediction result for symbol 5099.KL is different between the proposed model and Investing.com might be due to Investing.com has considering Bull/Bear Power indicator while the proposed model does not include this indicator as illustrate in Figure 5.2.2. Another possible reason that causes this difference is the prediction method. Investing.com suggest the trading action by follow the prediction made by majority indicators while the proposed model using DQN which considering all selected indicators as one. Other than that, as shown in Figure 5.2.1, the accuracy of the model is quite low, which is only 71.875%. This might indicate that more training for the model is needed to achieve higher accuracy and better prediction.



Stock Trend Predict

All Stocks

Level 3 stocks

Level 1 stocks

Level 2 stocks

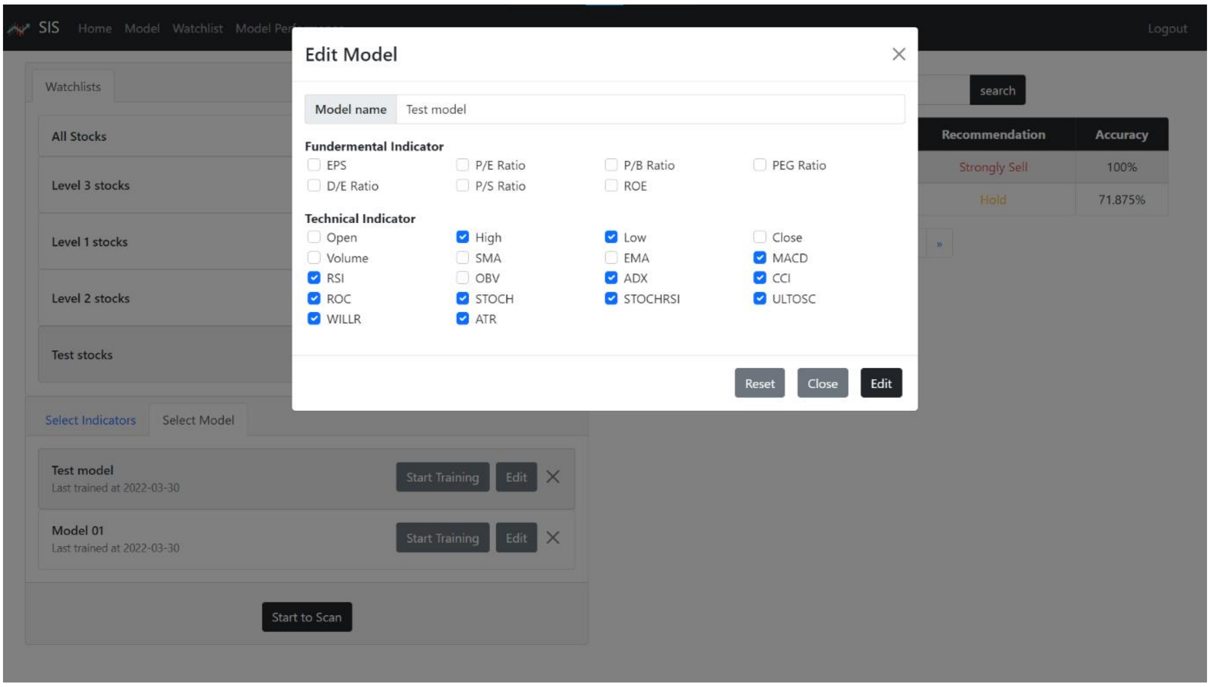
Test stocks

Test model
Last trained at 2022-03-30

Model 01
Last trained at 2022-03-30

Symbol	Name	Recommendation	Accuracy
1155.KL	MALAYAN BANKING BHD	Strongly Sell	100%
5099.KL	AIRASIA GROUP BERHAD	Hold	71.875%

Figure 5.2.1 *Prediction result of the proposed system*



Edit Model

Model name: Test model

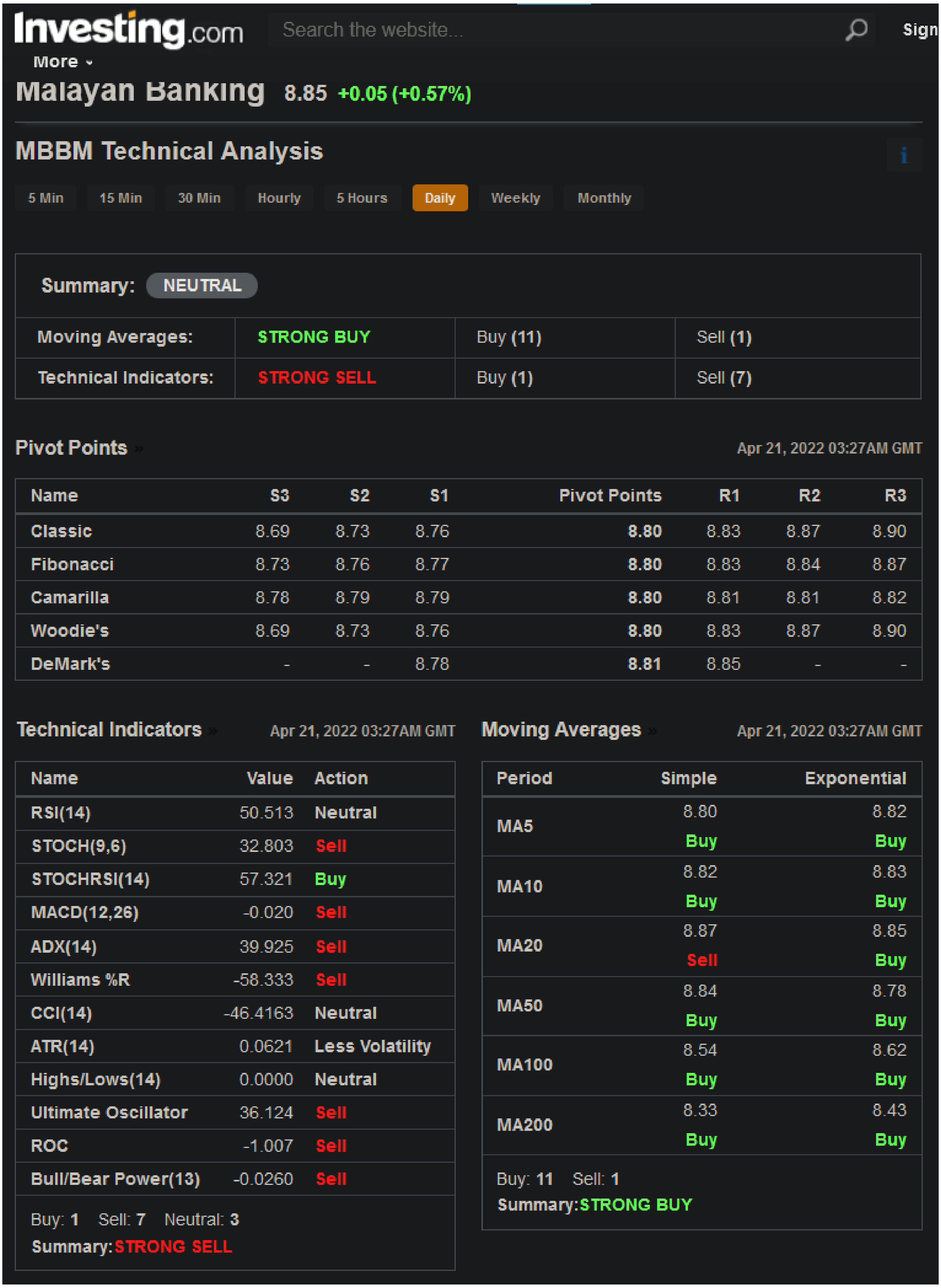
Fundermental Indicator

☐ EPS ☐ P/E Ratio ☐ P/B Ratio ☐ PEG Ratio
☐ D/E Ratio ☐ P/S Ratio ☐ ROE

Technical Indicator

☐ Open ☑ High ☑ Low ☐ Close
☐ Volume ☐ SMA ☐ EMA ☑ MACD
☑ RSI ☐ OBV ☑ ADX ☑ CCI
☑ ROC ☑ STOCH ☑ STOCHRSI ☑ ULTOSC
☑ WILLR ☑ ATR

Figure 5.2.2 *The indicators used in the model*



Malayan Banking 8.85 +0.05 (+0.57%)

MBBM Technical Analysis

5 Min | 15 Min | 30 Min | Hourly | 5 Hours | Daily | Weekly | Monthly

Summary: NEUTRAL

Moving Averages:	STRONG BUY	Buy (11)	Sell (1)
Technical Indicators:	STRONG SELL	Buy (1)	Sell (7)

Pivot Points » Apr 21, 2022 03:27AM GMT

Name	S3	S2	S1	Pivot Points	R1	R2	R3
Classic	8.69	8.73	8.76	8.80	8.83	8.87	8.90
Fibonacci	8.73	8.76	8.77	8.80	8.83	8.84	8.87
Camarilla	8.78	8.79	8.79	8.80	8.81	8.81	8.82
Woodie's	8.69	8.73	8.76	8.80	8.83	8.87	8.90
DeMark's	-	-	8.78	8.81	8.85	-	-

Technical Indicators » Apr 21, 2022 03:27AM GMT

Name	Value	Action
RSI(14)	50.513	Neutral
STOCH(9,6)	32.803	Sell
STOCHRSI(14)	57.321	Buy
MACD(12,26)	-0.020	Sell
ADX(14)	39.925	Sell
Williams %R	-58.333	Sell
CCI(14)	-46.4163	Neutral
ATR(14)	0.0621	Less Volatility
Highs/Lows(14)	0.0000	Neutral
Ultimate Oscillator	36.124	Sell
ROC	-1.007	Sell
Bull/Bear Power(13)	-0.0260	Sell

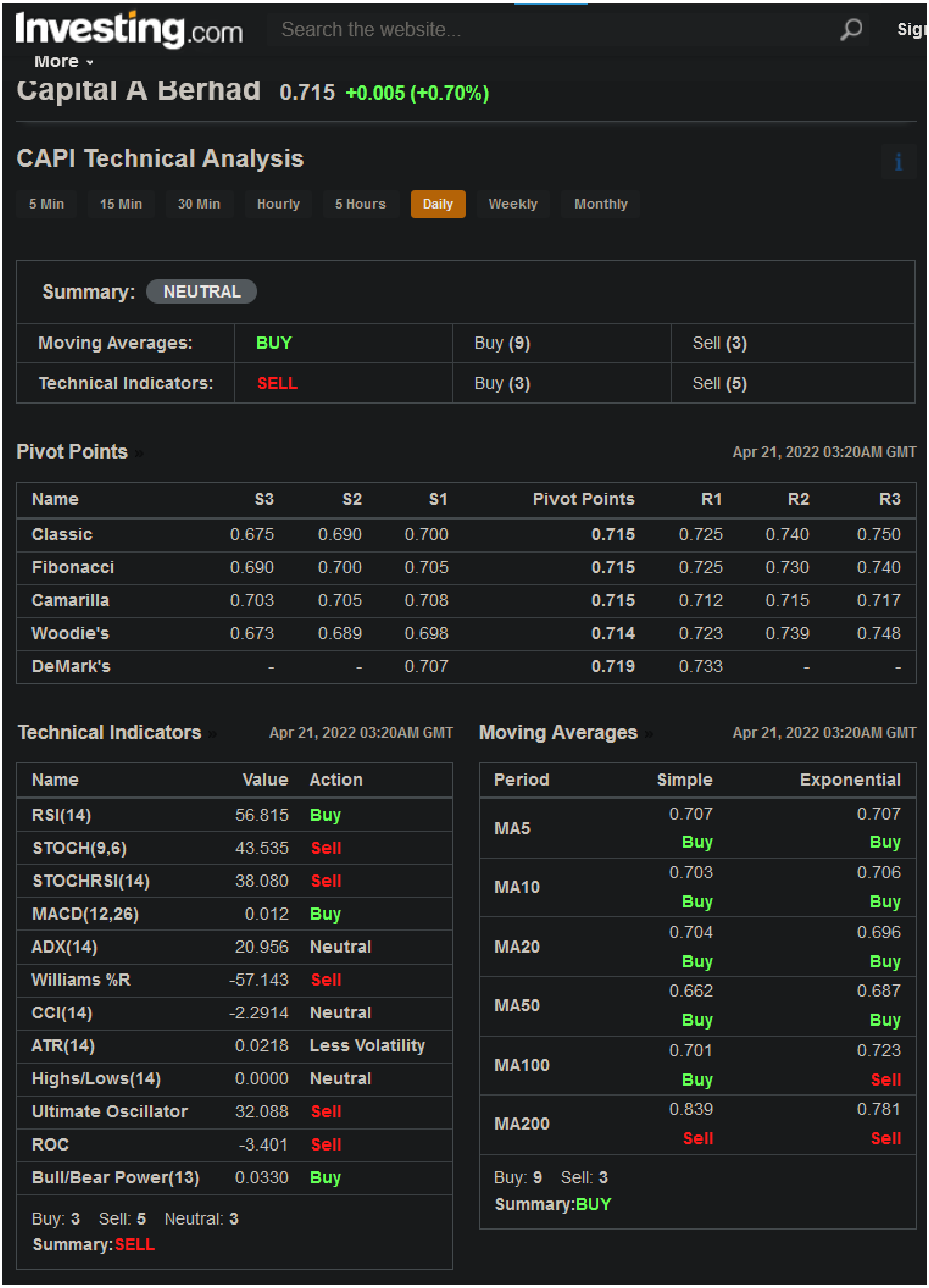
Buy: 1 Sell: 7 Neutral: 3
Summary:STRONG SELL

Moving Averages » Apr 21, 2022 03:27AM GMT

Period	Simple	Exponential
MA5	8.80 Buy	8.82 Buy
MA10	8.82 Buy	8.83 Buy
MA20	8.87 Sell	8.85 Buy
MA50	8.84 Buy	8.78 Buy
MA100	8.54 Buy	8.62 Buy
MA200	8.33 Buy	8.43 Buy

Buy: 11 Sell: 1
Summary:STRONG BUY

Figure 5.2.3 *Prediction result of Investing.com for symbol 1155.KL (Malayan Banking Bhd)*



Capital A Berhad 0.715 +0.005 (+0.70%)

CAPI Technical Analysis

Summary: NEUTRAL

Moving Averages:	BUY	Buy (9)	Sell (3)
Technical Indicators:	SELL	Buy (3)	Sell (5)

Pivot Points » Apr 21, 2022 03:20AM GMT

Name	S3	S2	S1	Pivot Points	R1	R2	R3
Classic	0.675	0.690	0.700	**0.715**	0.725	0.740	0.750
Fibonacci	0.690	0.700	0.705	**0.715**	0.725	0.730	0.740
Camarilla	0.703	0.705	0.708	**0.715**	0.712	0.715	0.717
Woodie's	0.673	0.689	0.698	**0.714**	0.723	0.739	0.748
DeMark's	-	-	0.707	**0.719**	0.733	-	-

Technical Indicators » Apr 21, 2022 03:20AM GMT

Name	Value	Action
RSI(14)	56.815	Buy
STOCH(9,6)	43.535	Sell
STOCHRSI(14)	38.080	Sell
MACD(12,26)	0.012	Buy
ADX(14)	20.956	Neutral
Williams %R	-57.143	Sell
CCI(14)	-2.2914	Neutral
ATR(14)	0.0218	Less Volatility
Highs/Lows(14)	0.0000	Neutral
Ultimate Oscillator	32.088	Sell
ROC	-3.401	Sell
Bull/Bear Power(13)	0.0330	Buy

Buy: 3 Sell: 5 Neutral: 3
Summary:SELL

Moving Averages » Apr 21, 2022 03:20AM GMT

Period	Simple	Exponential
MA5	0.707 Buy	0.707 Buy
MA10	0.703 Buy	0.706 Buy
MA20	0.704 Buy	0.696 Buy
MA50	0.662 Buy	0.687 Buy
MA100	0.701 Buy	0.723 Sell
MA200	0.839 Sell	0.781 Sell

Buy: 9 Sell: 3
Summary:BUY

Figure 5.2.4 *Prediction result of Investing.com for symbol 5099.KL (Capital A Bhd)*

CHAPTER 6: DISCUSSION

Based on the test results discussed in section 5.2, we can notice that the performance of the proposed system is better than Buy-and-Hold strategy. This indicates that the web application with dynamic stock prediction model which based on deep reinforcement learning not only enable input customization, but also enable user to gain better returns while manage the risk during stock investment. Thus objective 1 of this project is achieved. Furthermore, from the test results, we can notice that the prediction results of the proposed system are very close and similar to the existing system that is Investing.com. This shown that the prediction provided by the proposed system able to assist investor in decision making of stock trading just like the forecast Investing.com provided. Therefore, achieved objective 2 of this project.

6.1 System Limitation

From the test results we can observe that the accuracy of some model is not high enough. This might cause some future prediction result is not accurate can cause losses to user in stock investment. Other limitation of this system is only top 100 stocks in Malaysia is covered. Other financial instrument in other country is not covered. On the other hands, only limited indicators are available to choose in this system. If user would like to use indicator like BOP (Balance Of Power), PPO (Percentage Price Oscillator), and other indicators which are not covered, user unable to do so. Besides, user cannot change the parameter of an indicator. For example, the time period of SMA used in the system is 30 days and user unable to change it. Next, only structured data are used in the prediction, unstructured data like news and social media are not covered.

6.2 Future Enhancement

Due to the time constraint of the project, a lot of future work can be done to improve the system. For instance, more data of the stock can be used in model training and apply more episode in the model training to have better model accuracy. Moreover, the support of the system can increase by adding regions other than Malaysia. Other than that, more stocks, fundamental indicators, and technical indicators can be added to provide more option for user.

The coverage for other financial instrument like bonds, futures, and options contracts also is an update that can be done in the future. Furthermore, the accuracy of the prediction can be increased by adding the support of unstructured data like news and social media as input.

CHAPTER 7: CONCLUSION

Investor always try to forecast the trend of stock in order to gain better profit in stock market. As the capability of machine learning increase, many of the machine learning approaches had been proposed by researcher to forecast the stock market. Although these approaches can provide good prediction on stock market, investor cannot customize the type of input data used in the prediction. The objectives of this project are to deliver a web application with dynamic stock prediction model which enable input customization and assist investor to gain better profit in stock market. Although from the test results, we can notice that the objectives of this project is achieved and the performance of the proposed system is acceptable, however the system still have some limitation and many improvement can be done in the future.

REFERENCES

[1] V. Drakopoulou, “A Review of Fundamental and Technical Stock Analysis Techniques,” *Journal of Stock & Forex Trading*, vol. 05, no. 01, 2016, doi: 10.4172/2168-9458.1000163.

[2] J. Chen, “Stock Market | Investopedia,” *Investopedia*, 2019. https://www.investopedia.com/terms/s/stockmarket.asp (accessed Mar. 20, 2021).

[3] Bursa Malaysia, “Listing Criteria,” *Bursamalaysia.com*, 2016. https://www.bursamalaysia.com/listing/get_listed/listing_criteria (accessed Jul. 13, 2021).

[4] Bursa Malaysia, “Islamic Market,” *www.bursamalaysia.com*. https://www.bursamalaysia.com/trade/market/islamic_market (accessed Jul. 13, 2021).

[5] J. Young, “Understanding Market Indexes and Their Uses Helps Investors,” *Investopedia*, 2019. https://www.investopedia.com/terms/m/marketindex.asp (accessed Mar. 20, 2021).

[6] T. Segal, “Fundamental Analysis Definition,” *Investopedia*, 2019. https://www.investopedia.com/terms/f/fundamentalanalysis.asp (accessed Mar. 20, 2021).

[7] J. Fernando, “Understanding Earnings Per Share – EPS,” *Investopedia*, Sep. 17, 2020. https://www.investopedia.com/terms/e/eps.asp (accessed Mar. 22, 2021).

[8] J. Fernando, “What the Price-to-Earnings Ratio Tells Us,” *Investopedia*, Sep. 09, 2021. https://www.investopedia.com/terms/p/price-earningsratio.asp (accessed Mar. 22, 2021).

[9] A. Hayes, “What the Price-to-Book Ratio (P/B Ratio) Tells Us,” *Investopedia*, 2019. https://www.investopedia.com/terms/p/price-to-bookratio.asp (accessed Mar. 22, 2021).

[10] W. Kenton, “Why the Price/Earnings-to-Growth Ratio Matters,” *Investopedia*, 2019. https://www.investopedia.com/terms/p/pegratio.asp (accessed Mar. 22, 2021).

[11] A. Hayes, “Technical Analysis,” *Investopedia*, 2019. https://www.investopedia.com/terms/t/technicalanalysis.asp (accessed Mar. 28, 2021).

[12] J. Fernando, “Moving Average (MA),” *Investopedia*, 2019. https://www.investopedia.com/terms/m/movingaverage.asp (accessed Mar. 28, 2021).

[13] J. Fernando, “Moving Average Convergence Divergence – MACD Definition,” *Investopedia*, 2019. https://www.investopedia.com/terms/m/macd.asp (accessed Mar. 28, 2021).

[14] C. M. Brown and G. C. Lane, *Technical analysis for the trading professional : strategies and techniques for today’s turbulent global financial markets*. New York: Mcgraw-Hill, 2012, p. 13.

[15] J. Fernando, “Relative Strength Index – RSI,” *Investopedia*, 2019. https://www.investopedia.com/terms/r/rsi.asp (accessed Mar. 28, 2021).

[16] A. Hayes, “On-Balance Volume (OBV),” *Investopedia*, 2019. https://www.investopedia.com/terms/o/onbalancevolume.asp (accessed Sep. 30, 2021).

[17] X. Wu, H. Chen, J. Wang, L. Troiano, V. Loia, and H. Fujita, “Adaptive stock trading strategies with deep reinforcement learning methods,” *Information Sciences*, vol. 538, pp. 142–158, Oct. 2020, doi: 10.1016/j.ins.2020.05.066.

[18] A. Choudhary, “Introduction to Deep Q-Learning for Reinforcement Learning (in Python),” *Analytics Vidhya*, Apr. 18, 2019. https://www.analyticsvidhya.com/blog/2019/04/introduction-deep-q-learning-python/ (accessed Nov. 19, 2021).

[19] L. Faik, “Deep Q Network: Combining Deep & Reinforcement Learning,” *Medium*, Aug. 21, 2021. https://towardsdatascience.com/deep-q-network-combining-deep-reinforcement-learning-a5616bcfc207 (accessed Nov. 19, 2021).

[20] D. Mwiti, “10 Real-Life Applications of Reinforcement Learning,” *neptune.ai*, Jul. 22, 2020. https://neptune.ai/blog/reinforcement-learning-applications (accessed Nov. 20, 2021).

[21] A. Arévalo, J. Niño, G. Hernández, and J. Sandoval, “High-Frequency Trading Strategy Based on Deep Neural Networks,” *Intelligent Computing Methodologies*, pp. 424–436, 2016, doi: 10.1007/978-3-319-42297-8_40.

[22] W. Bao, J. Yue, and Y. Rao, “A deep learning framework for financial time series using stacked autoencoders and long-short term memory,” *PLOS ONE*, vol. 12, no. 7, p. e0180944, Jul. 2017, doi: 10.1371/journal.pone.0180944.

[23] J. Moody and M. Saffell, “Learning to trade via direct reinforcement,” *IEEE Transactions on Neural Networks*, vol. 12, no. 4, pp. 875–889, Jul. 2001, doi: 10.1109/72.935097.

[24] J. Carapuço, R. Neves, and N. Horta, “Reinforcement learning applied to Forex trading,” *Applied Soft Computing*, vol. 73, pp. 783–794, Dec. 2018, doi: 10.1016/j.asoc.2018.09.017.

[25] Y. Deng, F. Bao, Y. Kong, Z. Ren, and Q. Dai, “Deep Direct Reinforcement Learning for Financial Signal Representation and Trading,” *IEEE Transactions on Neural Networks and Learning Systems*, vol. 28, no. 3, pp. 653–664, Mar. 2017, doi: 10.1109/tnnls.2016.2522401.

[26] W. Si, J. Li, P. Ding, and R. Rao, “A Multi-objective Deep Reinforcement Learning

Approach for Stock Index Future's Intraday Trading," *2017 10th International Symposium on Computational Intelligence and Design (ISCID)*, Dec. 2017, doi: 10.1109/iscid.2017.210.

[27] T. Théate and D. Ernst, "An application of deep reinforcement learning to algorithmic trading," *Expert Systems with Applications*, vol. 173, p. 114632, Jul. 2021, doi: 10.1016/j.eswa.2021.114632.

[28] S. Carta, A. Ferreira, A. S. Podda, D. Reforgiato Recupero, and A. Sanna, "Multi-DQN: An ensemble of Deep Q-learning agents for stock market forecasting," *Expert Systems with Applications*, vol. 164, p. 113820, Feb. 2021, doi: 10.1016/j.eswa.2020.113820.

[29] H. Pedamallu, "RNN vs GRU vs LSTM," *Analytics Vidhya*, Nov. 30, 2020. https://medium.com/analytics-vidhya/rnn-vs-gru-vs-lstm-863b0b7b1573 (accessed Nov. 17, 2021).

[30] M. Phi, "Illustrated Guide to LSTM's and GRU's: A step by step explanation," *Medium*, Jul. 10, 2019. https://towardsdatascience.com/illustrated-guide-to-lstms-and-gru-s-a-step-by-step-explanation-44e9eb85bf21 (accessed Nov. 17, 2021).

[31] Bursa Malaysia, "THE LIST OF FTSE BURSA MALAYSIA KLCI INDEX CONSTITUENTS." Accessed: Aug. 19, 2021. [Online]. Available: https://www.bursamalaysia.com/sites/5d809dcf39fba22790cad230/assets/60cc666439fba26c355d5eae/FBMKLCI_Constituents_List_Jun2021.pdf.

[32] Bursa Malaysia, "THE LIST OF FTSE BURSA MALAYSIA MID 70 INDEX CONSTITUENTS." Accessed: Aug. 19, 2021. [Online]. Available: https://www.bursamalaysia.com/sites/5d809dcf39fba22790cad230/assets/60cc58165b711a70d6ad030f/FBM70_Constituents_List_Jun2021.pdf.

APPENDICES

APPENDIX A

The List of FTSE Bursa Malaysia Top 100 Index Constituents [31] and [32]

NO.	STOCK CODE	CONSTITUENT NAME
1	6888	AXIATA GROUP BERHAD
2	1023	CIMB GROUP HOLDINGS BERHAD
3	7277	DIALOG GROUP BHD
4	6947	DIGI.COM BHD
5	3182	GENTING BHD
6	4715	GENTING MALAYSIA BERHAD
7	3034	HAP SENG CONSOLIDATED BHD
8	5168	HARTALEGA HOLDINGS BHD
9	5819	HONG LEONG BANK BHD
10	1082	HONG LEONG FINANCIAL GROUP BHD
11	5225	IHH HEALTHCARE BERHAD
12	1961	IOI CORPORATION BHD
13	2445	KUALA LUMPUR KEPONG BHD
14	1155	MALAYAN BANKING BHD
15	6012	MAXIS BERHAD
16	3816	MISC BHD
17	5296	MR D.I.Y. GROUP (M) BERHAD
18	4707	NESTLE (M) BHD
19	5183	PETRONAS CHEMICALS GROUP BHD
20	5681	PETRONAS DAGANGAN BHD
21	6033	PETRONAS GAS BHD
22	4065	PPB GROUP BHD
23	8869	PRESS METAL ALUMINIUM HOLDINGS BERHAD
24	1295	PUBLIC BANK BHD
25	1066	RHB BANK BERHAD

NO.	STOCK CODE	CONSTITUENT NAME
26	4197	SIME DARBY BHD
27	5285	SIME DARBY PLANTATION BERHAD
28	4863	TELEKOM MALAYSIA BHD
29	5347	TENAGA NASIONAL BHD
30	7113	TOP GLOVE CORPORATION BHD
31	5139	AEON CREDIT SERVICE (M) BHD
32	5099	AIRASIA GROUP BERHAD
33	2488	ALLIANCE BANK MALAYSIA BERHAD
34	1015	AMMB HOLDINGS BHD
35	6399	ASTRO MALAYSIA HOLDINGS BERHAD
36	8176	ATA IMS BERHAD
37	5106	AXIS REITS
38	1562	BERJAYA SPORTS TOTO BHD
39	5248	BERMAZ AUTO BERHAD
40	4162	BRITISH AMERICAN TOBACCO (M)
41	5210	BUMI ARMADA BERHAD
42	1818	BURSA MALAYSIA BHD
43	2852	CAHYA MATA SARAWAK BHD
44	2836	CARLSBERG BREWERY MALAYSIA BHD
45	7204	D & O GREEN TECHNOLOGIES BERHAD
46	1619	DRB-HICOM BHD
47	7148	DUOPHARMA BIOTECH BERHAD
48	5222	FGV HOLDINGS BERHAD
49	3689	FRASER & NEAVE HOLDINGS BHD
50	128	FRONTKEN CORPORATION BHD
51	5398	GAMUDA BHD
52	78	GDEX BERHAD
53	2291	GENTING PLANTATIONS BERHAD
54	208	GREATECH TECHNOLOGY BERHAD
55	5102	GUAN CHONG BHD
56	3255	HEINEKEN MALAYSIA BERHAD
57	3301	HONG LEONG INDUSTRIES BHD
58	5227	IGB REAL ESTATE INV TRUST
59	3336	IJM CORPORATION BHD
60	166	INARI AMERTRON BERHAD

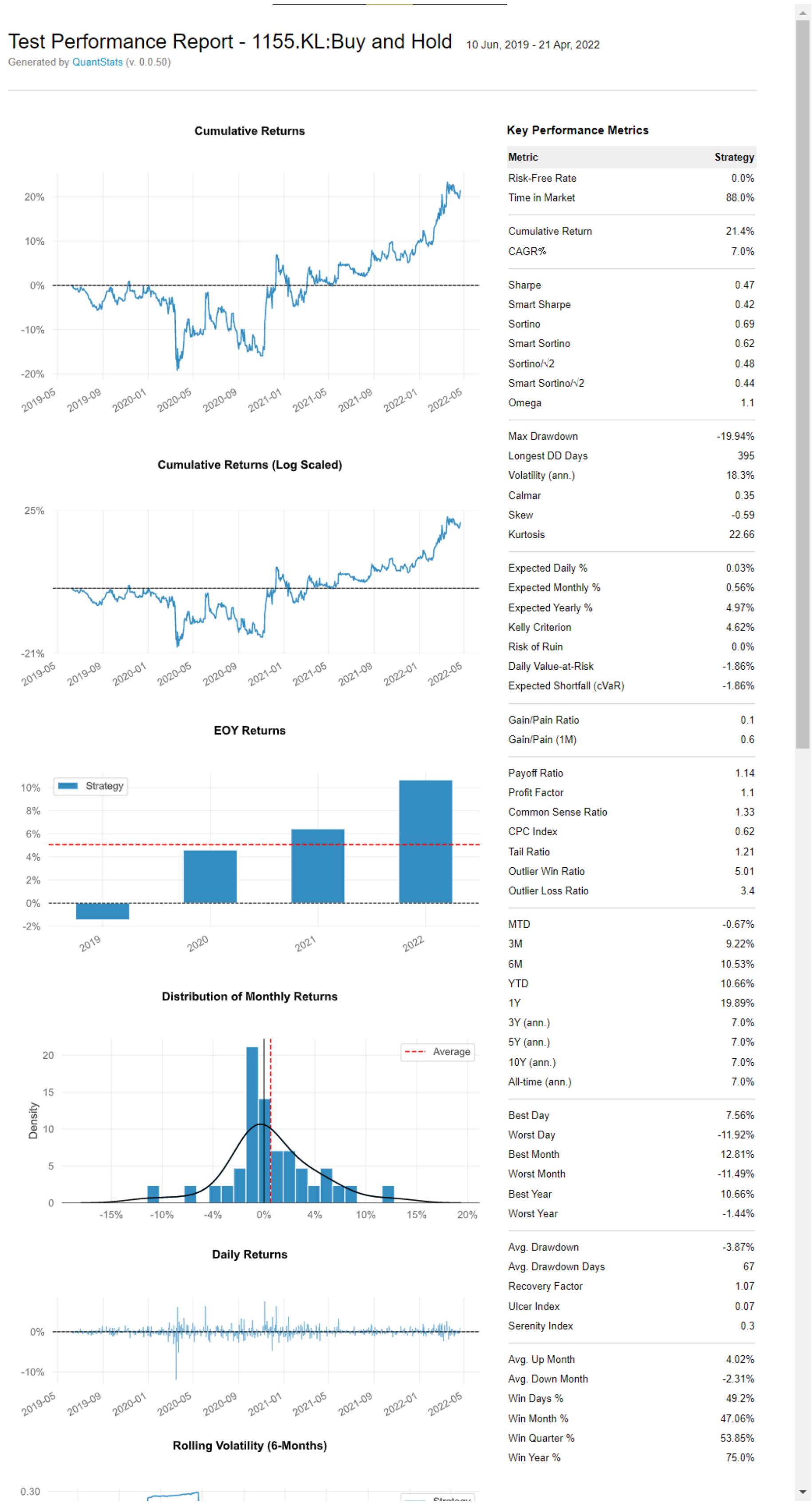
NO.	STOCK CODE	CONSTITUENT NAME
61	7153	KOSSAN RUBBER INDUSTRIES BHD
62	5878	KPJ HEALTHCARE BHD
63	6633	LEONG HUP INTERNATIONAL BERHAD
64	5284	LOTTE CHEMICAL TITAN HOLDING BERHAD
65	3859	MAGNUM BERHAD
66	5264	MALAKOFF CORPORATION BERHAD
67	5014	MALAYSIA AIRPORTS HOLDINGS BHD
68	1171	MALAYSIA BUILDING SOCIETY BHD
69	3867	MALAYSIAN PACIFIC INDUSTRIES
70	1651	MALAYSIAN RESOURCES CORPORATION BERHAD
71	5236	MATRIX CONCEPTS HOLDINGS BHD
72	3069	MEGA FIRST CORPORATION BHD
73	5286	MI TECHNOVATION BERHAD
74	2194	MMC CORPORATION BHD
75	138	MY E.G. SERVICES BHD
76	7052	PADINI HOLDINGS BHD
77	7160	PENTAMASTER CORPORATION BHD
78	7084	QL RESOURCES BHD
79	5218	SAPURA ENERGY BERHAD
80	4731	SCIENTEX BERHAD
81	5279	SERBA DINAMIK HOLDINGS BERHAD
82	5288	SIME DARBY PROPERTY BERHAD
83	7155	SKP RESOURCES BHD
84	8664	SP SETIA BHD
85	5211	SUNWAY BERHAD
86	5176	SUNWAY REAL ESTATE INVT TRUST
87	7106	SUPERMAX CORPORATION BHD
88	6139	SYARIKAT TAKAFUL MALAYSIA KELUARGA BERHAD
89	5031	TIME DOTCOM BHD
90	5148	UEM SUNRISE BERHAD
91	4588	UMW HOLDINGS BHD
92	5005	UNISEM (M) BHD
93	5292	UWC BERHAD
94	6963	V.S INDUSTRY BHD
95	97	VITROX CORPORATION BHD
96	5246	WESTPORTS HOLDINGS BERHAD

NO.	STOCK CODE	CONSTITUENT NAME
97	7293	YINSON HOLDINGS BHD
98	4677	YTL CORPORATION BHD
99	5109	YTL HOSPITALITY REIT
100	6742	YTL POWER INTERNATIONAL BHD

(Note: The constituents list was last updated on 21 June 2021)

APPENDIX B

The performance report of Buy-and-Hold strategy on symbol 1155.KL



Test Performance Report - 1155.KL:Buy and Hold 10 Jun, 2019 - 21 Apr, 2022

Generated by QuantStats (v. 0.0.50)

Cumulative Returns

Cumulative Returns (Log Scaled)

EOY Returns

Distribution of Monthly Returns

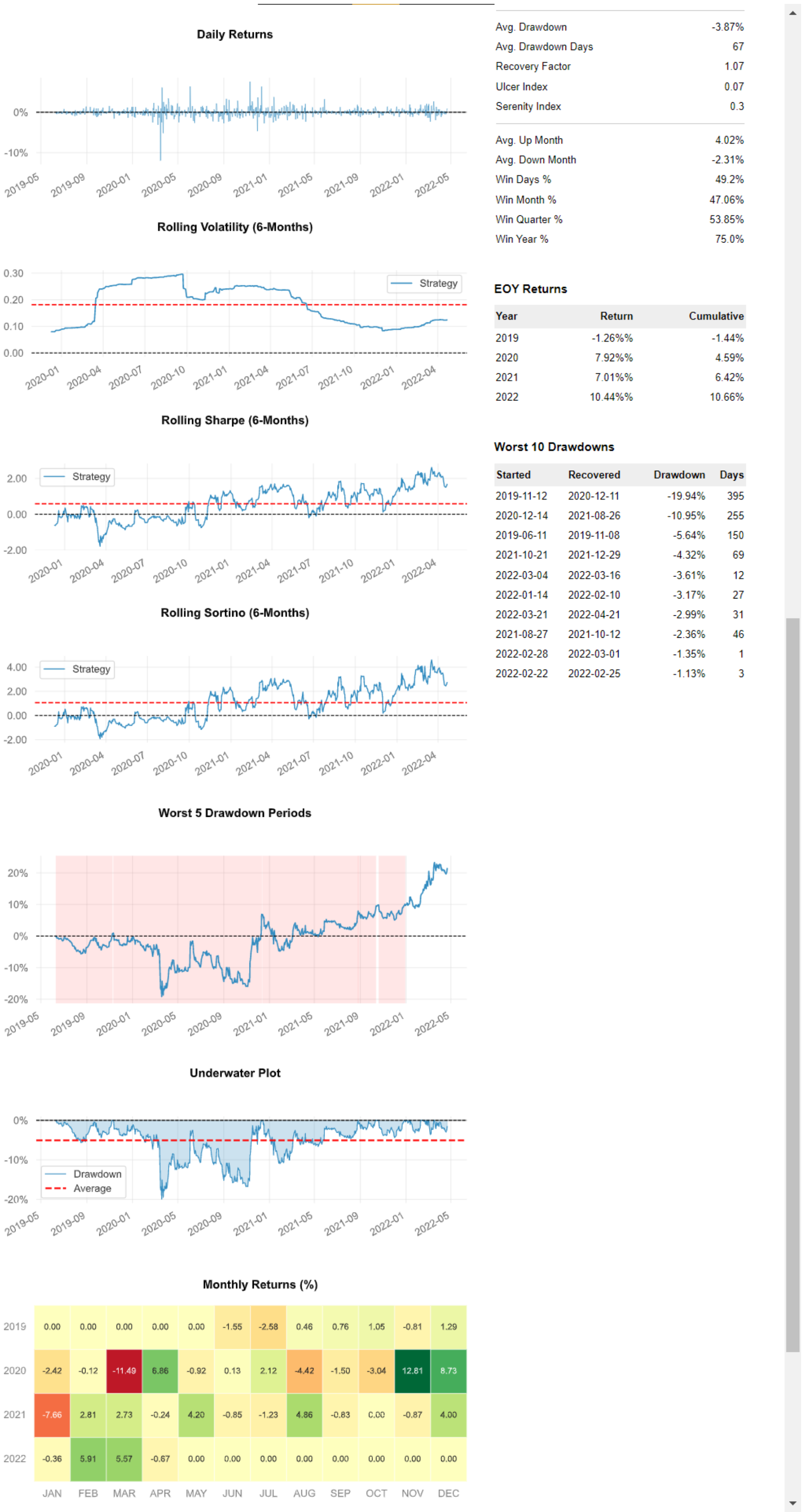
Daily Returns

Rolling Volatility (6-Months)

Key Performance Metrics

Metric	Strategy
Risk-Free Rate	0.0%
Time in Market	88.0%
Cumulative Return	21.4%
CAGR﹪	7.0%
Sharpe	0.47
Smart Sharpe	0.42
Sortino	0.69
Smart Sortino	0.62
Sortino/√2	0.48
Smart Sortino/√2	0.44
Omega	1.1
Max Drawdown	-19.94%
Longest DD Days	395
Volatility (ann.)	18.3%
Calmar	0.35
Skew	-0.59
Kurtosis	22.66
Expected Daily %	0.03%
Expected Monthly %	0.56%
Expected Yearly %	4.97%
Kelly Criterion	4.62%
Risk of Ruin	0.0%
Daily Value-at-Risk	-1.86%
Expected Shortfall (cVaR)	-1.86%
Gain/Pain Ratio	0.1
Gain/Pain (1M)	0.6
Payoff Ratio	1.14
Profit Factor	1.1
Common Sense Ratio	1.33
CPC Index	0.62
Tail Ratio	1.21
Outlier Win Ratio	5.01
Outlier Loss Ratio	3.4
MTD	-0.67%
3M	9.22%
6M	10.53%
YTD	10.66%
1Y	19.89%
3Y (ann.)	7.0%
5Y (ann.)	7.0%
10Y (ann.)	7.0%
All-time (ann.)	7.0%
Best Day	7.56%
Worst Day	-11.92%
Best Month	12.81%
Worst Month	-11.49%
Best Year	10.66%
Worst Year	-1.44%
Avg. Drawdown	-3.87%
Avg. Drawdown Days	67
Recovery Factor	1.07
Ulcer Index	0.07
Serenity Index	0.3
Avg. Up Month	4.02%
Avg. Down Month	-2.31%
Win Days %	49.2%
Win Month %	47.06%
Win Quarter %	53.85%
Win Year %	75.0%

(Part 1)



Avg. Drawdown	-3.87%
Avg. Drawdown Days	67
Recovery Factor	1.07
Ulcer Index	0.07
Serenity Index	0.3
Avg. Up Month	4.02%
Avg. Down Month	-2.31%
Win Days %	49.2%
Win Month %	47.06%
Win Quarter %	53.85%
Win Year %	75.0%

EOY Returns

Year	Return	Cumulative
2019	-1.26%%	-1.44%
2020	7.92%%	4.59%
2021	7.01%%	6.42%
2022	10.44%%	10.66%

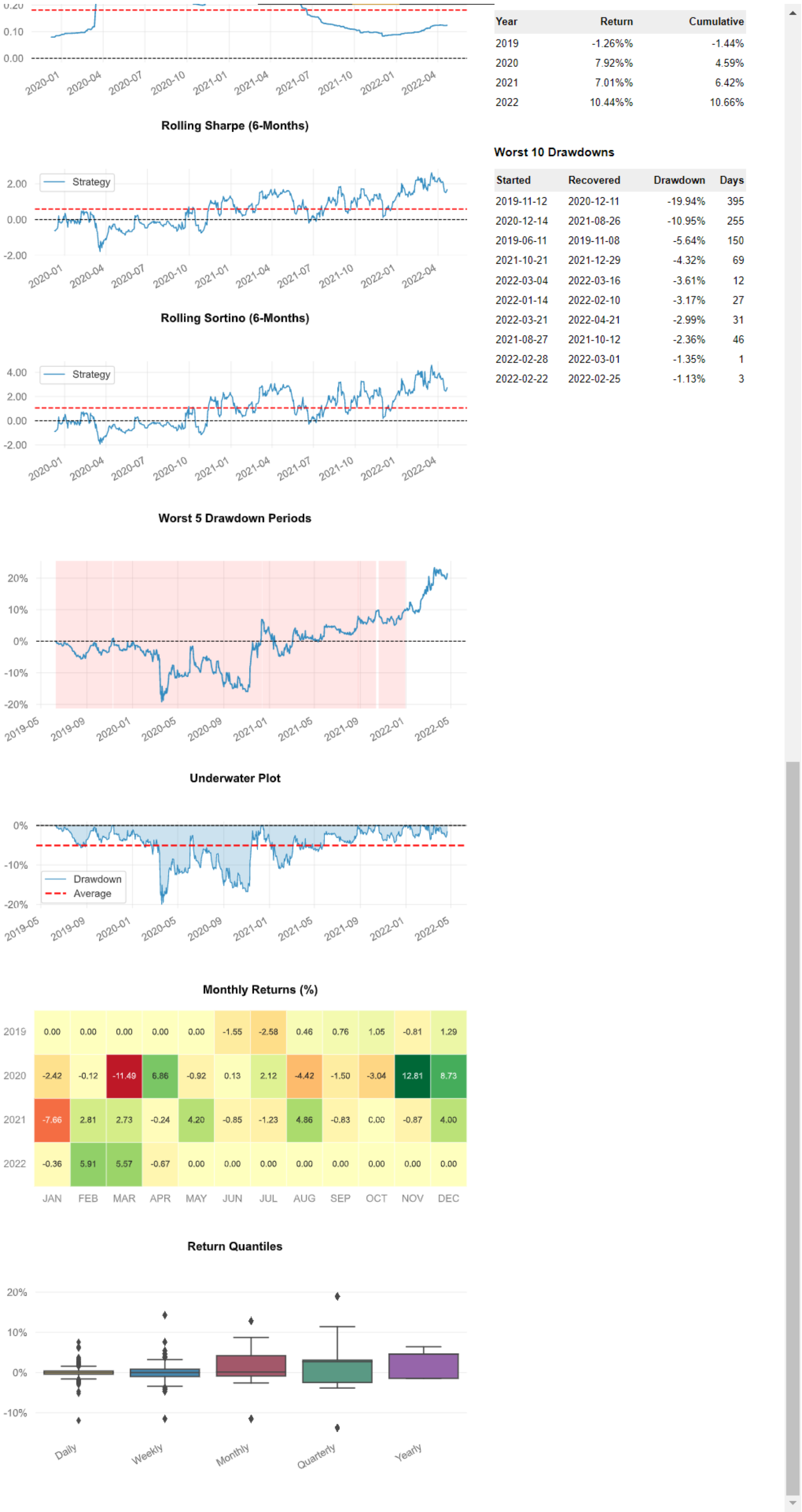
Worst 10 Drawdowns

Started	Recovered	Drawdown	Days
2019-11-12	2020-12-11	-19.94%	395
2020-12-14	2021-08-26	-10.95%	255
2019-06-11	2019-11-08	-5.64%	150
2021-10-21	2021-12-29	-4.32%	69
2022-03-04	2022-03-16	-3.61%	12
2022-01-14	2022-02-10	-3.17%	27
2022-03-21	2022-04-21	-2.99%	31
2021-08-27	2021-10-12	-2.36%	46
2022-02-28	2022-03-01	-1.35%	1
2022-02-22	2022-02-25	-1.13%	3

Monthly Returns (%)

	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	SEP	OCT	NOV	DEC
2019	0.00	0.00	0.00	0.00	0.00	-1.55	-2.58	0.46	0.76	1.05	-0.81	1.29
2020	-2.42	-0.12	-11.49	6.86	-0.92	0.13	2.12	-4.42	-1.50	-3.04	12.81	8.73
2021	-7.66	2.81	2.73	-0.24	4.20	-0.85	-1.23	4.86	-0.83	0.00	-0.87	4.00
2022	-0.36	5.91	5.57	-0.67	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00

(Part 2)



Year	Return	Cumulative
2019	-1.26%%	-1.44%
2020	7.92%%	4.59%
2021	7.01%%	6.42%
2022	10.44%%	10.66%

Rolling Sharpe (6-Months)

Worst 10 Drawdowns

Started	Recovered	Drawdown	Days
2019-11-12	2020-12-11	-19.94%	395
2020-12-14	2021-08-26	-10.95%	255
2019-06-11	2019-11-08	-5.64%	150
2021-10-21	2021-12-29	-4.32%	69
2022-03-04	2022-03-16	-3.61%	12
2022-01-14	2022-02-10	-3.17%	27
2022-03-21	2022-04-21	-2.99%	31
2021-08-27	2021-10-12	-2.36%	46
2022-02-28	2022-03-01	-1.35%	1
2022-02-22	2022-02-25	-1.13%	3

Rolling Sortino (6-Months)

Worst 5 Drawdown Periods

Underwater Plot

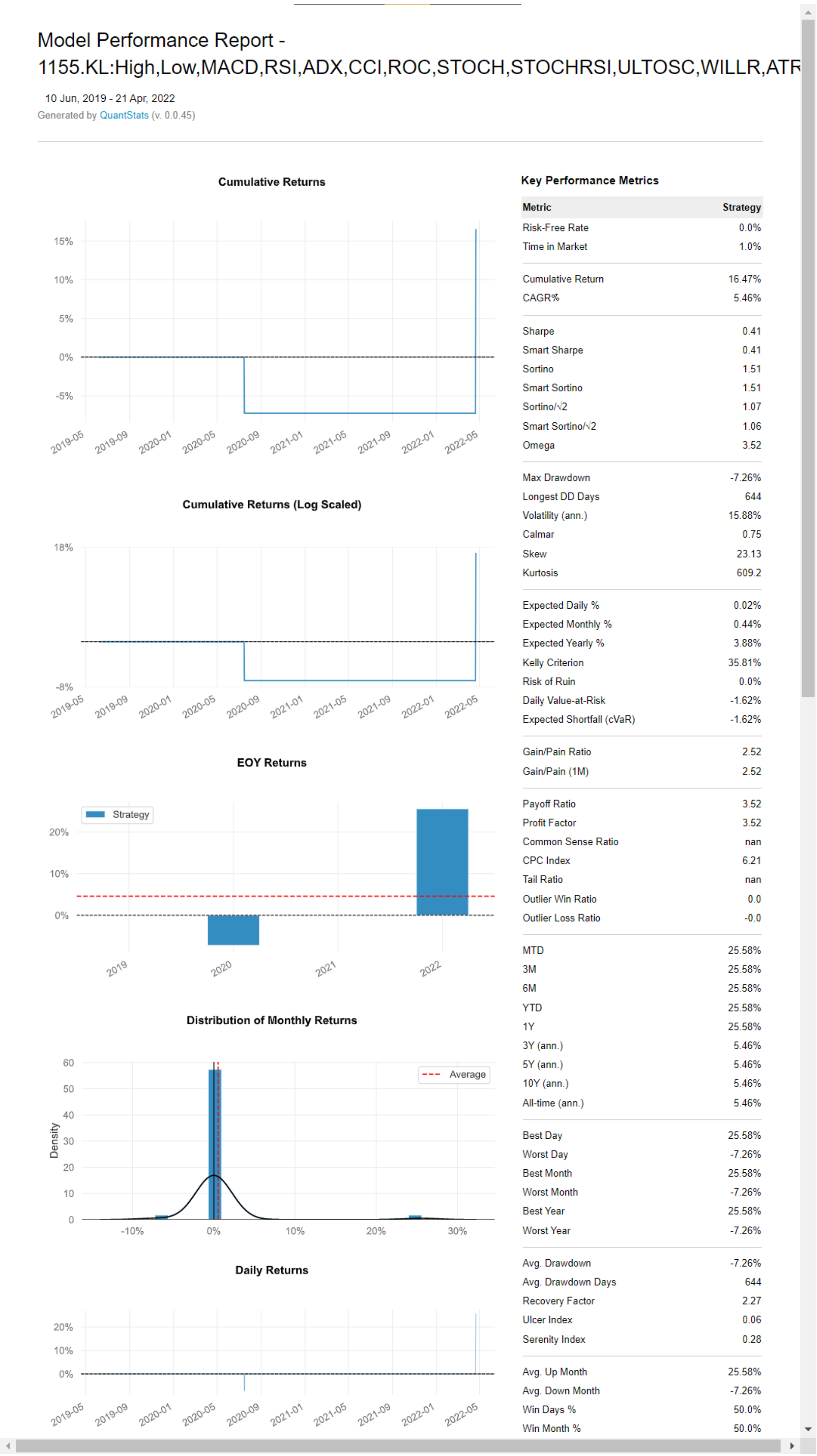
Monthly Returns (%)

	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	SEP	OCT	NOV	DEC
2019	0.00	0.00	0.00	0.00	0.00	-1.55	-2.58	0.46	0.76	1.05	-0.81	1.29
2020	-2.42	-0.12	-11.49	6.86	-0.92	0.13	2.12	-4.42	-1.50	-3.04	12.81	8.73
2021	-7.66	2.81	2.73	-0.24	4.20	-0.85	-1.23	4.86	-0.83	0.00	-0.87	4.00
2022	-0.36	5.91	5.57	-0.67	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00

Return Quantiles

(Part 3)

The performance report of proposed model on symbol 1155.KL



Model Performance Report - 1155.KL:High,Low,MACD,RSI,ADX,CCI,ROC,STOCH,STOCHRSI,ULTOSC,WILLR,ATR

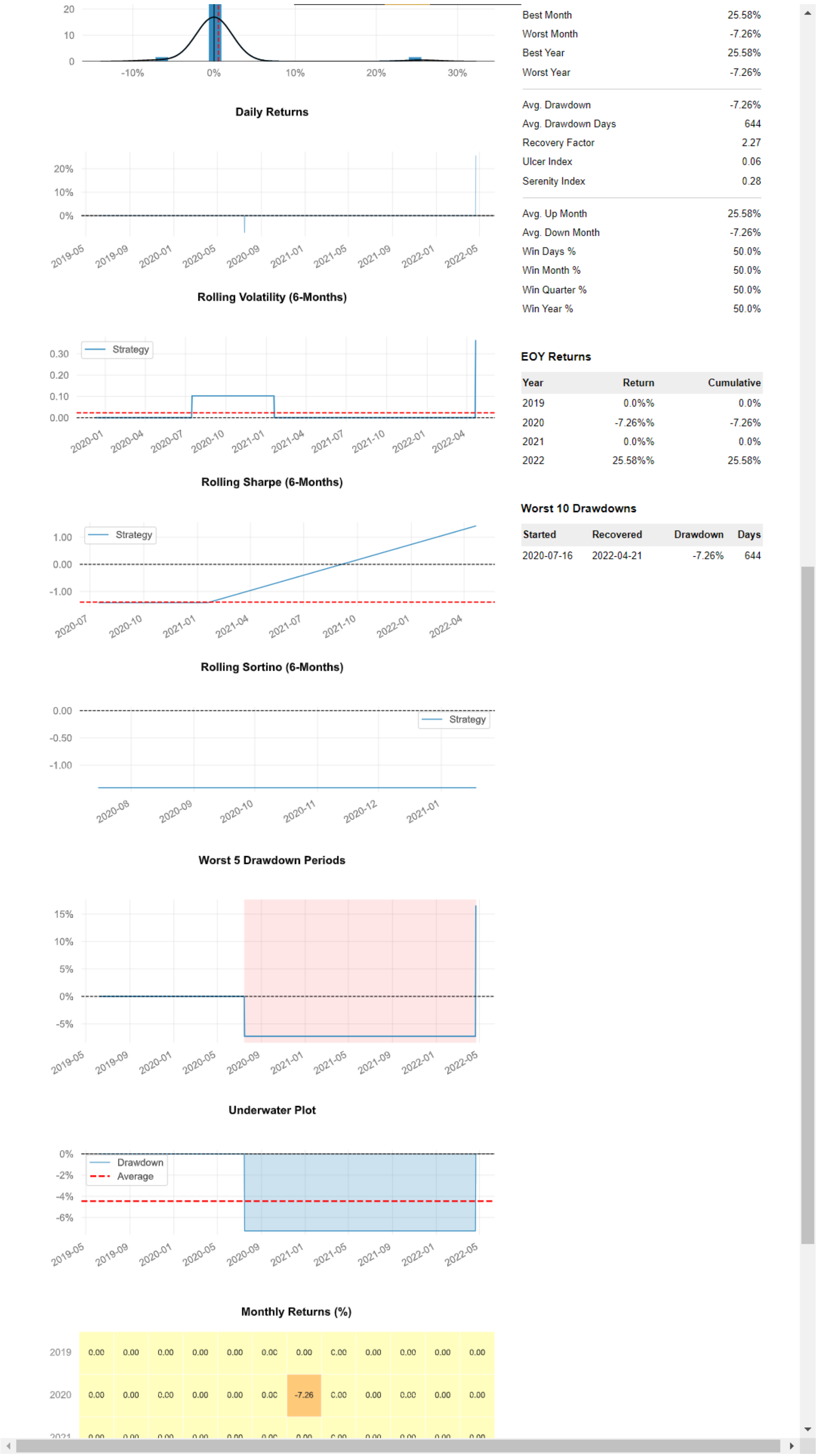
10 Jun, 2019 - 21 Apr, 2022

Generated by QuantStats (v. 0.0.45)

Key Performance Metrics

Metric	Strategy
Risk-Free Rate	0.0%
Time in Market	1.0%
Cumulative Return	16.47%
CAGR﹪	5.46%
Sharpe	0.41
Smart Sharpe	0.41
Sortino	1.51
Smart Sortino	1.51
Sortino/√2	1.07
Smart Sortino/√2	1.06
Omega	3.52
Max Drawdown	-7.26%
Longest DD Days	644
Volatility (ann.)	15.88%
Calmar	0.75
Skew	23.13
Kurtosis	609.2
Expected Daily %	0.02%
Expected Monthly %	0.44%
Expected Yearly %	3.88%
Kelly Criterion	35.81%
Risk of Ruin	0.0%
Daily Value-at-Risk	-1.62%
Expected Shortfall (cVaR)	-1.62%
Gain/Pain Ratio	2.52
Gain/Pain (1M)	2.52
Payoff Ratio	3.52
Profit Factor	3.52
Common Sense Ratio	nan
CPC Index	6.21
Tail Ratio	nan
Outlier Win Ratio	0.0
Outlier Loss Ratio	-0.0
MTD	25.58%
3M	25.58%
6M	25.58%
YTD	25.58%
1Y	25.58%
3Y (ann.)	5.46%
5Y (ann.)	5.46%
10Y (ann.)	5.46%
All-time (ann.)	5.46%
Best Day	25.58%
Worst Day	-7.26%
Best Month	25.58%
Worst Month	-7.26%
Best Year	25.58%
Worst Year	-7.26%
Avg. Drawdown	-7.26%
Avg. Drawdown Days	644
Recovery Factor	2.27
Ulcer Index	0.06
Serenity Index	0.28
Avg. Up Month	25.58%
Avg. Down Month	-7.26%
Win Days %	50.0%
Win Month %	50.0%

(Part 1)



Best Month	25.58%
Worst Month	-7.26%
Best Year	25.58%
Worst Year	-7.26%
Avg. Drawdown	-7.26%
Avg. Drawdown Days	644
Recovery Factor	2.27
Ulcer Index	0.06
Serenity Index	0.28
Avg. Up Month	25.58%
Avg. Down Month	-7.26%
Win Days %	50.0%
Win Month %	50.0%
Win Quarter %	50.0%
Win Year %	50.0%

EOY Returns

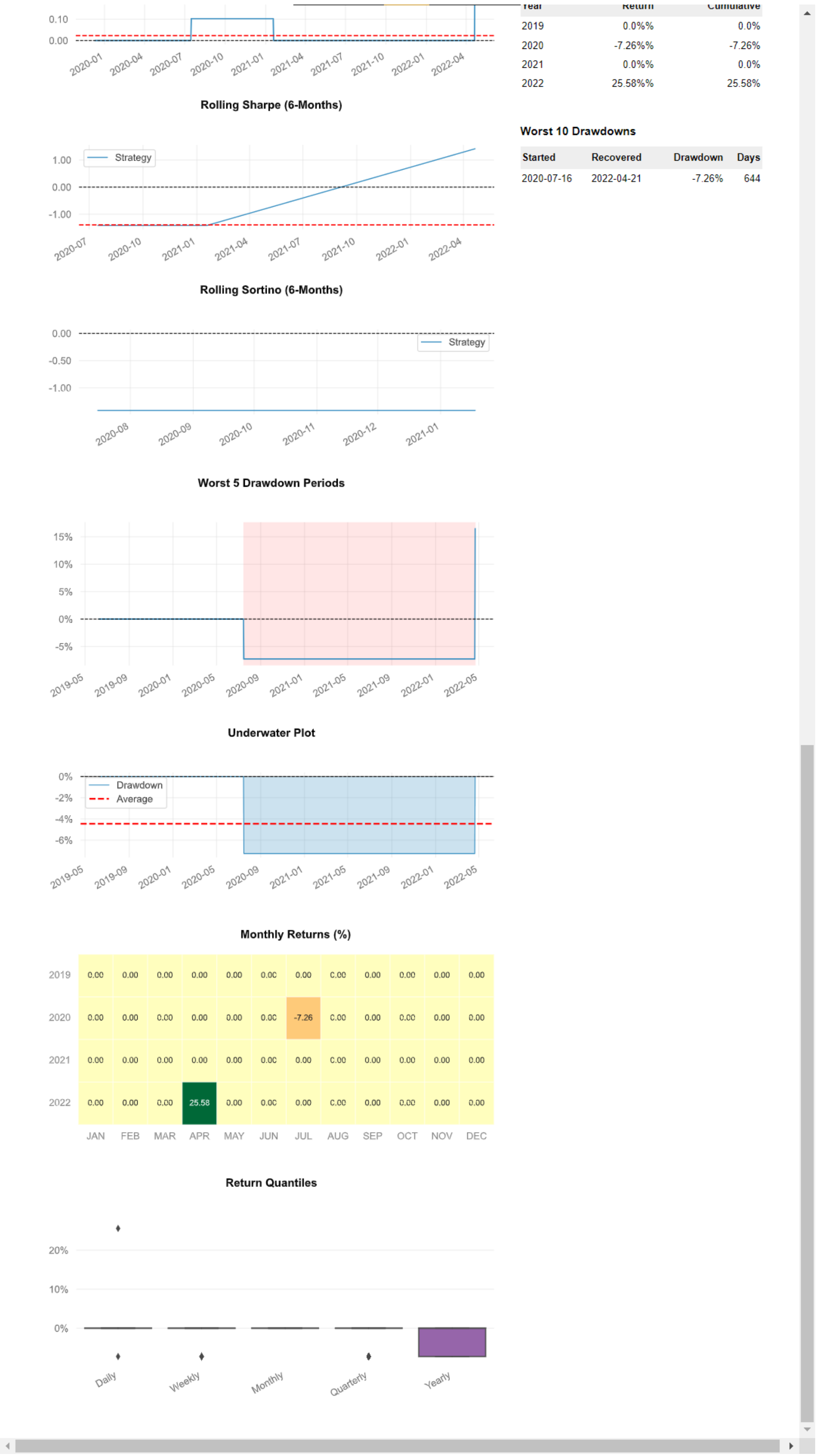
Year	Return	Cumulative
2019	0.0%%	0.0%
2020	-7.26%%	-7.26%
2021	0.0%%	0.0%
2022	25.58%%	25.58%

Worst 10 Drawdowns

Started	Recovered	Drawdown	Days
2020-07-16	2022-04-21	-7.26%	644

2019	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
2020	0.00	0.00	0.00	0.00	0.00	0.00	-7.26	0.00	0.00	0.00	0.00	0.00
2021	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00

(Part 2)



Year	Return	Cumulative
2019	0.0%%	0.0%
2020	-7.26%%	-7.26%
2021	0.0%%	0.0%
2022	25.58%%	25.58%

Worst 10 Drawdowns

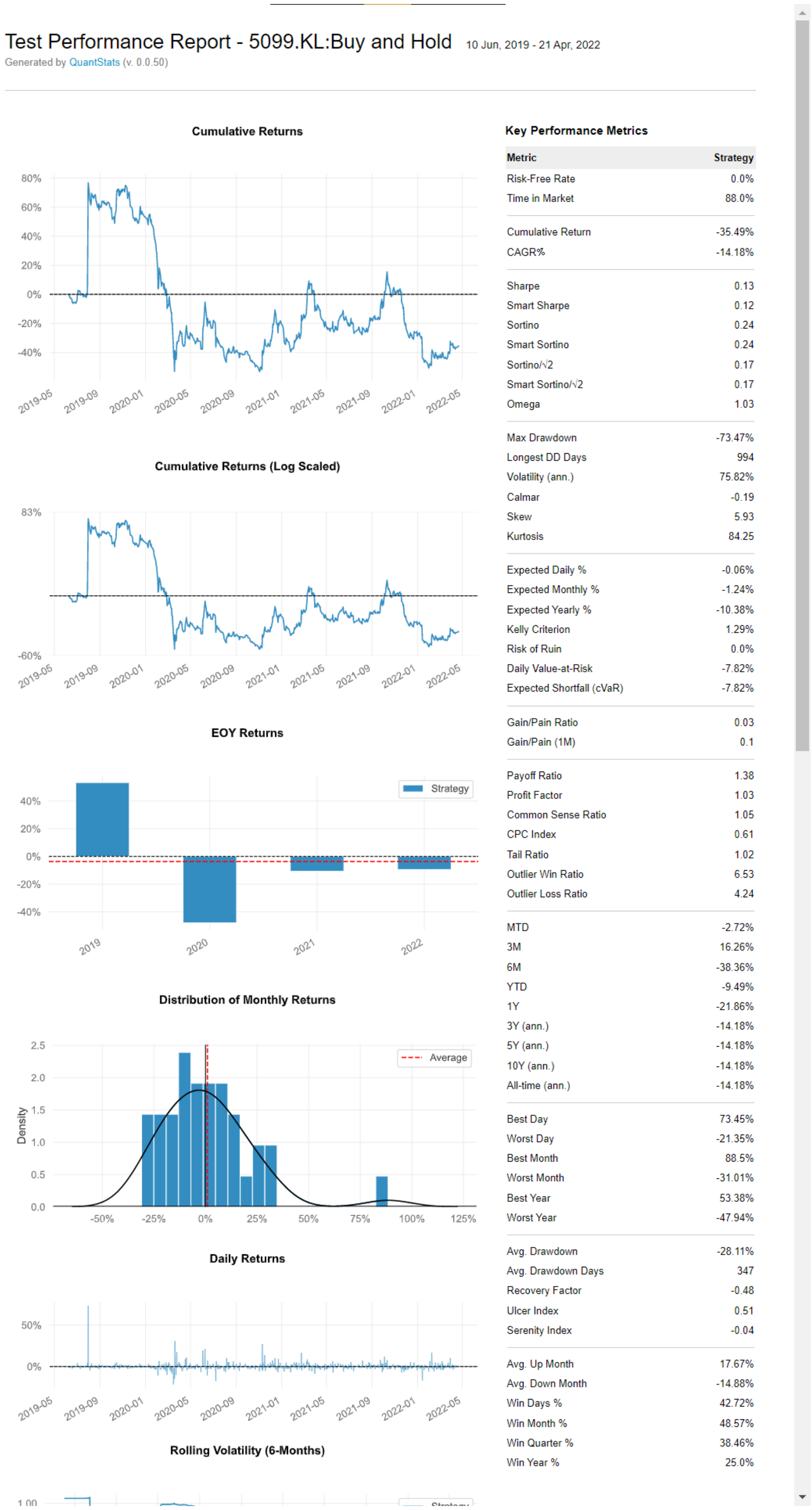
Started	Recovered	Drawdown	Days
2020-07-16	2022-04-21	-7.26%	644

Monthly Returns (%)

	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	SEP	OCT	NOV	DEC
2019	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
2020	0.00	0.00	0.00	0.00	0.00	0.00	-7.26	0.00	0.00	0.00	0.00	0.00
2021	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
2022	0.00	0.00	0.00	25.58	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00

(Part 3)

The performance report of Buy-and-Hold strategy on symbol 5099.KL



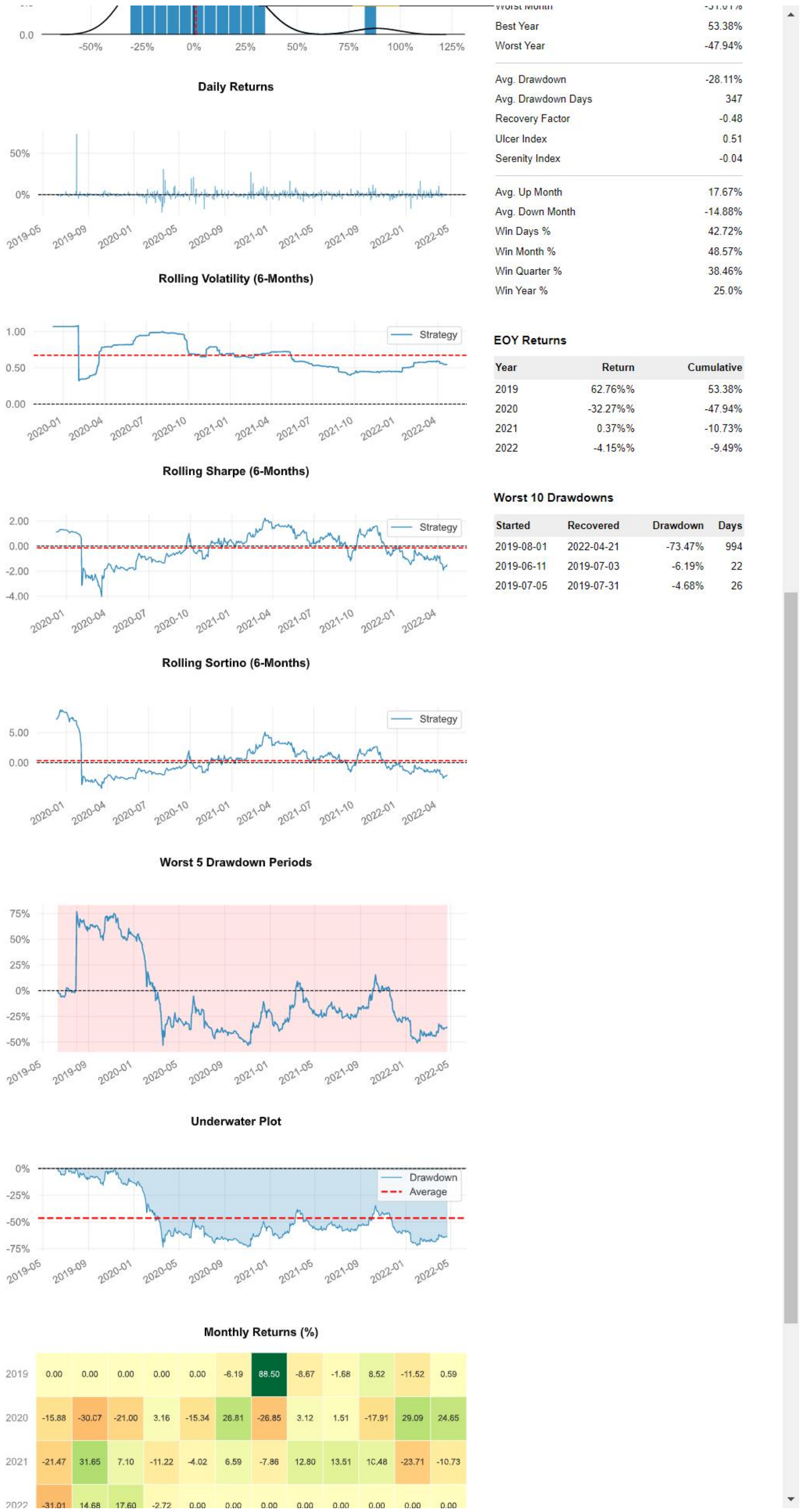
Test Performance Report - 5099.KL:Buy and Hold 10 Jun, 2019 - 21 Apr, 2022

Generated by QuantStats (v. 0.0.50)

Key Performance Metrics

Metric	Strategy
Risk-Free Rate	0.0%
Time in Market	88.0%
Cumulative Return	-35.49%
CAGR﹪	-14.18%
Sharpe	0.13
Smart Sharpe	0.12
Sortino	0.24
Smart Sortino	0.24
Sortino/√2	0.17
Smart Sortino/√2	0.17
Omega	1.03
Max Drawdown	-73.47%
Longest DD Days	994
Volatility (ann.)	75.82%
Calmar	-0.19
Skew	5.93
Kurtosis	84.25
Expected Daily %	-0.06%
Expected Monthly %	-1.24%
Expected Yearly %	-10.38%
Kelly Criterion	1.29%
Risk of Ruin	0.0%
Daily Value-at-Risk	-7.82%
Expected Shortfall (cVaR)	-7.82%
Gain/Pain Ratio	0.03
Gain/Pain (1M)	0.1
Payoff Ratio	1.38
Profit Factor	1.03
Common Sense Ratio	1.05
CPC Index	0.61
Tail Ratio	1.02
Outlier Win Ratio	6.53
Outlier Loss Ratio	4.24
MTD	-2.72%
3M	16.26%
6M	-38.36%
YTD	-9.49%
1Y	-21.86%
3Y (ann.)	-14.18%
5Y (ann.)	-14.18%
10Y (ann.)	-14.18%
All-time (ann.)	-14.18%
Best Day	73.45%
Worst Day	-21.35%
Best Month	88.5%
Worst Month	-31.01%
Best Year	53.38%
Worst Year	-47.94%
Avg. Drawdown	-28.11%
Avg. Drawdown Days	347
Recovery Factor	-0.48
Ulcer Index	0.51
Serenity Index	-0.04
Avg. Up Month	17.67%
Avg. Down Month	-14.88%
Win Days %	42.72%
Win Month %	48.57%
Win Quarter %	38.46%
Win Year %	25.0%

(Part 1)



Daily Returns

Rolling Volatility (6-Months)

Rolling Sharpe (6-Months)

Rolling Sortino (6-Months)

Worst 5 Drawdown Periods

Underwater Plot

Monthly Returns (%)

2019	0.00	0.00	0.00	0.00	0.00	-6.19	88.50	-8.67	-1.68	8.52	-11.52	0.59
2020	-15.88	-30.07	-21.00	3.16	-15.34	26.81	-26.85	3.12	1.51	-17.91	29.09	24.65
2021	-21.47	31.65	7.10	-11.22	-4.02	6.59	-7.86	12.80	13.51	10.48	-23.71	-10.73
2022	-31.01	14.68	17.60	-2.72	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00

Worst Month	-31.01%
Best Year	53.38%
Worst Year	-47.94%
Avg. Drawdown	-28.11%
Avg. Drawdown Days	347
Recovery Factor	-0.48
Ulcer Index	0.51
Serenity Index	-0.04
Avg. Up Month	17.67%
Avg. Down Month	-14.88%
Win Days %	42.72%
Win Month %	48.57%
Win Quarter %	38.46%
Win Year %	25.0%

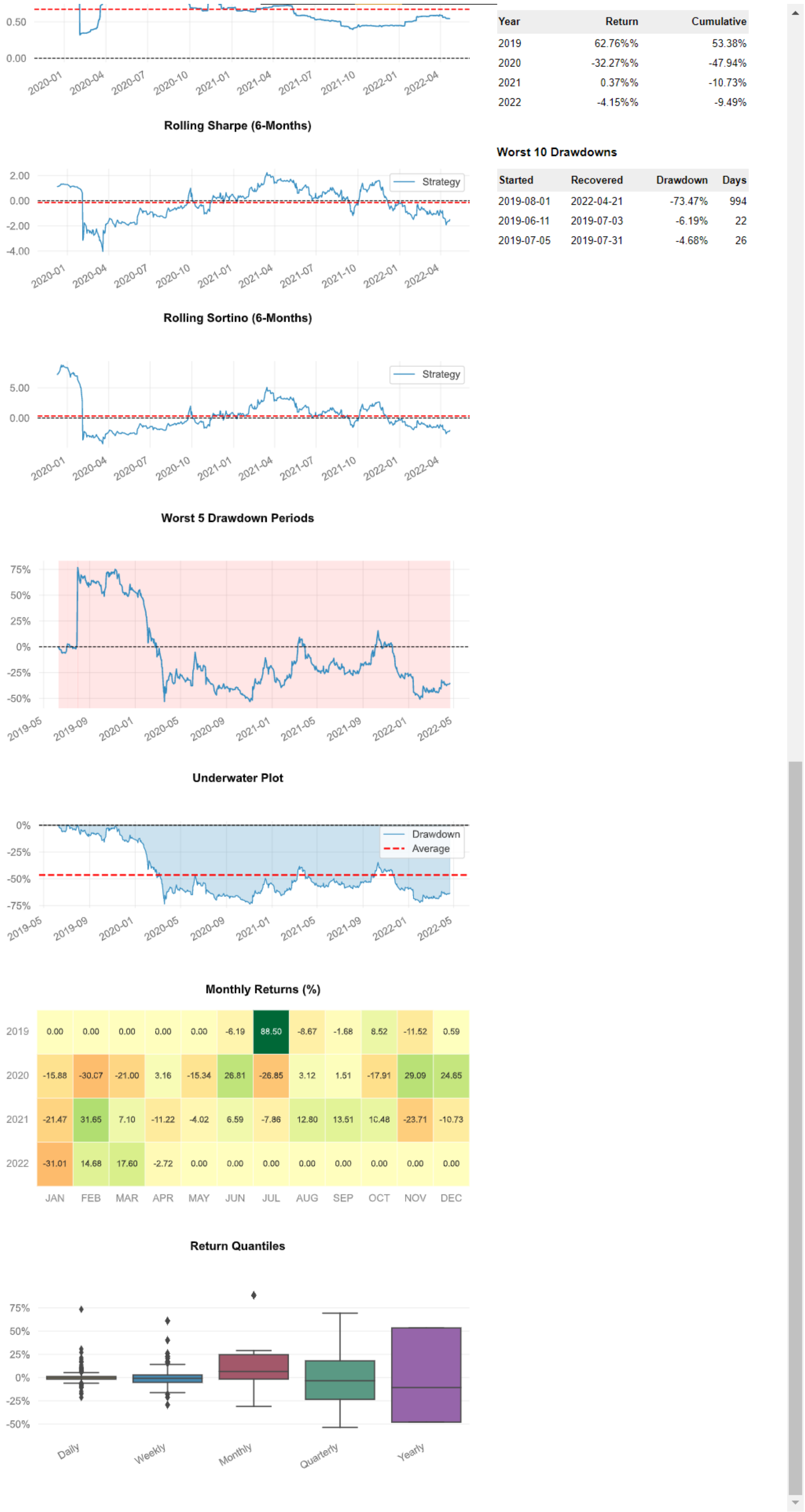
EOY Returns

Year	Return	Cumulative
2019	62.76%%	53.38%
2020	-32.27%%	-47.94%
2021	0.37%%	-10.73%
2022	-4.15%%	-9.49%

Worst 10 Drawdowns

Started	Recovered	Drawdown	Days
2019-08-01	2022-04-21	-73.47%	994
2019-06-11	2019-07-03	-6.19%	22
2019-07-05	2019-07-31	-4.68%	26

(Part 2)



Rolling Sharpe (6-Months)

Rolling Sortino (6-Months)

Worst 5 Drawdown Periods

Underwater Plot

Monthly Returns (%)

	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	SEP	OCT	NOV	DEC
2019	0.00	0.00	0.00	0.00	0.00	-6.19	88.50	-8.67	-1.68	8.52	-11.52	0.59
2020	-15.88	-30.07	-21.00	3.16	-15.34	26.81	-26.85	3.12	1.51	-17.91	29.09	24.65
2021	-21.47	31.65	7.10	-11.22	-4.02	6.59	-7.86	12.80	13.51	10.48	-23.71	-10.73
2022	-31.01	14.68	17.60	-2.72	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00

Return Quantiles

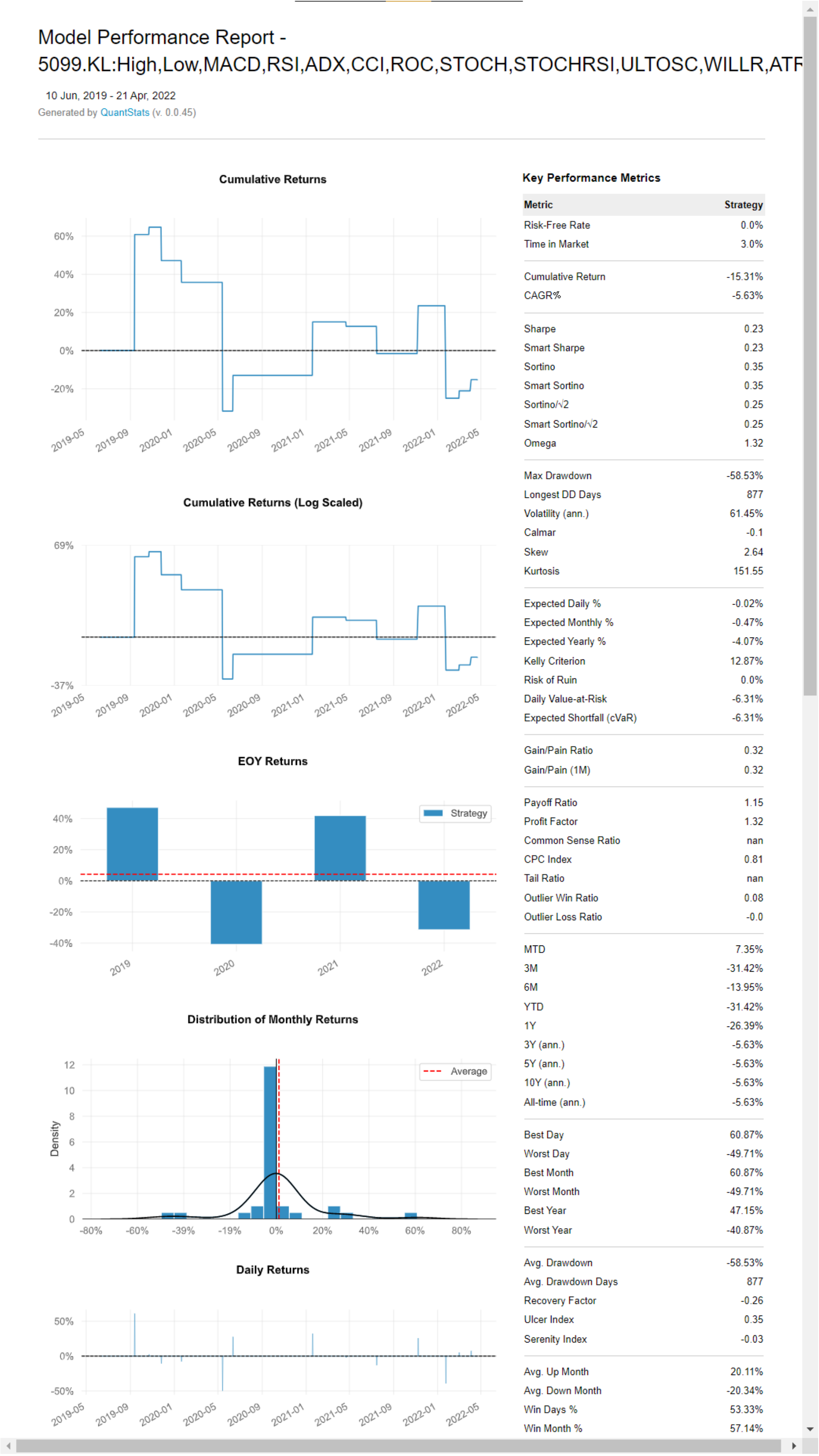
Year	Return	Cumulative
2019	62.76%%	53.38%
2020	-32.27%%	-47.94%
2021	0.37%%	-10.73%
2022	-4.15%%	-9.49%

Worst 10 Drawdowns

Started	Recovered	Drawdown	Days
2019-08-01	2022-04-21	-73.47%	994
2019-06-11	2019-07-03	-6.19%	22
2019-07-05	2019-07-31	-4.68%	26

(Part 3)

The performance report of proposed model on symbol 5099.KL



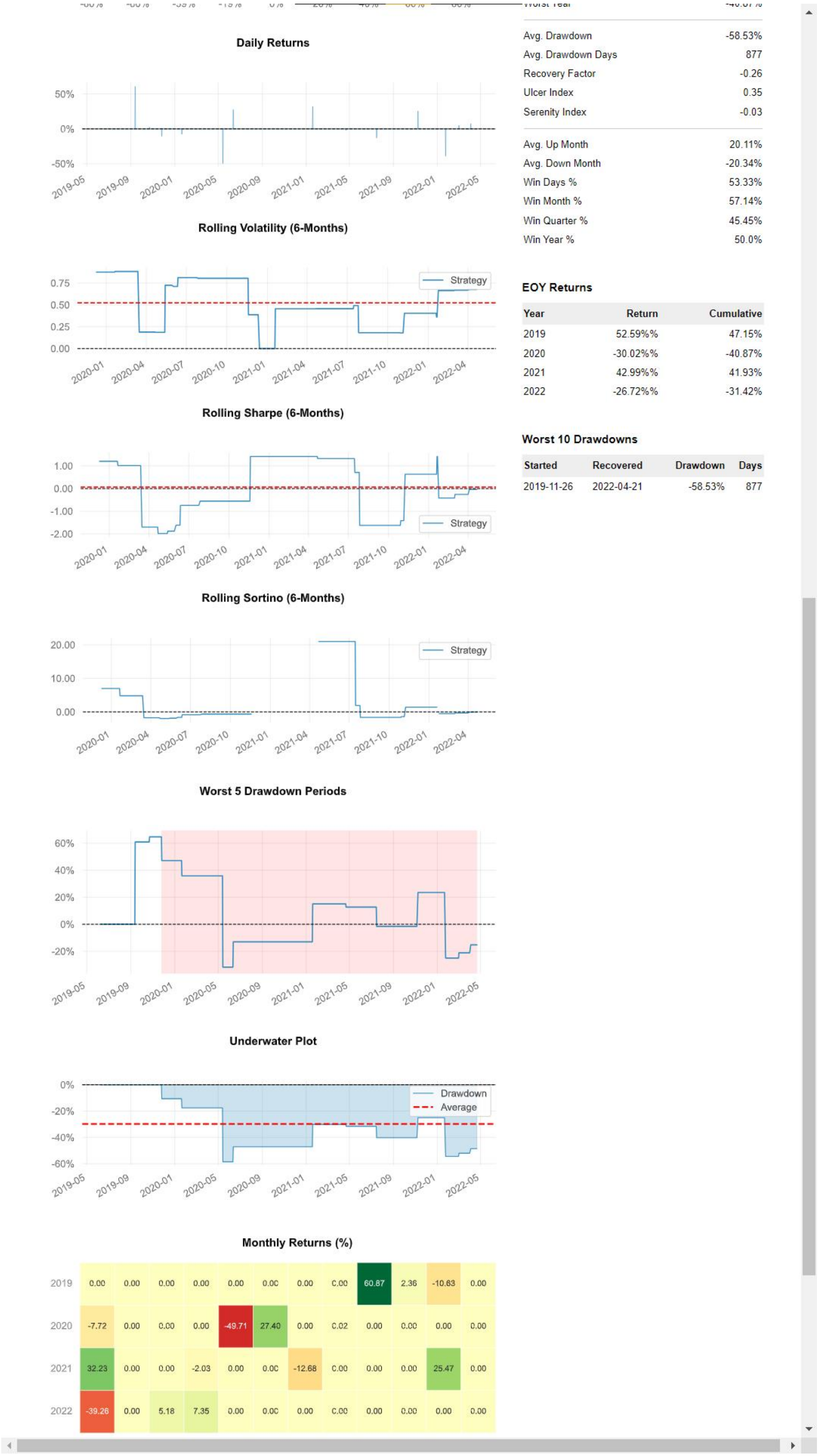
Model Performance Report -
5099.KL:High,Low,MACD,RSI,ADX,CCI,ROC,STOCH,STOCHRSI,ULTOSC,WILLR,ATF

10 Jun, 2019 - 21 Apr, 2022
Generated by QuantStats (v. 0.0.45)

Key Performance Metrics

Metric	Strategy
Risk-Free Rate	0.0%
Time in Market	3.0%
Cumulative Return	-15.31%
CAGR﹪	-5.63%
Sharpe	0.23
Smart Sharpe	0.23
Sortino	0.35
Smart Sortino	0.35
Sortino/√2	0.25
Smart Sortino/√2	0.25
Omega	1.32
Max Drawdown	-58.53%
Longest DD Days	877
Volatility (ann.)	61.45%
Calmar	-0.1
Skew	2.64
Kurtosis	151.55
Expected Daily %	-0.02%
Expected Monthly %	-0.47%
Expected Yearly %	-4.07%
Kelly Criterion	12.87%
Risk of Ruin	0.0%
Daily Value-at-Risk	-6.31%
Expected Shortfall (cVaR)	-6.31%
Gain/Pain Ratio	0.32
Gain/Pain (1M)	0.32
Payoff Ratio	1.15
Profit Factor	1.32
Common Sense Ratio	nan
CPC Index	0.81
Tail Ratio	nan
Outlier Win Ratio	0.08
Outlier Loss Ratio	-0.0
MTD	7.35%
3M	-31.42%
6M	-13.95%
YTD	-31.42%
1Y	-26.39%
3Y (ann.)	-5.63%
5Y (ann.)	-5.63%
10Y (ann.)	-5.63%
All-time (ann.)	-5.63%
Best Day	60.87%
Worst Day	-49.71%
Best Month	60.87%
Worst Month	-49.71%
Best Year	47.15%
Worst Year	-40.87%
Avg. Drawdown	-58.53%
Avg. Drawdown Days	877
Recovery Factor	-0.26
Ulcer Index	0.35
Serenity Index	-0.03
Avg. Up Month	20.11%
Avg. Down Month	-20.34%
Win Days %	53.33%
Win Month %	57.14%

(Part 1)



Avg. Drawdown	-58.53%
Avg. Drawdown Days	877
Recovery Factor	-0.26
Ulcer Index	0.35
Serenity Index	-0.03
Avg. Up Month	20.11%
Avg. Down Month	-20.34%
Win Days %	53.33%
Win Month %	57.14%
Win Quarter %	45.45%
Win Year %	50.0%

EOY Returns

Year	Return	Cumulative
2019	52.59%%	47.15%
2020	-30.02%%	-40.87%
2021	42.99%%	41.93%
2022	-26.72%%	-31.42%

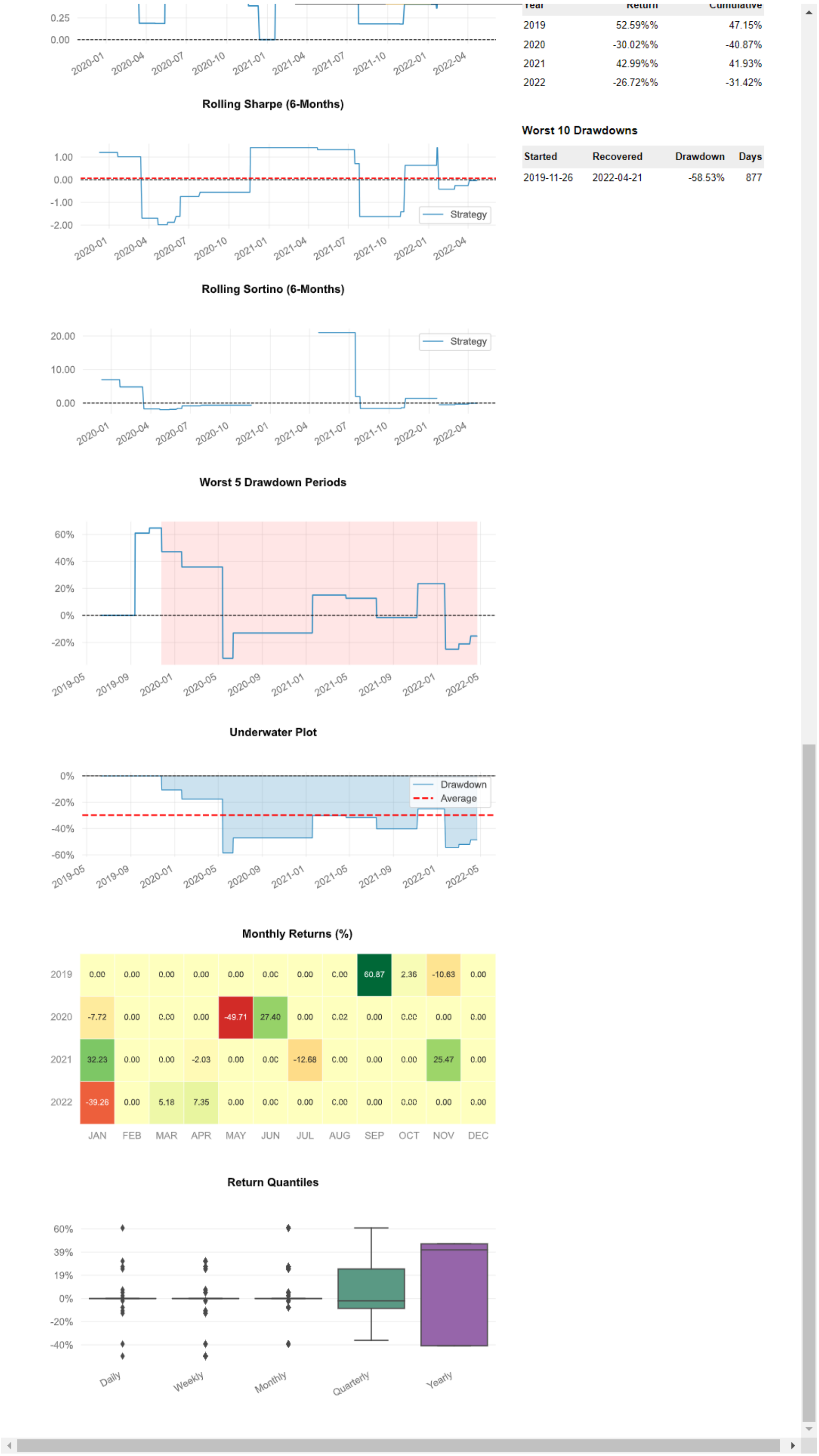
Worst 10 Drawdowns

Started	Recovered	Drawdown	Days
2019-11-26	2022-04-21	-58.53%	877

Monthly Returns (%)

2019	0.00	0.00	0.00	0.00	0.00	0.0C	0.00	C.00	60.87	2.36	-10.63	0.00
2020	-7.72	0.00	0.00	0.00	-49.71	27.40	0.00	C.02	0.00	0.00	0.00	0.00
2021	32.23	0.00	0.00	-2.03	0.00	0.0C	-12.68	C.00	0.00	0.00	25.47	0.00
2022	-39.26	0.00	5.18	7.35	0.00	0.0C	0.00	C.00	0.00	0.00	0.00	0.00

(Part 2)



Year	Return	Cumulative
2019	52.59%%	47.15%
2020	-30.02%%	-40.87%
2021	42.99%%	41.93%
2022	-26.72%%	-31.42%

Worst 10 Drawdowns

Started	Recovered	Drawdown	Days
2019-11-26	2022-04-21	-58.53%	877

Monthly Returns (%)

	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	SEP	OCT	NOV	DEC
2019	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	60.87	2.36	-10.63	0.00
2020	-7.72	0.00	0.00	0.00	-49.71	27.40	0.00	0.02	0.00	0.00	0.00	0.00
2021	32.23	0.00	0.00	-2.03	0.00	0.00	-12.68	0.00	0.00	0.00	25.47	0.00
2022	-39.26	0.00	5.18	7.35	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00

(Part 3)

APPENDIX C

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: Jan, 2022	**Study week no.:** 2
Student Name & ID: DESMOND CHEONG YONGHONG 18ACB04824	
Supervisor: TS DR. KU CHIN SOON	
Project Title: STOCK INDICATOR SCANNER CUSTOMIZATION TOOL USING DEEP REINFORCEMENT LEARNING	

1. WORK DONE

- Refine UI
- Implement watchlist feature
- Add watchlist UI

2. WORK TO BE DONE

- Implement generate report feature
- Add more agent actions and trading positions
- Add more indicators
- Change some processes to parallel
- Compare with other system
- Refine the DQN model to get better prediction result
- Refine the report

3. PROBLEMS ENCOUNTERED

- Large change of the UI.

4. SELF EVALUATION OF THE PROGRESS

- Good

Supervisor's signature

Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: Jan, 2022	**Study week no.:** 4
Student Name & ID: DESMOND CHEONG YONGHONG 18ACB04824	
Supervisor: TS DR. KU CHIN SOON	
Project Title: STOCK INDICATOR SCANNER CUSTOMIZATION TOOL USING DEEP REINFORCEMENT LEARNING	

1. WORK DONE

- Implement generate report feature
- Add more agent actions and trading positions

2. WORK TO BE DONE

- Change some processes to parallel
- Add more indicators
- Compare with other system
- Refine the DQN model to get better prediction result
- Refine the report

3. PROBLEMS ENCOUNTERED

- Change of the number of agent actions and trading positions cause the re-train of DQN model needed.

4. SELF EVALUATION OF THE PROGRESS

- Good

Supervisor's signature

Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: Jan, 2022	**Study week no.:** 6
Student Name & ID: DESMOND CHEONG YONGHONG 18ACB04824	
Supervisor: TS DR. KU CHIN SOON	
Project Title: STOCK INDICATOR SCANNER CUSTOMIZATION TOOL USING DEEP REINFORCEMENT LEARNING	

1. WORK DONE

- Change some processes to parallel

2. WORK TO BE DONE

- Add more indicators
- Compare with other system
- Refine the DQN model to get better prediction result
- Refine the report

3. PROBLEMS ENCOUNTERED

- Decide on method to implement parallel processing in Python.
- Try to implement multithreading in Python.

4. SELF EVALUATION OF THE PROGRESS

- Need improvements

Supervisor's signature

Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: Jan, 2022	**Study week no.:** 8
Student Name & ID: DESMOND CHEONG YONGHONG 18ACB04824	
Supervisor: TS DR. KU CHIN SOON	
Project Title: STOCK INDICATOR SCANNER CUSTOMIZATION TOOL USING DEEP REINFORCEMENT LEARNING	

1. WORK DONE

- Add more indicators

2. WORK TO BE DONE

- Update database
- Compare with other system
- Refine the DQN model to get better prediction result
- Refine the report

3. PROBLEMS ENCOUNTERED

- Search source that provides financial data for some indicators calculation.
- The calculated values of indicators are different with other source.

4. SELF EVALUATION OF THE PROGRESS

- Slow progress, need improvements

Supervisor’s signature

Student’s signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: Jan, 2022	**Study week no.:** 10
Student Name & ID: DESMOND CHEONG YONGHONG 18ACB04824	
Supervisor: TS DR. KU CHIN SOON	
Project Title: STOCK INDICATOR SCANNER CUSTOMIZATION TOOL USING DEEP REINFORCEMENT LEARNING	

1. WORK DONE

- Update database

2. WORK TO BE DONE

- Compare with other system
- Refine the DQN model to get better prediction result
- Refine the report

3. PROBLEMS ENCOUNTERED

- Large database.

4. SELF EVALUATION OF THE PROGRESS

- Slow progress, need improvements

Supervisor's signature

Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: Jan, 2022	**Study week no.:** 12
Student Name & ID: DESMOND CHEONG YONGHONG 18ACB04824	
Supervisor: TS DR. KU CHIN SOON	
Project Title: STOCK INDICATOR SCANNER CUSTOMIZATION TOOL USING DEEP REINFORCEMENT LEARNING	

1. WORK DONE

- Train the DQN model
- Compare with other system
- Refine the report

2. WORK TO BE DONE

- Refine the DQN model to get better prediction result

3. PROBLEMS ENCOUNTERED

- The training of the model very time consuming.

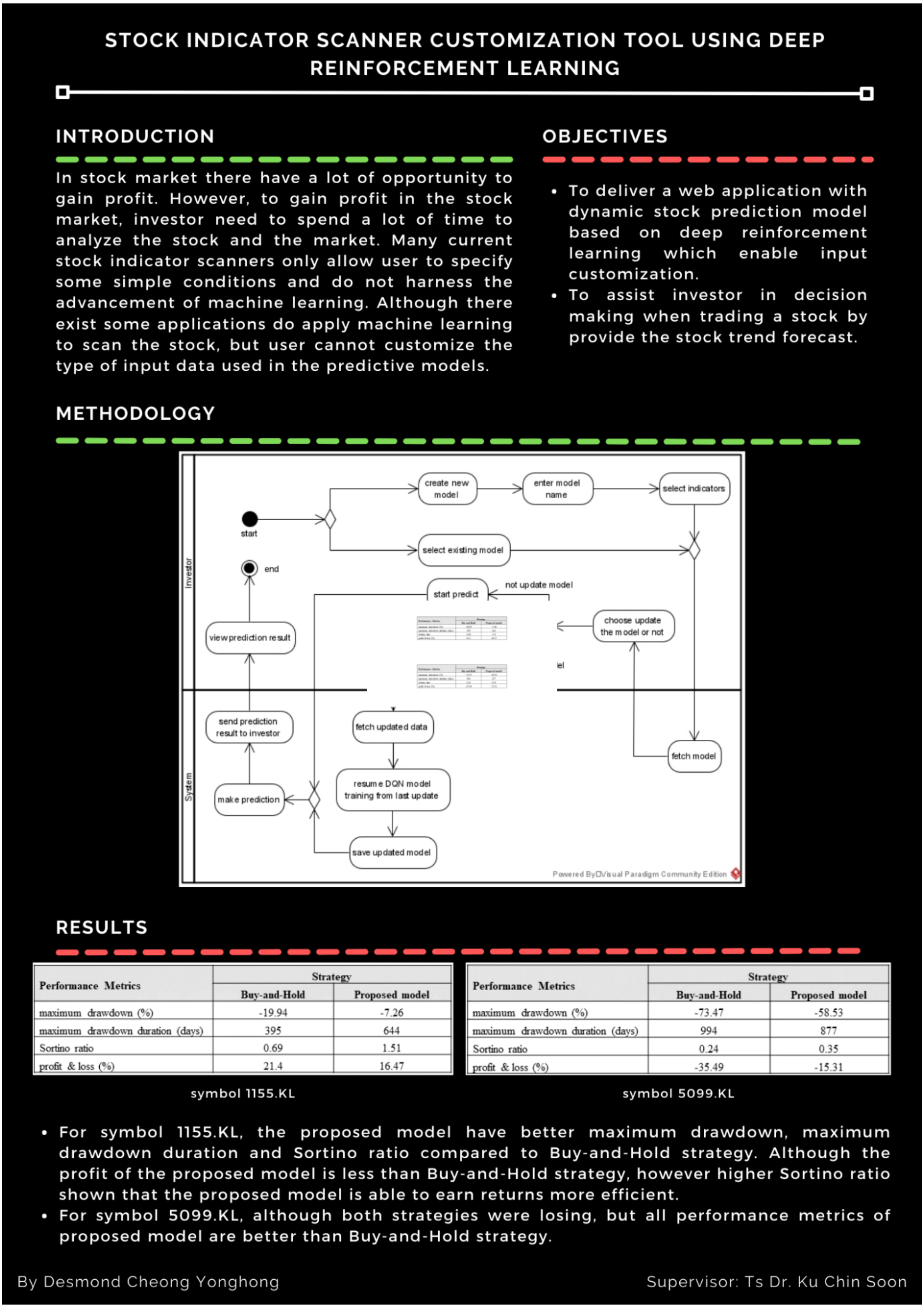
4. SELF EVALUATION OF THE PROGRESS

- Slow progress, need improvements

Supervisor's signature

Student's signature

POSTER



STOCK INDICATOR SCANNER CUSTOMIZATION TOOL USING DEEP REINFORCEMENT LEARNING

INTRODUCTION

In stock market there have a lot of opportunity to gain profit. However, to gain profit in the stock market, investor need to spend a lot of time to analyze the stock and the market. Many current stock indicator scanners only allow user to specify some simple conditions and do not harness the advancement of machine learning. Although there exist some applications do apply machine learning to scan the stock, but user cannot customize the type of input data used in the predictive models.

OBJECTIVES

- To deliver a web application with dynamic stock prediction model based on deep reinforcement learning which enable input customization.
- To assist investor in decision making when trading a stock by provide the stock trend forecast.

METHODOLOGY

RESULTS

Performance Metrics	Strategy	
	Buy-and-Hold	Proposed model
maximum drawdown (%)	-19.94	-7.26
maximum drawdown duration (days)	395	644
Sortino ratio	0.69	1.51
profit & loss (%)	21.4	16.47

symbol 1155.KL

Performance Metrics	Strategy	
	Buy-and-Hold	Proposed model
maximum drawdown (%)	-73.47	-58.53
maximum drawdown duration (days)	994	877
Sortino ratio	0.24	0.35
profit & loss (%)	-35.49	-15.31

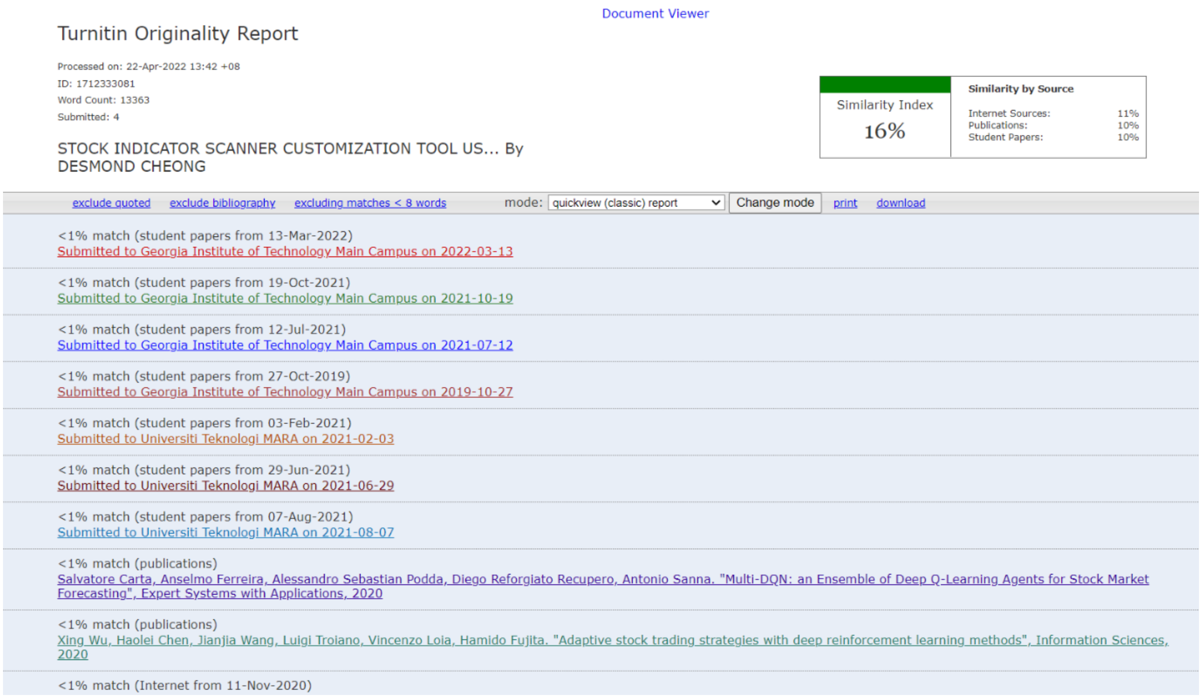
symbol 5099.KL

- For symbol 1155.KL, the proposed model have better maximum drawdown, maximum drawdown duration and Sortino ratio compared to Buy-and-Hold strategy. Although the profit of the proposed model is less than Buy-and-Hold strategy, however higher Sortino ratio shown that the proposed model is able to earn returns more efficient.
- For symbol 5099.KL, although both strategies were losing, but all performance metrics of proposed model are better than Buy-and-Hold strategy.

By Desmond Cheong Yonghong

Supervisor: Ts Dr. Ku Chin Soon

PLAGIARISM CHECK RESULT



Document Viewer

Turnitin Originality Report

Processed on: 22-Apr-2022 13:42 +08
ID: 1712333081
Word Count: 13363
Submitted: 4

STOCK INDICATOR SCANNER CUSTOMIZATION TOOL US... By DESMOND CHEONG

Similarity Index	Similarity by Source	
16%	Internet Sources:	11%
	Publications:	10%
	Student Papers:	10%

<1% match (student papers from 13-Mar-2022)
Submitted to Georgia Institute of Technology Main Campus on 2022-03-13

<1% match (student papers from 19-Oct-2021)
Submitted to Georgia Institute of Technology Main Campus on 2021-10-19

<1% match (student papers from 12-Jul-2021)
Submitted to Georgia Institute of Technology Main Campus on 2021-07-12

<1% match (student papers from 27-Oct-2019)
Submitted to Georgia Institute of Technology Main Campus on 2019-10-27

<1% match (student papers from 03-Feb-2021)
Submitted to Universiti Teknologi MARA on 2021-02-03

<1% match (student papers from 29-Jun-2021)
Submitted to Universiti Teknologi MARA on 2021-06-29

<1% match (student papers from 07-Aug-2021)
Submitted to Universiti Teknologi MARA on 2021-08-07

<1% match (publications)
Salvatore Carta, Anselmo Ferreira, Alessandro Sebastian Podda, Diego Reforgiato Recupero, Antonio Sanna. "Multi-DQN: an Ensemble of Deep Q-Learning Agents for Stock Market Forecasting", Expert Systems with Applications, 2020

<1% match (publications)
Xing Wu, Haolei Chen, Jianjia Wang, Luigi Troiano, Vincenzo Loia, Hamido Fujita. "Adaptive stock trading strategies with deep reinforcement learning methods", Information Sciences, 2020

<1% match (Internet from 11-Nov-2020)

Universiti Tunku Abdul Rahman			
Form Title: Supervisor's Comments on Originality Report Generated by Turnitin for Submission of Final Year Project Report (for Undergraduate Programmes)			
Form Number: FM-IAD-005	Rev No.: 0	Effective Date: 01/10/2013	Page No.: 1of 1

FACULTY OF INFORMATION AND COMMUNICATION TECHNOLOGY

Full Name(s) of Candidate(s)	DESMOND CHEONG YONGHONG
ID Number(s)	18ACB04824
Programme / Course	CS
Title of Final Year Project	STOCK INDICATOR SCANNER CUSTOMIZATION TOOL USING DEEP REINFORCEMENT LEARNING

Similarity	**Supervisor's Comments (Compulsory if parameters of originality exceed the limits approved by UTAR)**
Overall similarity index: _16_ **%** **Similarity by source** Internet Sources: 11 % Publications: 10 % Student Papers: 10 %	
Number of individual sources listed of more than 3% similarity: 0	
Parameters of originality required, and limits approved by UTAR are as Follows: **(i) Overall similarity index is 20% and below, and** **(ii) Matching of individual sources listed must be less than 3% each, and** **(iii) Matching texts in continuous block must not exceed 8 words** ***Note: Parameters (i) – (ii) shall exclude quotes, bibliography and text matches which are less than 8 words.***	

Note: Supervisor/Candidate(s) is/are required to provide softcopy of full set of the originality report to Faculty/Institute

Based on the above results, I hereby declare that I am satisfied with the originality of the Final Year Project Report submitted by my student(s) as named above.

____
Signature of Supervisor

Name: Ku Chin Soon

Date: 22/04/2022

____
Signature of Co-Supervisor

Name: ______________________

Date: ______________________

FYP 2 CHECKLIST

UNIVERSITI TUNKU ABDUL RAHMAN

FACULTY OF INFORMATION & COMMUNICATION TECHNOLOGY
(KAMPAR CAMPUS)

CHECKLIST FOR FYP2 THESIS SUBMISSION

Student Id	18ACB04824
Student Name	DESMOND CHEONG YONGHONG
Supervisor Name	TS DR. KU CHIN SOON

TICK (√)	**DOCUMENT ITEMS** Your report must include all the items below. Put a tick on the left column after you have checked your report with respect to the corresponding item.
	Front Plastic Cover (for hardcopy)
√	Title Page
√	Signed Report Status Declaration Form
√	Signed FYP Thesis Submission Form
√	Signed form of the Declaration of Originality
√	Acknowledgement
√	Abstract
√	Table of Contents
√	List of Figures (if applicable)
√	List of Tables (if applicable)
√	List of Symbols (if applicable)
√	List of Abbreviations (if applicable)
√	Chapters / Content
√	Bibliography (or References)
√	All references in bibliography are cited in the thesis, especially in the chapter of literature review
√	Appendices (if applicable)
√	Weekly Log
√	Poster
√	Signed Turnitin Report (Plagiarism Check Result - Form Number: FM-IAD-005)
√	I agree 5 marks will be deducted due to incorrect format, declare wrongly the ticked of these items, and/or any dispute happening for these items in this report.

*Include this form (checklist) in the thesis (Bind together as the last page)

I, the author, have checked and confirmed all the items listed in the table are included in my report.

____
(Signature of Student)

FYP 2 CHECKLIST

Date: 22 APRIL 2022