alamSYS: Development of Stock Market Price Forecasting System Using Dynamic Mode Decomposition, Long Short-Term Memory with Arnaud Legoux Moving Average Convergence-Divergence Integration

A Special Problem
Presented to
the Faculty of the Division of Physical Sciences and Mathematics
College of Arts and Sciences
University of the Philippines Visayas
Miag-ao, Iloilo

In Partial Fulfillment
of the Requirements for the Degree of
Bachelor of Science in Computer Science by

OLARTE, John Markton M.

Nilo C. Araneta
Adviser

June 2023

Contents

List of Figures

Chapter 1

Introduction

1.1 Background and Rationale

The stock market is a type of market that allows businesses to raise capital by selling stock shares to investors. These shares represent a portion of the company's ownership and entitle the holder to a portion of the company's profits as well as voting rights. The stock exchange also serves as a marketplace for investors to buy and sell these shares, allowing for the efficient trading of company ownership. The stock market plays an important role in the growth and development of the economy by allowing companies to raise capital and investors to buy and sell shares (Chen, 2022; The Economic Times, n.d.).

The stock market, contrary to popular belief, is not a form of gambling. It necessitates a significant amount of analytical thinking and risk management, and the returns are determined by supply and demand for a specific stock, rather than false promises or assurances. In other words, rather than being a scam or a gamble, the stock market is a legitimate platform for investing and generating returns (Schwab-Pomerantz, 2021; Adams, 2022; Summers, 2022).

1.1.1 The Philippine Stock Exchange (PSE)

The Philippine Stock Exchange (PSE), Inc. is the country's official stock exchange market. It is a non-stock company founded in 1992 that manages and operates the country's stock market. Individuals who are registered with the PSE can participate in market exchanges (The Philippine Stock Exchange, Inc., n.d.-a).

Furthermore, the Philippine Stock Exchange Index (PSEI) is the main index of the PSE. (PSEI). The PSEI is a market capitalization-weighted price index composed of the PSE's 30 largest and most actively traded companies. These businesses have been pre-selected based on strict criteria such as liquidity and market capitalization. The PSEI is frequently used as a proxy for the overall performance of the Philippine stock market. (Bangko Sentral ng Pilipinas, n.d.) These PSEI companies are often referred to as blue-chip companies because they are typically large, well-established companies with a track record of strong financial performance. In total the PSE has 286 companies listed as of October 2022, offering investors a diverse range of investment opportunities (Fayed, 2022; The Philippine Stock Exchange, Inc., n.d.-b).

1.1.2 Economic Relevance and Benefits of Stock Market Investment

The stock market is widely acknowledged to play an important role in economic growth because it allocates and provides capital to businesses, which drives economic activity and growth. This is evident from the fact that stock market performance is frequently correlated with the gross domestic product (GDP) of the country. (Trade Brains, 2022; Hall, 2022; Bae & Kang, 2017) Furthermore, historical stock price trends can provide insight into broader economic movements (Campbell, 2021).

In a study conducted by Balaba (2017), they discovered that the stock market has a positive impact on the Philippines' economy. The study's findings showed that as the stock market rose, the unemployment rate fell. This is because the performance of the stock market leads to job creation, which in turn leads to economic growth. This, in turn, drives economic growth. This relationship was observed in the Philippines from 2007 to 2017.

1.1.3 Benefits of Investing for the Individual

Individuals in the Philippines can trade shares of publicly traded companies on the Philippine Stock Exchange. Investing in the stock market can provide several advantages to an individual, including:

(a) Protects an individual's money from inflation: Inflation in the Philippines was 6.9% as of September 2022 (Trading Economics, n.d.), while savings account deposit interest rates are only 1-3% annually, (Bureau of the Treasury Bangko Sentral ng Pilipinas, n.d.). This means that savings in deposit banks may not keep pace with inflation, potentially reducing an individual's purchasing power over time. (Royal Bank of Canada Direct Investing Inc., n.d.; EdwardJones, n.d.).

(b) Capital growth opportunities: Investing in the stock market can provide individuals with the opportunity for significant capital growth without the need for direct investment involvement in business operations. This may benefit individuals. Students and working professionals, for example, can increase their capital while remaining focused on their studies or careers. (U.S. Securities and Exchange Commission, n.d.).

1.1.4 Utilization of Machine Learning in Stock Market Trading

In recent years, there has been a surge in interest in the use of machine learning. Learning techniques for predicting stock market movement in the short and long term. As a result, numerous studies and practical applications investigating the use of machine learning in stock market prediction have been conducted.

These efforts aim to improve prediction accuracy and assist investors in making informed decisions (Kumbure, Lohrmann, Luukka, & Porras, 2022; Strader, Rozycki, Root, & Huang, 2020; Soni, Tewari, & Krishnan, 2022; Rea, 2020; Guo, 2022). In this regard, one of the common techniques used is the Long Short-Term Memory (LSTM). LSTM is a deep learning model that is widely used to forecast the stock market. A study by Budiharto (2021) found that LSTM was effective in predicting the Indonesian stock market with 95% accuracy using a short-term data. Which suggests that LSTM can be a useful tool for making short-term stock market forecasts.

The use of Dynamic Mode Decomposition (DMD) for predicting stock market price trends has recently gained traction in the financial industry. DMD is a mathematical method for identifying patterns and trends in large data sets, such as stock market data. It is possible to make more accurate predictions about future stock price movements by applying DMD to stock market data. This can help investors make more informed investment decisions and potentially generate higher returns. However, a study by Lu and Tartakovsky (2020) found that DMD is faster than Proper Orthogonal Decomposition, but it is less accurate.

Other studies have shown that DMD can be effectively applied to the Turkish and Indian stock markets to predict market price trends (Savaș, 2017; Kuttichira, Gopalakrishnan, Menon, & Soman, 2017). These studies show that DMD is simple to implement and can be used as a useful enhancer for making stock market predictions.

1.2 Statement of the Problem

Economic growth in the Philippines is expected to slow in the coming years as a result of the global pandemic, high inflation, and low employment rates (Alegado, Lopez, & Calonzo, 2022; Canto & Romano, 2022; Reuters, 2022).

The lack of free and publicly available stock market predictive systems or tools currently creates a gap in the information available to the public when compared to large private individuals or institutions. These large institutions have the resources to spend a significant amount of money on stock market research, giving them a significant advantage in the investing market. Where, the public is disadvantaged by this lack of access to the same information (Kim, 2022).

Furthermore, the lack of publicly available stock market prediction tools can lead to individuals, particularly first-time investors, making unwise investment decisions, resulting in significant losses and discouragement from investing in the stock market. This is a significant issue because the number of local investors in the Philippine Stock Exchange is already quite small, accounting for only about 1% of the total population. In addition, there has been a significant decline in foreign investment in the Philippines in recent years (Business World, 2022), leading to a corresponding decline in investment volume. As suggested in the study of Balaba (2017), this is expected to have a negative multiplier effect on the country's economic development in the future.

As a result, the creation of a publicly available, simple-to-use, and accurate stock market price trend prediction system could aid in closing the information gap and leveling the playing field for individual investors. This system could help to increase transparency and fairness in the stock market by providing the public with timely and reliable information, resulting in more informed and confident investing decisions and, ultimately, a more stable and prosperous market. Furthermore, such a system could help to increase individual investor participation in the market, resulting in a more diverse and stable market overall. (Statista Research Department, 2022; Commission on Population and Development, 2021).

However, despite the clear and functional benefits of investing in the stock market, many Filipinos remain hesitant to do so for the following reasons:

(a) The difficulties that come with learning the fundamentals of effective stock investing.

(b) The time-consuming nature of technical and fundamental analysis, especially for students and working people on a tight schedule; and

(c) The increased financial risk associated with stock market volatility, as well as the potential for emotional decision-making to jeopardize investments.

These factors *(along with other external and internal factors not listed above)* contribute to a lack of confidence and understanding among potential investors, making it difficult for them to take advantage of the opportunities offered by the stock market.

As such the development of this system, aims to address the following:

(a) The lack of free and publicly available stock market prediction systems or tools.

(b) The time and resources required to study complex traditional market analysis tools, such as fundamental and technical analysis.

(c) The potential for inaccurate market decisions leading to significant investment losses; and

(d) The hesitancy of the Filipino public to begin investing in the Philippine stock market.

1.3 Significance of the Study

The significance of this particular problem lies in the developed system to greatly benefit the stock market, individual investors, and the economy as a whole. Contributions of the system to data-driven investing, financial protection and management, and economic development could provide a valuable resource for investors while also promoting financial stability and growth. Furthermore, the creation of publicly accessible data-driven investing tools or systems may enable more Filipinos to participate in the market and take control of their own financial future. Overall, this special problem has the potential to have a significant impact on the Philippine stock market and economy.

Specifically, this study is significant for the following reasons:

(a) The development of the alamSYS aims to provide the following benefits to the Filipino people:

    1. Access to simplified yet accurate information – The proposed system could provide Filipino investors with fast, accurate, and relevant information necessary for effective decision making in the stock market. Using a deep learning model such as LSTM, the system could provide users with the two most important pieces of information: which stocks to buy, and which stocks to sell. This simplified investing model could help investors to make informed decisions and navigate the stock market with confidence.

    2. Provide an application interface to facilitate data-driven market decisions – The system could provide users with an intuitive and user-friendly application interface to facilitate data-driven investment decisions, particularly during times when the market is unpredictable or experiencing a downturn. Whereas traditional market analysis tools may not be sufficient to navigate these challenging conditions, the system's forecasting model could provide investors with the insights and guidance they need to make informed and wise decisions. Which would help to promote confidence and stability in the market, even during times of uncertainty.

    3. A platform for accessible stock market investment – The system aims to provide all investors, regardless of their investment knowledge, educational attainment, and societal status, with a platform for participating in the stock market. By offering a simplified yet accurate model for investment decision making, the system could empower users to make informed decisions and invest with confidence. This could help to democratize access to the stock market and promote financial inclusion for all Filipinos.

(b) The development of the alamSYS, aims to provide the following benefits to the future developers or researchers:

1. Extension of functionality to other financial markets – The system can be easily adapted or expanded to address related problems in other financial markets, such as investing in government bonds or personal finance management. This flexibility and versatility could make the system a valuable tool for a wide range of investment and financial management scenarios.
2. Testing of new trading algorithms and other machine learning models – The system provides a platform for introducing and testing new data-driven trading algorithms and machine learning models. This could allow future researchers and developers to continually improve the system and keep it at the forefront of data-driven investing technology.
3. Development of a graphical user interface – To further improve the public accessibility of the system, a user-friendly graphical user interface can be developed as a web or mobile application. This could make the system easy to use and intuitive for all users, regardless of their technical expertise.

(c) The development of the alamSYS could help to stimulate economic recovery and development in the country by increasing the number of local investors. As discussed in previous sections, the benefits of the system could encourage more people to invest in the stock market, leading to a multiplier effect that could benefit the economy in several ways. For instance, the increased participation in the market could lead to the creation of jobs and a lowering of unemployment rates. Additionally, the influx of capital into the market could drive fast developments and innovations in various industries. Finally, the increased consumer spending that results from successful investing, stimulates economic growth as well. Overall, the development of the alamSYS could have a positive and far-reaching impact on the economy of the Philippines.

1.4 Objectives

This special problem aims to develop a system that makes investing easier, more publicly available, data-driven, and more approachable by minimizing both the time required for stock price trend analysis, and potential financial risk by using DMD-LSTM and integrate Arnaud Legoux Moving Average Convergence-Divergence (ALMACD) as a trading algorithm. More specifically, it aims to do following:

(a) Develop a Data Preprocessor. Which includes a Data Collector Module (DCM), which collects the end-of-day historical data of a stock from Mondays to Fridays. The data collected is then processed by the Data Processor Module (DPM), which applies the deep learning model and integrate the trading algorithm to the data. Finally, the processed data is given to the Database Updater Module (DUM).

(b) Develop a RESTful API, referred to as alamSYS, using the combination of Python's FastAPI and MongoDB for API endpoints and database, respectively.

Specifically, this was done by doing the following:

1. Develop the following API endpoints:

   1.1 `Home` – This API endpoint outputs a welcome message. Which should inform the user that they have successfully connected to the alamAPI.

   1.2 `Stocks to Buy` – This API endpoint outputs a list of suggested stocks to buy based from the current market price and the predicted price up-trend.

   1.3 `Stocks to Sell` – This API endpoint outputs a list of suggested stocks to sell based from the current market price and the predicted price down-trend.

   1.4 `Stocks Info` - This API endpoint outputs a list of stocks included in the alamSYS and their corresponding information.

1.5 `ML Model Info` - This API endpoint outputs a list of the Machine Learning Models used in the alamSYS and their corresponding information.

1.6 `Stocks Risks Info` - This API endpoint outputs a list of the stocks included in the alamSYS and their corresponding risks values based on value at risk (%), volatility (%), and drawdown (%).

2. Develop a database that stores the results provided by the DPM, and other essential data such as stock information, deep learning model information, and stock risks information about the stock market that is needed to be provided.

(c) Develop a Stock Market Price Trend Forecasting Deep Learning Models by utilizing the dynamic modes in DMD as an additional input parameter to an LSTM model. Afterwards, integrate the forecasting with ALMACD as a trading algorithm and basis for entry and exit positions.

(d) Finally, develop a mobile-based test application, which from hereon maybe referred to as: alamAPP, to showcase the main functionalities of the developed RESTful API. Specifically which stocks to buy and to sell for a given period of time.

1.5 Scope and Limitations

This study was limited only within the companies listed in the Philippine Stock Exchange. Specifically, 20 high volume trade stocks from the year 2021 to 2022 were selected, which are as follows: (1) MEG, (2) JGS, (3) BDO, (4) FGEN, (5) ICT, (6) ALI, (7) SMC, (8) TEL, (9) GLO, (10) BLOOM, (11) RLC, (12) MER, (13) AC, (14) PGOLD, (15) LTG, (16) MPI, (17) AP, (18) RRHI, (19) URC, and (20) PSE Index will be included in the system, instead of the total 286 listed under the Philippine Stock Exchange.

Chapter 2

Review of Related Works and Literature

One of the challenges facing investors in the Philippine Stock Market is the limited availability of resources and tools for making market decisions. In contrast, other countries have begun implementing machine learning techniques for stock market prediction and analysis, which allows for more accurate decision-making and reduces the risk of poor investment outcomes. As a result, these countries are likely to experience better returns on their investments.

In this literature review, the following general topics are reviewed, discussed, and synthesized: (a) Integration of Machine Learning based Trading Algorithms; and (b) Utilization of Dynamic Mode Decomposition on the Financial markets.

2.1 Integration of Machine Learning based Trading Algorithms

Stock market analysis is crucial for effective risk management. This involves using various methods, such as technical and fundamental analysis, to make informed decisions for investors and traders. In recent years, the growth of computing power and resources has led to the increasing use of machine learning techniques for stock market prediction and analysis. These advances help companies better predict upcoming market trends and make more informed decisions.

The integration of machine learning algorithms in the stock market is growing, as investors and traders increasingly rely on fast and accurate market information to reduce potential risks and make better decisions. These algorithms allow for more efficient analysis of market data, leading to more informed decisions and improved investment outcomes (Obthong, Tantisantiwong, Jeamwatthanachai, & Wills, 2020).

2.1.1 Comparison of Machine Learning and Deep Learning Models in Stock Market Predictions

To have a better grasp in the accuracy of the different models used in algorithmic trading it is essential that different models are compared against each other.

Combination of Computational Efficient Functional Link Artificial Neural Network (CEFLANN) and Traditional Technical Analysis

This hybrid model combines a classification-based model: CEFLANN and the traditional technical analysis to create a stock trading framework Dash and Dash (2016), which the results show a profit of 24.29%.

Deep Long Short-Term Neural Network (LSTM) with Embedded Layer

In one of the models developed by Pang, Zhou, Wang, Lin, and Chang (2020), it shows that by adding an embedded layer to the LSTM it yields to a stock market price prediction accuracy of 57.2%. However, its accuracy dips to 52.4% when the model is applied to individual stocks.

LSTM with Automatic Encoder

As part of the second model developed by Pang et al. (2020), this model shows a slightly inaccurate stock market prediction, by only having a measured accuracy of 56.9%. However, compared to the first model developed by the group this is 0.1% more effective for individual stocks.

Optimal Deep Learning (ODL)

In the study conducted by Agrawal, Khan, and Shukla (2019) they have created a stock price prediction model using an Optimal Deep Learning (ODL) which combine the concepts of Correlation-Tensor and an Optimal LSTM algorithm. Whereas their results show a mean and highest accuracy of the model as 59.24% and 65.64%.

NMC-BERT-LSTM-DQN-X Algorithm

More recently, a team have applied a combination of three models for forecasting the market trends. Namely, (1) Non-stationary Markov Chain (NMC), (2) Bidirectional Encoder Representations from Transformers (BERT), (3) Long Short-Term Memory (LSTM). Wherein their model shows an accuracy of 61.77%. Furthermore, the team also mentioned that the model produces 29.25% annual return on investment, with a maximum losses rating of -8.29% (Liu, Yan, Guo, & Guo, 2022).

2.2 Utilization of Dynamic Mode Decomposition (DMD) on the Financial Markets

Dynamic Mode Decomposition (DMD) as an emerging data-driven technique which allows spatial-temporal pattern recognition from a complex set of data and was first introduced in the field of fluid mechanics by (SCHMID, 2010).

2.2.1 Chronological Utilization of DMD in the Financial Markets

In (2015) Mann and Kutz proved that DMD can be used as data-driven analytics on the financial market data. Wherein, DMD allows a predictive assessment of the market dynamics, which helps in the capitalization of stock market strategies and decisions to be applied.

Utilization of DMD for Determining the Cyclic Behavior in the Stock Market (2016)

By utilizing the reproducible Koopman modes it made it possible to have extracted four cyclic variations (also reproducible modes) in the stock market, which were previously unknown and have persisted since the 1870s' global economic crisis (Hua, Roy, McCauley, & Gunaratne, 2016; Williamson, 2015).

Utilization of DMD as part of an Algorithmic Trading Strategies for the Turkish Stock Market (2015 and 2017)

The study of Mann and Kutz (2015) in the utilization of DMD for financial stock market prediction has become the foundation of the study by Savaș (2017) on the algorithmic trading strategies with Dynamic Mode Decomposition for the Turkish Stock Market. Wherein, based on their results they found out that the timing of DMD analysis was not significantly accurate, as such they have used a simple moving average with genetic algorithm to improve the market timing of DMD, which prevents 80% of the false trade signals.

Furthermore, this also shows that DMD is an effective alpha model that is easy to implement and use for any algorithmic trading strategy, and the addition of technical analysis tools can further improve its capabilities, especially on the predictive temporal side of the data.

Utilization of DMD-based Trading Strategy in the Chinese Stock Market (2016)

In the study by Cui and Long (2016), they have found that DMD was able to capture the dynamic patterns of the Chinese Stock Market, especially in a sideway trending market.

Their study also shows that the predictive ability of DMD can effectively model the behavior of the Chinese Stock Market, even if there are no clear trends that can be observed.

Utilization of Adaptive Elastic DMD to Improve Momentum Strategies (2021)

A study by Uchiyama and Nakagawa (2021), using Adaptive Elastic Dynamic Mode Decomposition (AEDMD) shows that they were able to estimate the market trend, and were able to demonstrate that the approach is better than existing momentum strategy which are only based on simple past trends.

2.3 Synthesis

Fast and accurate market information is an essential tool for stock market participants. In recent years, the development of machine learning models for the financial markets, such as stocks, has proven to be increasingly effective in predicting future stock prices and trends. The use of Dynamic Mode Decomposition (DMD) in the stock market has also been shown to be effective in predicting stock price trends. The simplicity and elegance of the Koopman Decomposition Operator make it an ideal basis for the development of a Stock Market Price Trend Forecasting System.

Hence, these studies are crucial for the development of the alamSYS. As it can provide investors with fast and accurate information about which stocks are likely to go up or down, allowing them to make more informed decisions about buying or selling those stocks.

In addition to the potential benefits for investors and traders, the implementation of machine learning techniques in the stock market can also help improve market efficiency and reduce the risk of market manipulation. By providing a more accurate and comprehensive view of market trends, these techniques can help ensure that prices reflect the true value of stocks and other assets, leading to more stable and fair market conditions.

Chapter 3

Materials and Methods

This chapter discusses the materials and methods used for the design and development of the system: alamSYS. Specifically, the following are discussed in this chapter:

(a) Development Tools and Software Requirements

(b) System Diagrams

(c) Hardware Requirements

(d) Methodology

(e) Gantt Chart

3.1 Development Tools and Software Requirements

The development of the alamSYS utilized the following development tools and software requirements:

3.1.1 Development Tools

(a) Visual Studio (VS) Code – This is a highly functional code editor that served as the project's primary development interface.

(b) MongoDB Compass – This is a graphical user interface for developing and managing various MongoDB databases.

(c) GitHub – This serves as the project's code repository and version control system (via git).

3.1.2 Software Requirements

(a) Python (version 3.9.x) – This served as the primary programming language for the development of the various components of alamSYS, with the following libraries specifically used:

- For the development of the API and Database ODM
  - FastAPI (version 0.85.0) – A library that is primarily used to create modern, fast, and high-performance web framework APIs. (Tiangolo, n.d.). Specifically, utilized in the development of the project because of its (1) ease of utilization; (2) fast implementation; (3) high-performance; (4) built-in robust API documentation; and (5) high scalability.
  - mongoengine (version 0.24.2) – A library designed as an Object-Document Mapper that allows Python to connect to and work with MongoDB. (MongoEngine, n.d.) This was used in the alamSYS to connect the API endpoints to the MongoDB database, and vice versa.
  - json (pre-installed) – This is a Python library for converting a Python dictionary to a JSON object and vice versa. This was used in the development of alamSYS for data parsing and conversion from the API to the MongoDB database via an ODM.
  - datetime (pre-installed) – This python library was used for creating a datatime object, which as the name suggests is an object that

contains the date and time information. This was used in the system to keep track of all the processes that occur in the system using date and time logs.

  - os (pre-installed) – A Python library that allows the user to perform operating system operations such as creating directories and files, accessing operating system information, and so on. This was used to access the operating system's environment variables as well as to assist with other OS-based functions.

- For the preprocessor (main)
  - schedule (version 1.1.0) – This library allows the user to schedule a function to be executed at a specific date and time. This was used in the system to schedule the processes that occurs in the alamSYS.

- For the preprocessor (data collector)
  - requests (version 2.28.1) – This library allows the user to create web requests to an external or internal servers. This was used to connect and collect the current EOD market data from the third-party market historical data provider: EODHD.
    EODHD – A third-party market fundamental and historical data APIs provider (EODHD, n.d.).

- For the preprocessor (data processor):
  Note that some of these libraries are also used in the development of the DMD-LSTM model.
  - numpy (version 1.23.5) - Utilized for handling large data arrays. This is because, compared to Python's List, numpy is better in terms of performance and memory utilization (Geeks for Geeks, 2022).
  - tensorflow (version 2.11.0) - Utilized for the development of the DMD-LSTM model.
  - matplotlib (version 3.7.0) - Utilized for creating graphical diagrams and plots for the results of the data gathering during the developmental stages of the system, specifically during the development of the DMD-LSTM model.

- ❍ pyDMD (version 0.4.0post2301) - This library was used to extract the dynamic modes from the stock market data as an additional training input for the DMD-LSTM model.
- ❍ pandas (version 1.5.3) - This library was used to handle the dataframes during the testing period of the alamSYS.

(b) MongoDB – A non-relational (document-based) database, used to hold the necessary data for the alamSYS. Such as stocks info, which stocks to buy or to sell, and the risk profile of each stocks.

(c) Jupyter Notebook – This was used during the training and testing of the DMD-LSTM model.

(e) Docker – A useful tool to creating containers. Containers contains the source code and all its dependencies in one standard unit of software, which can be run in different machines regardless of its difference from the development machine used (Docker, n.d.). As such this was used to create containers for each of the component of alamSYS, to enable it to run in different deployment machines.

(f) Docker-compose – In order to run multiple containers at once, docker-compose was used. This is further discussed in the Container Diagram section of this chapter.

(g) Dart and Flutter - This was used for the development of the mobile-based test application (alamAPP) to showcase how the alamSYS can be used in an actual application. In addition, the following libraries were used:

- http (version 0.13.5) - This library was used to create HTTP requests to the API endpoints of the alamSYS.
- path_provider (version 2.0.13) - This library was used to allow the alamAPP to access the storage of the device, which then allows the application to save the details collected from alamAPI through the http request library.
- syncfusion_flutter_charts (version 20.4.52) - This library was used to show or visualize the predicted graph based on the price predictions given by the alamSYS.
- lottie (version 2.2.0) - This was used to show the loading animation when the alamAPP is waiting for the response from the alamSYS,

as well as animations when the alamAPP failed to connect to the alamSYS through the alamAPI. Overall, this library makes the application more dynamic, interactive, and more user-friendly.

(h) Git - Used as the version control system for the development of the alamSYS.

(i) GitHub - Used as the repository for the alamSYS.

3.2 System Diagrams

In this chapter, the appropriate system diagrams will be shown and discussed. This shall help in the understanding of the system's features, data flow, and processes. Whereas all the diagrams can be viewed in full resolution, using the GitHub repository, provided in the author's note at the title page.

3.2.1 Top-Level Overview Diagram of the alamSYS and Its Interactions to External Systems

Figure 3.1 shows the top-level overview of the alamSYS and its interactions to any third-party or external applications.

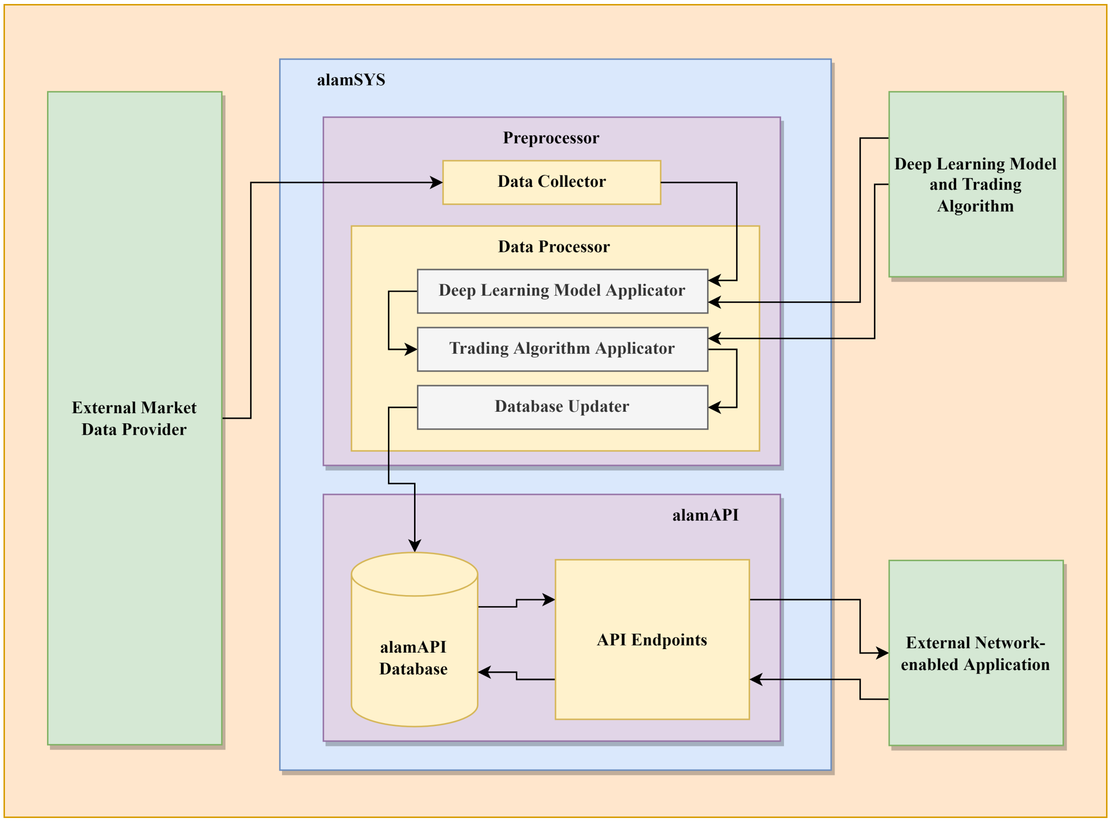

Figure 3.1: Top-Level Overview of the alamSYS and Interactions with External Applications/Systems

As shown from the figure above, the alamSYS is connected to three external entities: (1) External Market Data Provider, which provides the system with the needed historical market data; (2) Machine Learning Model or Trading Algorithm, in the case of this special problem, a machine learning model will be developed and will be utilized by the system, however as previously discussed the system is created to accept any other machine learning model or proprietary trading algorithms that other developers may or want to develop in the future; and (3) External Application, which can be a web-based or mobile-based application, that will utilize and showcase the functionalities provided by the alamSYS, through the API endpoints.

On the middle of the diagram the alamSYS is observed to have three main components, namely, (1) Pre-processor, which is further divided into sub-components:

(a) Data Collector, which collects the data from the external market data provider; (b) Pre-Database Processor, which processes the historical market data collected by applying the developed machine learning model and sending it to the database updater module; (2) Database, which is based on MongoDB, which is a document-based and non-relational database; finally, the database is connected to the (3) API endpoints which processes the request and responses of the system to any external application connected to the API via a network.

3.2.2 Process Flow Diagram

The diagram shown in Figure 3.2 the different processes that the system will undergo once it has been deployed in the server.

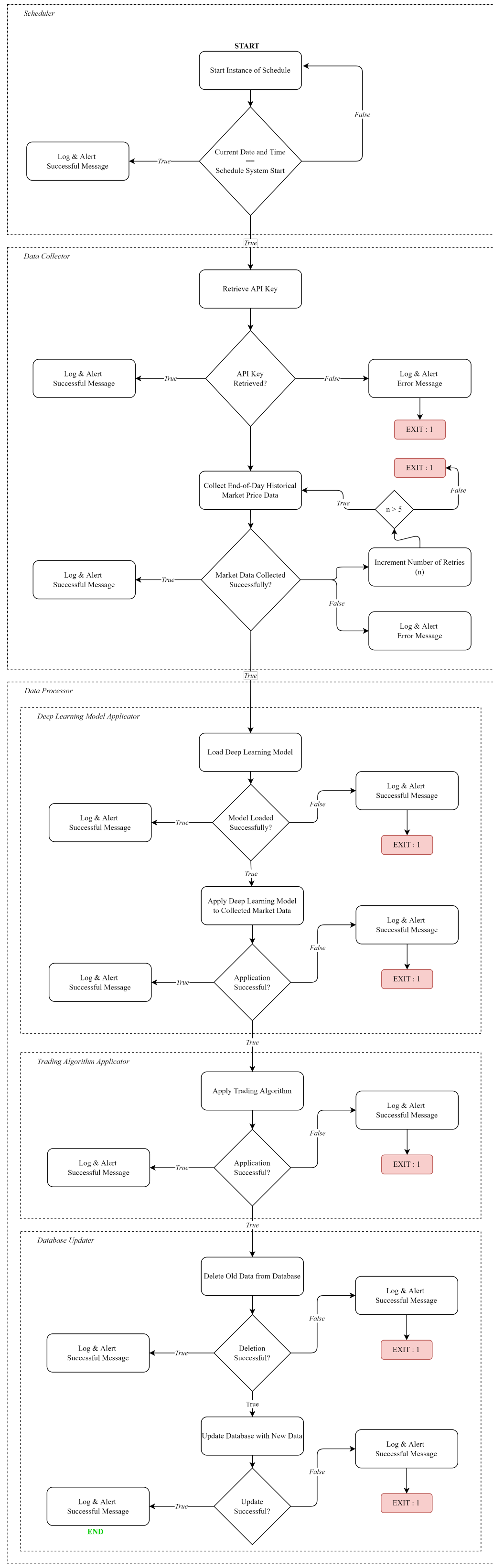

Figure 3.2: Full Overview of the Process Flow Diagram for the alamSYS

To better view and understand the flow of the processes, we can divide the discussions per components in the diagram.

Scheduler

Using Python's Schedule Library, an instance of a scheduled task is initialized upon the startup of the alamSYS. The scheduled task shall execute every Mondays to Fridays at exactly 6 P.M. Where the whole scheduling process is illustrated in Figure 3.3.

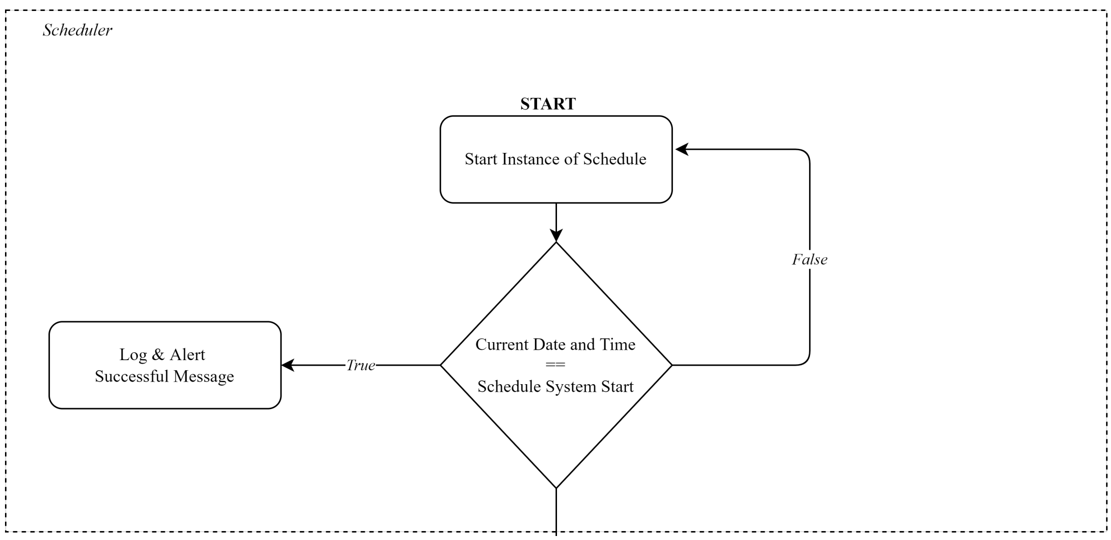

Figure 3.3: Overview of the Process Flow Diagram for the Scheduler

Data Collector

The collection of end-of-day (eod) market data is the first process in the scheduled task.
To collect eod market data, the system first looks for the EODHD API key, which is stored in system variables or provided by the user in the tools directory. If the system is unable to locate an API key, it logs the error and notifies the user before exiting the program.

Once the API key is obtained, the system connects to the EOD market data provider and attempts to collect all market data five times. If it fails to collect data for the fifth time due to errors (i.e. incomplete payments, unstable network, and no established internet connection), the system logs the error, sends an alert to the user, and exits the program.

All successful processes are also logged and sent to the command line interface to notify the user. Figure 3.4 depicts this, as well as the entire data collection process.

The flow of processes discussed above can be observed in Figure 3.4.

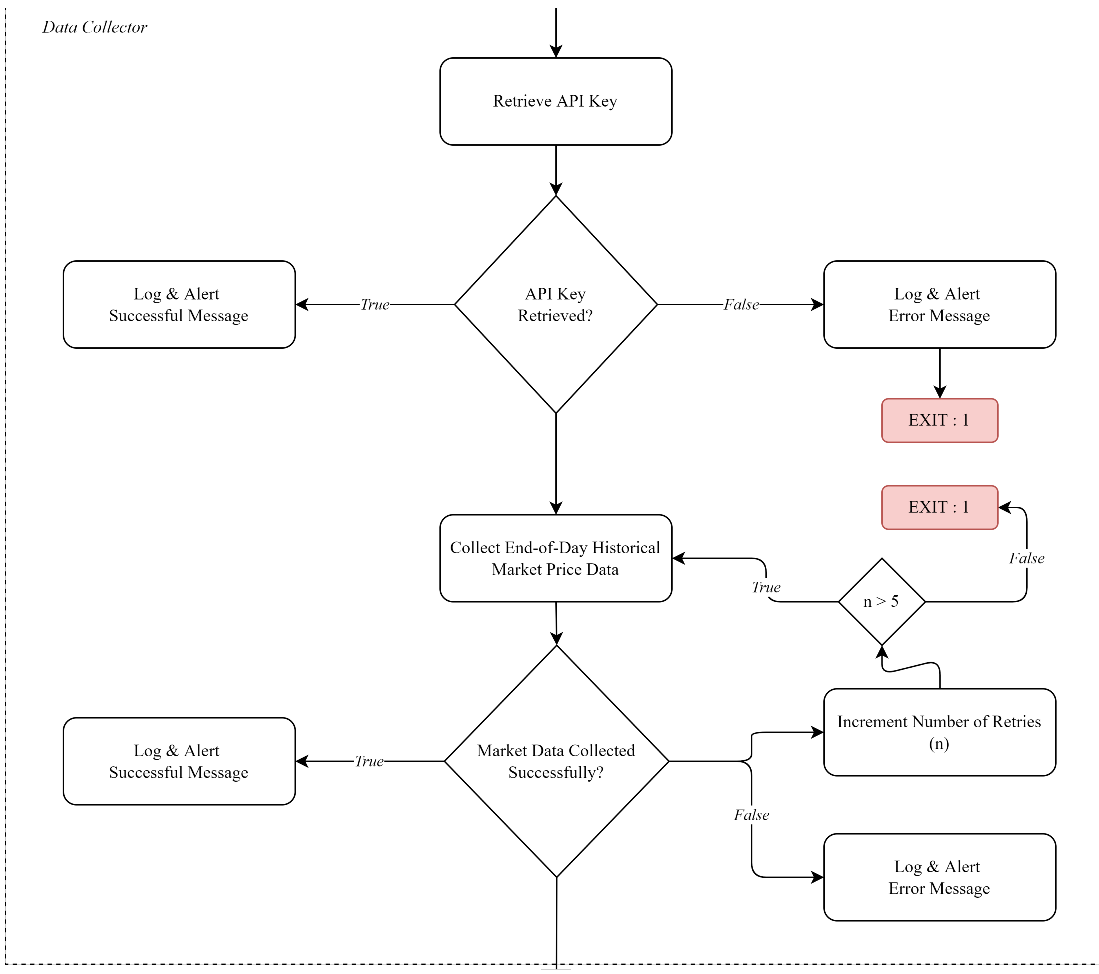

Figure 3.4: Overview of the Process Flow Diagram for the Data Collector

Data Processor

Data Processor is a process that is divided into three subprocesses, as shown in Figure 3.5.

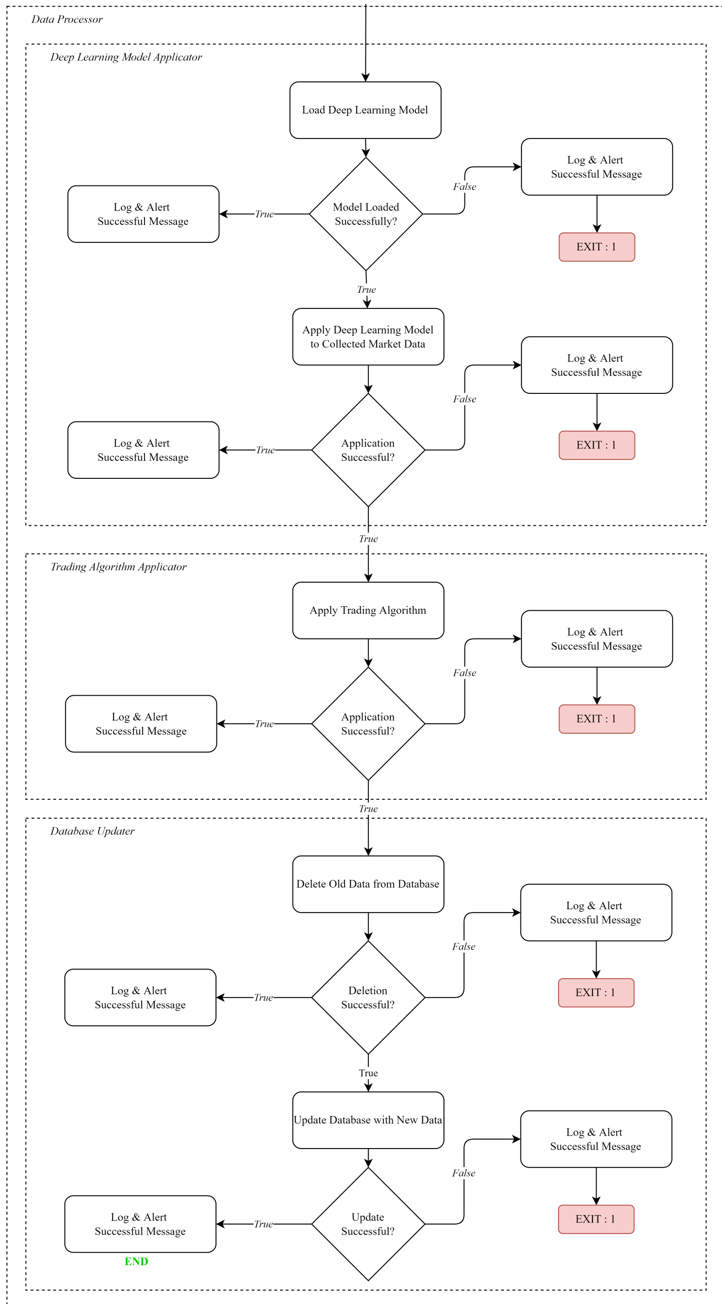

Figure 3.5: Overview of the Process Flow Diagram for Data Processor

Where the three subprocesses are as follows:

(a) Deep Learning Model Applicator - This subprocess applies the deep learning model to the collected eod market data for each stock. Specifically, alamSYS applies the DMD-LSTM model developed as part of this special problem. This is done as illustrated in Figure 3.6.

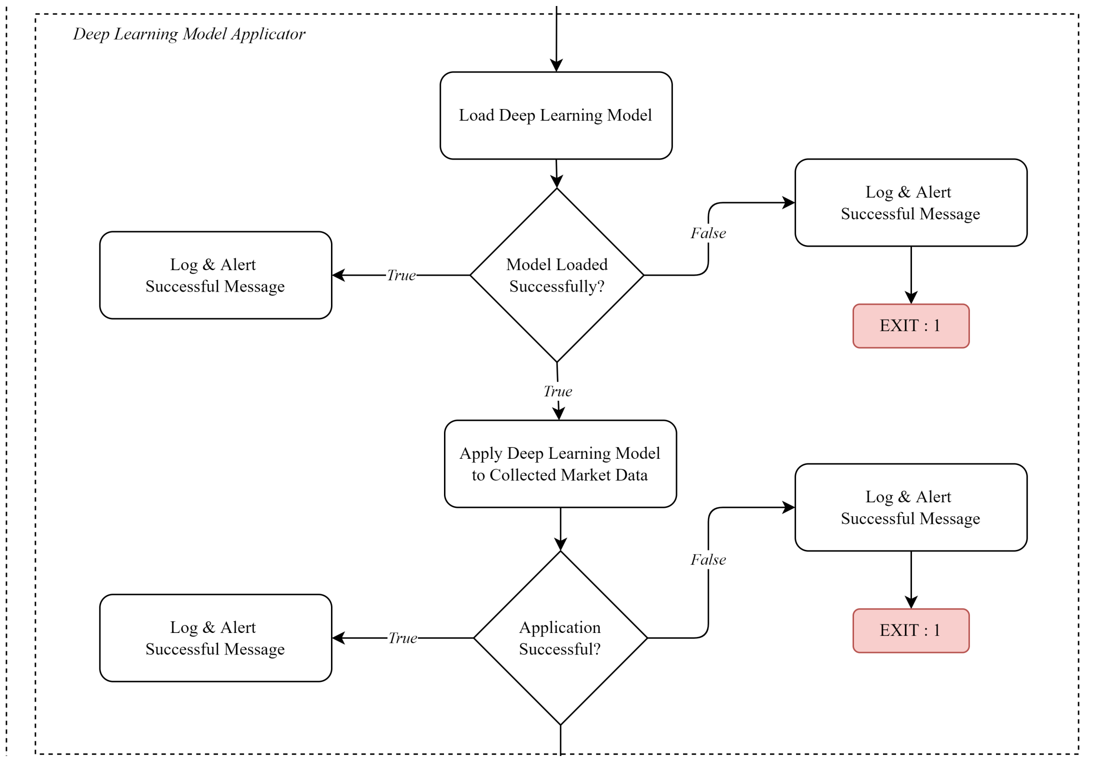

Figure 3.6: Overview of the Process Flow Diagram for the Deep Learning Model Applicator

(b) Trading Algorithm Applicator - This subprocess applies the trading algorithm to the output data from the deep learning model applicator. Specifically, alamSYS applies ALMACD algorithm developed as part of this special problem. This is done as illustrated in Figure 3.7.

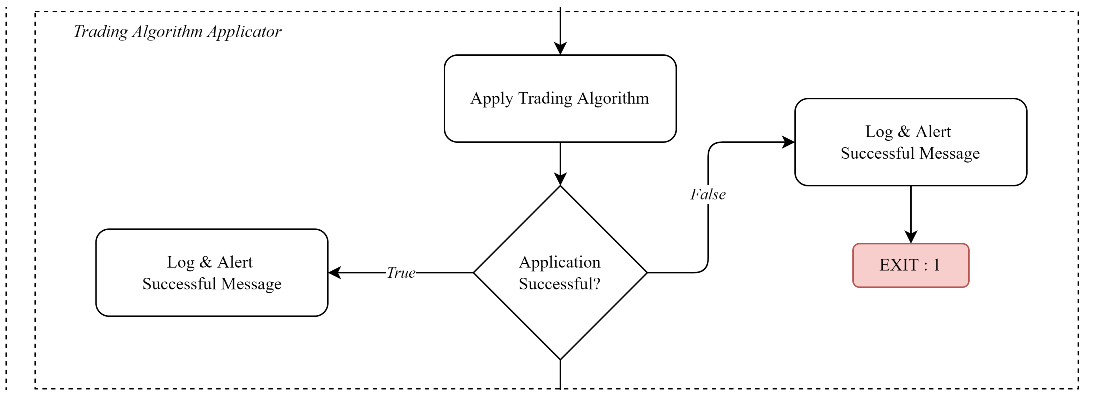

Figure 3.7: Overview of the Process Flow Diagram for the Trading Algorithm Applicator

(c) Database Updater - This subprocess updates the database with the output data from the trading algorithm applicator. This is done as illustrated in Figure 3.8.

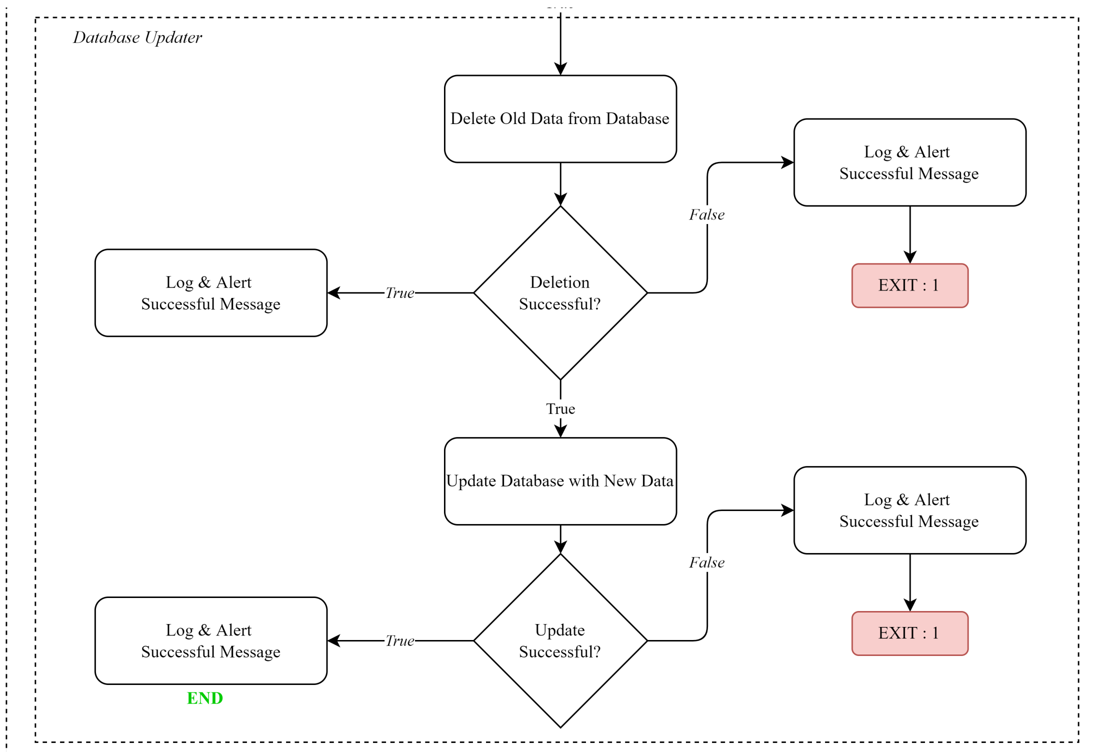

Figure 3.8: Overview of the Process Flow Diagram for the Database Updater

3.2.3 Data-Flow Diagram (DFD)

A data-flow diagram (DFD) helps to understand how processes work and how data flows from one process to the next. This is especially important because it provides an overview of the data's security by demonstrating how it can be accessed. In the case of alamSYS, the only publicly accessible data is the listed stock to buy and sell, as well as other functions as provided in its database and as permitted by the API endpoints.

Furthermore, the DFD paradigm used in the diagrams in this section adheres to the Gane-Sarson DFD symbols, which employ four basic symbols: (1) Entity / External Entity; (2) Data Flow; (3) Process; and (4) Data Store (VisualParadigm, n.d.)

Context Diagram

The overview of the entire process is depicted in a context diagram of the system, labeled process 0, as shown in Figure 3.9.

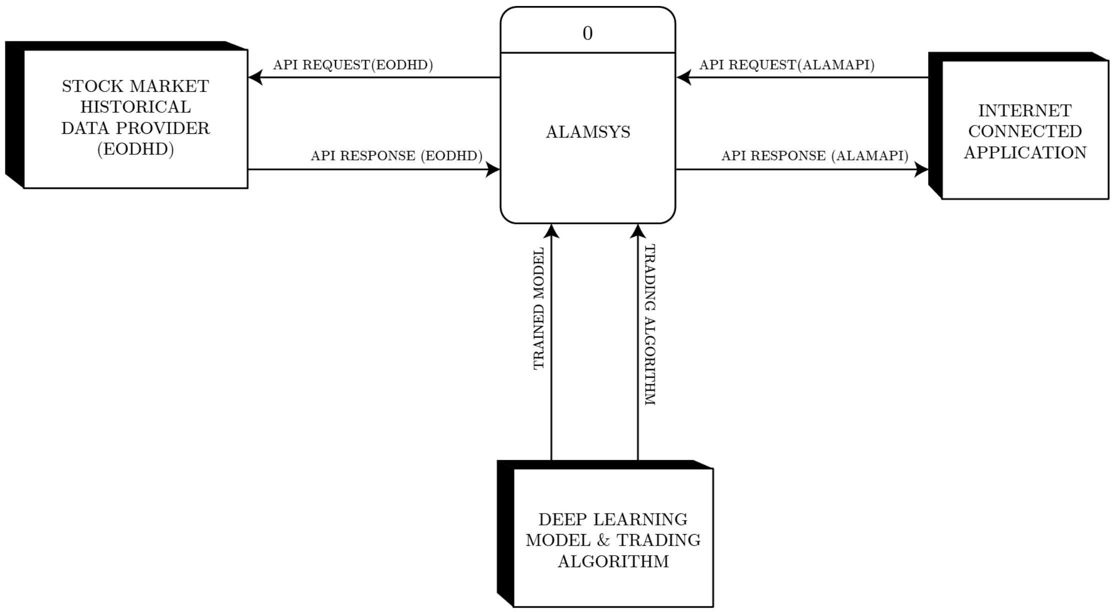

Figure 3.9: Context Diagram of the alamSYS

The diagram above depicts the root process (0), which is the alamSYS itself, and is linked to three external entities:

(a) Stock Market Historical Data Provider - which was provided by EODHD.

(b) Deep Learning Model & Trading Algorithm - these were developed alongside the alamSYS, specifically the DMD-LSTM Model and ALMACD, respectively.

(c) Internet Connected Application - these are any type of applications with internet access. Other applications may include a web-based application, a smart home speaker, and so on.

The data flow lines shown Figure 3.9 include the following:

(a) API Request(EODHD) - This is the request sent to the EODHD API to collect the stock market historical data.

(b) API Response(EODHD) - This is the response received from the EODHD API, which contains the end of day stock market data.

(c) API Request(alamAPI) - This are the requests sent by any internet connected application to the alamSYS via the alamAPI.

(d) API Response(alamAPI) - This are the responses sent by the alamSYS to any internet connected application via the alamAPI.
Whereas, in connection to the alamAPI, the following API points may be requested by the internet connected application:

    - Home - This API endpoint returns a greeting message. Which should notify the user that they have connected to the alamAPI successfully.
    - Stocks to Buy - This API endpoint returns a json data of recommended stocks to buy based on the current market price, the predicted price uptrend, and the entry signal of the trading algorithm in use.
    - Stocks to Sell - This API endpoint returns a json data of recommended stocks to sell based on the current market price, the predicted price downtrend, and the exit signal of the trading algorithm in use.
    - ML Model Info - This API endpoint returns a json data of the Machine Learning Models used in the alamSYS, as well as their associated information.
    - Stock Risks Profile - This API endpoint returns a json data of stocks in the alamSYS as well as their risk values based on value at risk (%), volatility (%), and drawdown (%).

(e) Trained Model - This is the trained model, referring to the DMD-LSTM, that was used to predict the price movement of the stocks in the Philippine Stock Market.

(f) Trading Algorithm - This is the deployed trading algorithm, referring to the ALMACD, that was used to determine the entry and exit signals of the stocks in the Philippine Stock Market.

DFD of Diagram 0

To better understand how each data stream entering and exiting the root process is processed, we must look inside the inner workings of the root process, which is illustrated in the DFD of Diagram 0, as shown in Figure 3.10.

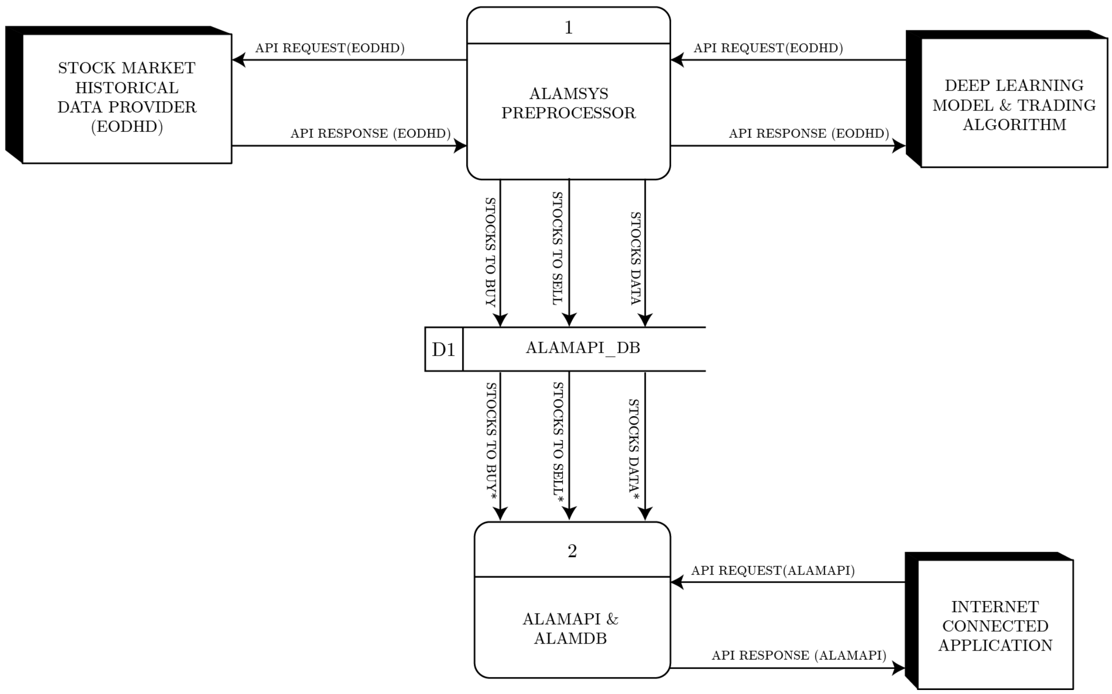

Figure 3.10: DFD of Diagram 0

From the figure above, the root process, has two main processes:

(a) alamSYS Preprocessor - which is the system's stock market data processing unit, which deploys the deep learning model (DMD-LSTM), and the trading algorithm (ALMACD) to predict the price movement of the stocks in the Philippine Stock Market.

(b) alamAPI & alamDB - which is the system's API and database unit, which is responsible for processing the API requests and responses, as well as storing the data of the system.

DFD of Diagram 1

To better understand the internal workings of the Process 1, it is useful to check the DFD of that process, which is illustrated in Figure 3.11

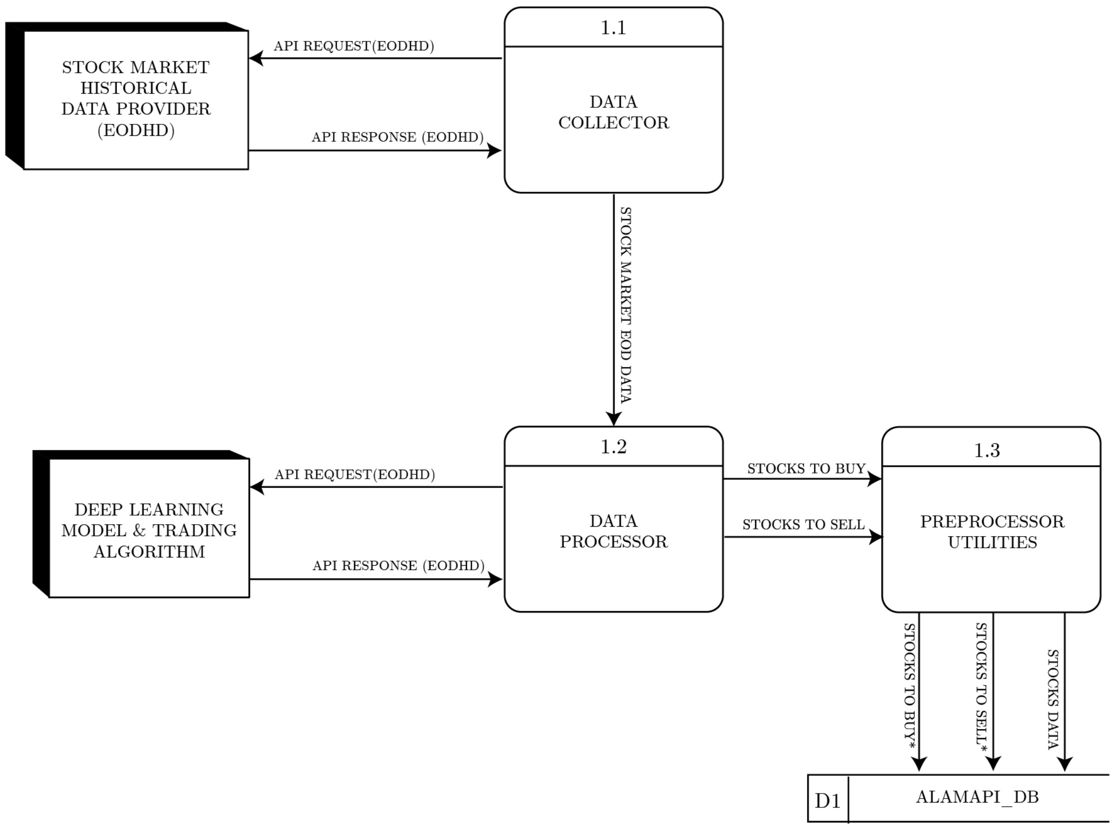

Figure 3.11: DFD of Diagram 1

From the figure shown above, it can be observed that Process 1 is composed of three internal processes, which are as follows:

(a) Data Collector - which is the main process responsible for collecting the historical market data using EODHD End of Day Market Data API.

(b) Data Processor - which is the main process responsible for processing the collected stock market data using the DMD-LSTM Model and the ALMACD Trading Algorithm.

(c) Preprocessor Utilities - these processes contains the utilities used by the data processor, such as the initialization of the database, database related actions, database models, stock symbols, and logs and alerts module.

DFD of Diagram 1.2

This shows the processes inside the process 1.2, which is the data processor.

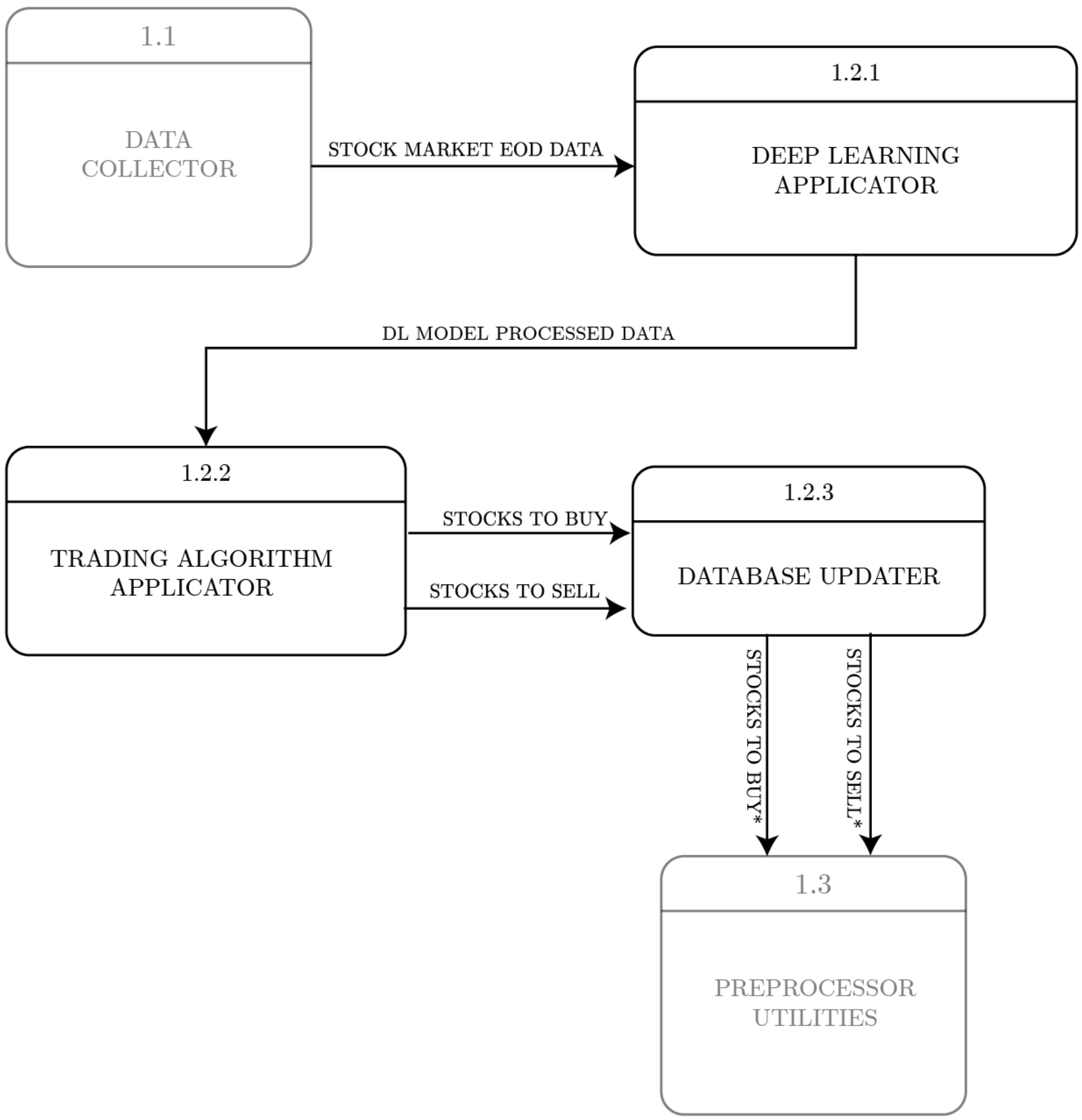

Figure 3.12: DFD of Diagram 1.2

The data processor is further composed of three processes, which are as follows:

(a) Deep Learning Applicator - This subprocess applies the DMD-LSTM model to the collected stock market end-of-day (eod) data.

(b) Trading Algorithm Applicator - The eod data alongside the list of predicted stock prices composes the "DL Model Processed Data", which is sent to this subprocess. This subprocess applies the ALMACD to better determine the entry (buy or hold), and exit (sell) signals for each stocks.

(c) Database Updater - A subprocess responsible for updating the contents of the database, based on the data processed by the Deep Learning Applicator and Trading Algorithm Applicator.

DFD of Diagram 2

Figure 3.13 shows the inner processes of the process 2 (alamAPI & alamDB).

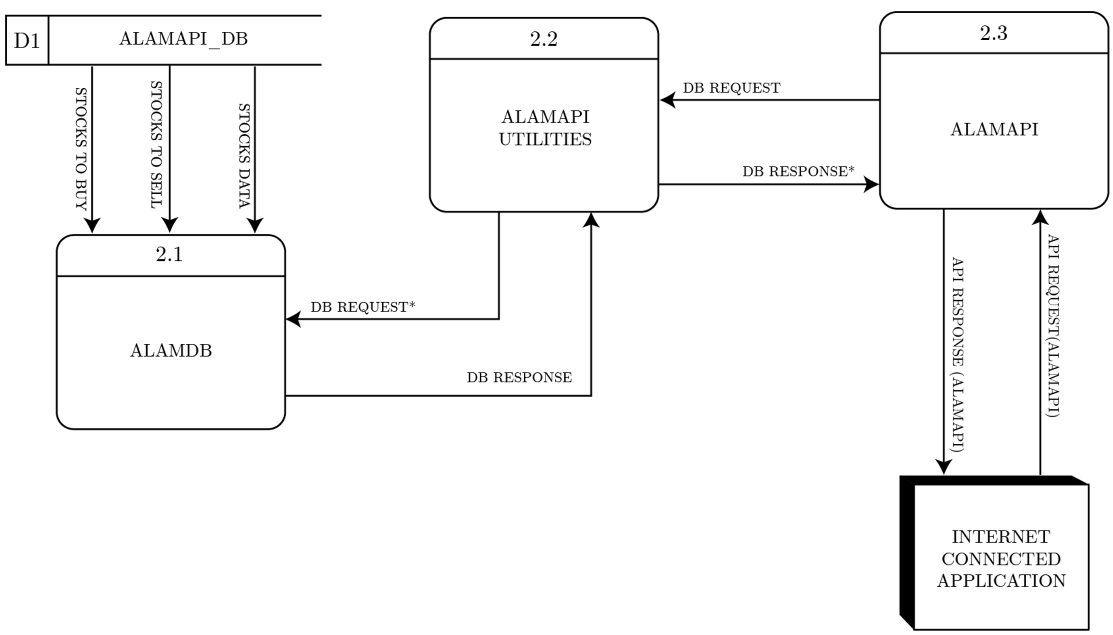

Figure 3.13: DFD 2: Data-Flow Diagram for the alamSYS

The figure above shows three internal processes of the Process 2, namely:

(a) alamDB - This process, processes the database request sent by the alamAPI as requested by the Internet Connected Application through the utilization of alamAPI utilities. It also sends the database response to the alamAPI via the same utilities module.

(b) alamAPI Utilities - This process serves as a mediator of database request and responses between the alamDB and alamAPI.

(c) alamAPI - This process contains all the API endpoints for the alamAPI, which is responsible for processing the requests and API responses from and to the connected users.

3.2.4 Object Document Mapper (ODM) Diagram

Because the system's database is non-relational, an Object Document Mapper (ODM) diagram rather than an Entity Relationship Diagram (ERD) is shown in this section.

The ODM diagram is shown in Figure 3.14:

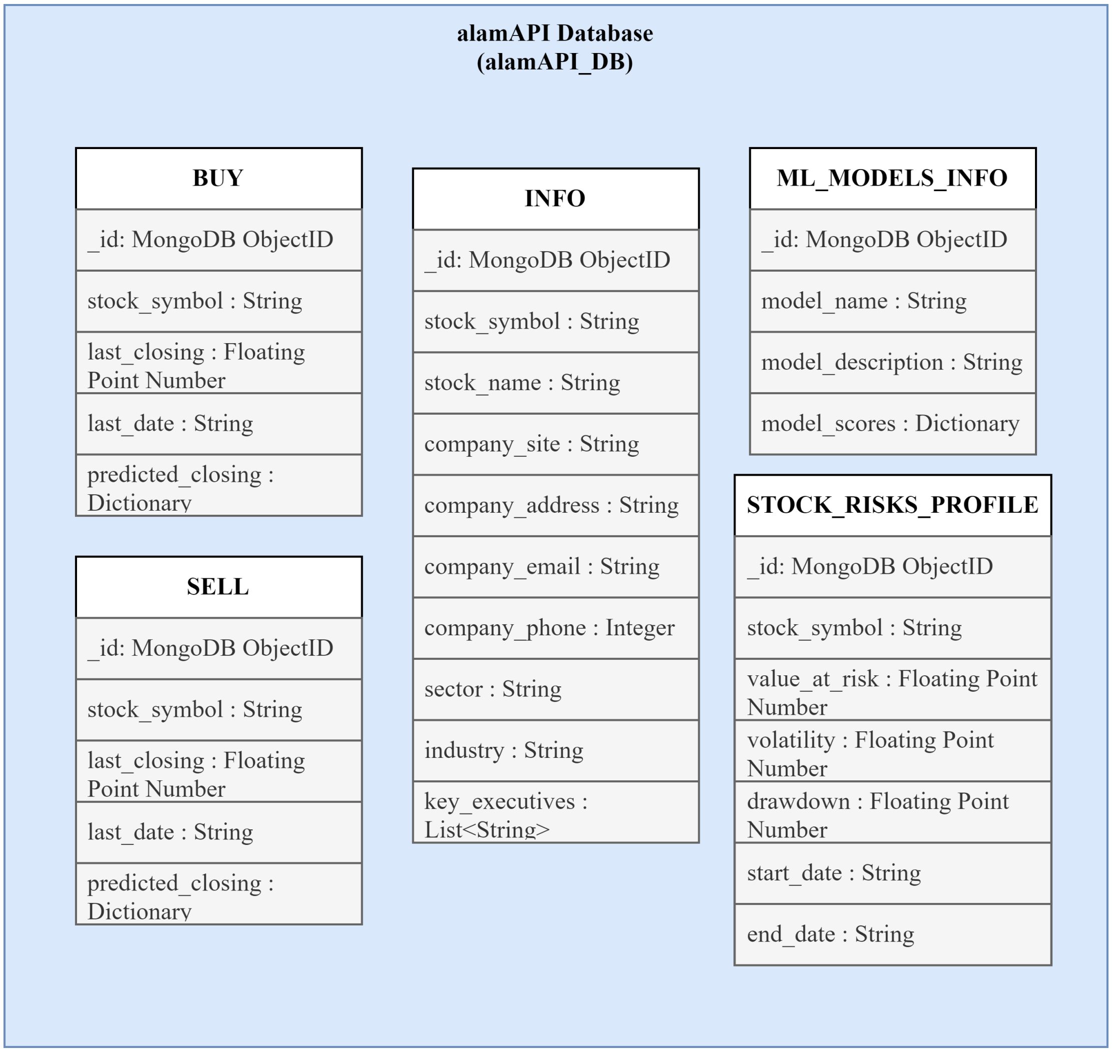

Figure 3.14: Object Document Mapper for the alamSYS Database

As shown from the Figure 3.14, the "alamAPI_DB" is the database name of the system. Where it is composed of five collections namely:

(a) Buy – this collection contains all the stocks that the data processor predicted and classified as a stock to buy. A sample of which is presented in Figure 3.15.

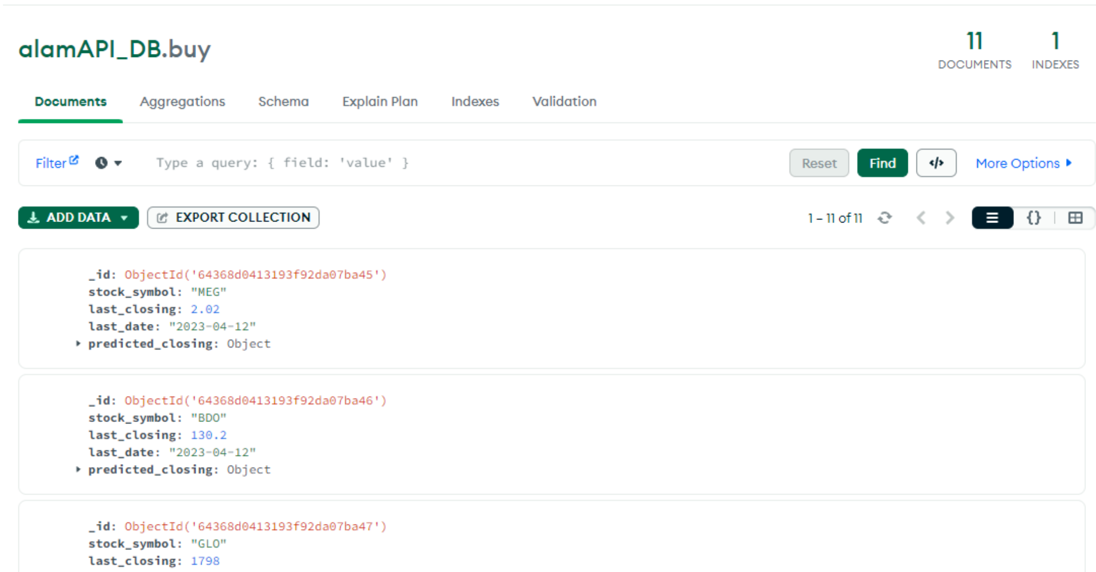

Figure 3.15: Sample Buy Collection from the alamSYS Database

(b) Sell – this collection contains all the stocks that the data processor predicted and classified as a stock to sell. a sample of which is presented in Figure 3.16.

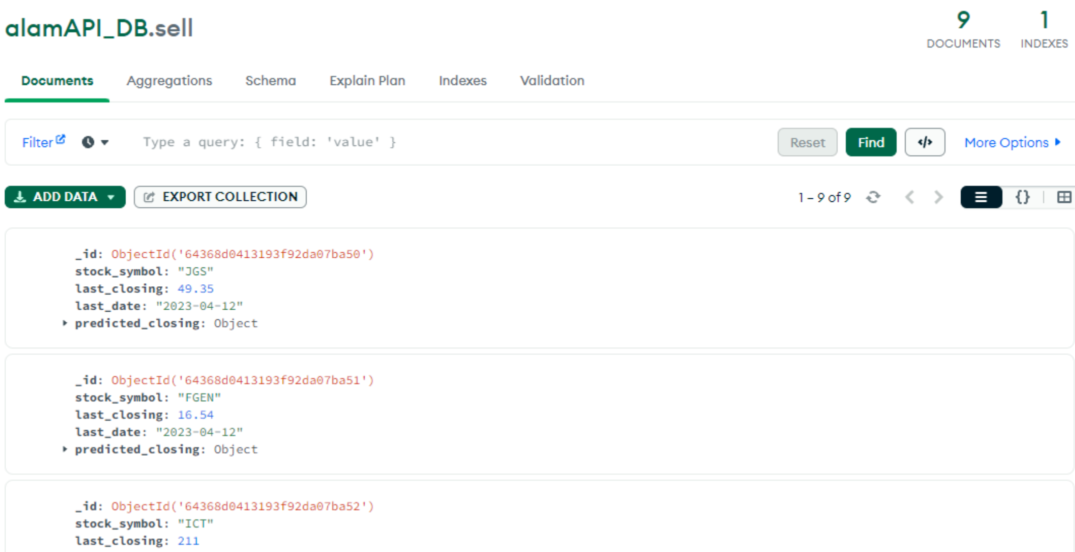

Figure 3.16: Sample Sell Collection from the alamSYS Database

(c) Info – this collection contains the general and relevant information about a stock, or the general company information. Such as the stock symbol, stock name, company site, company address, company email, company phone number, sector, industry, and the company's key executives. Where all of this information are gathered from their official listing accessed in the database of the Philippine Stock Exchange (PSE). A sample of which is presented in Figure 3.17.

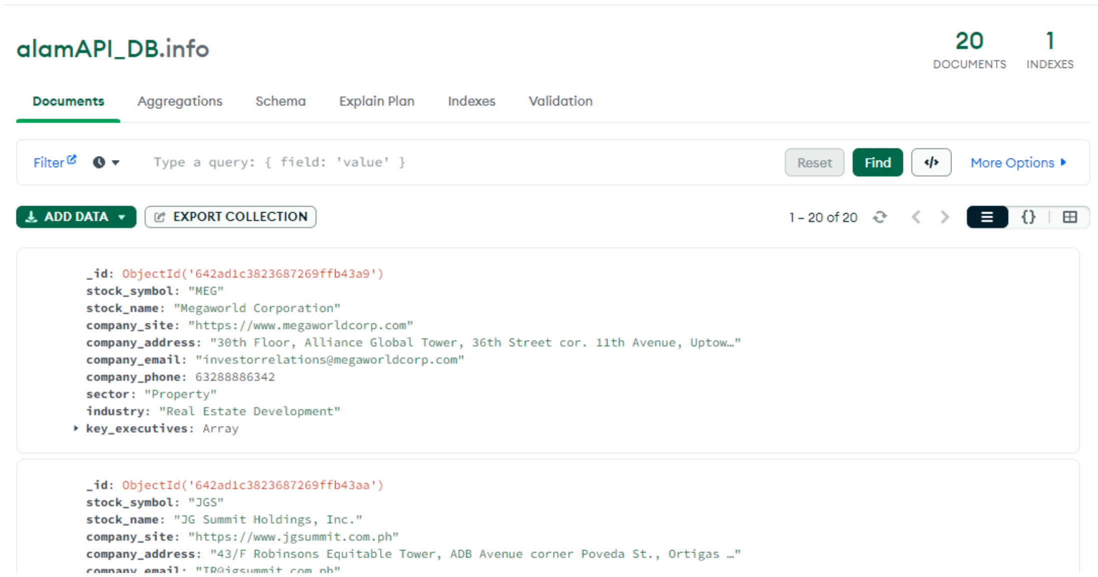

Figure 3.17: Sample Info Collection from the alamSYS Database

(d) Machine Learning (ML) Models Info – this collection contains the details about the Machine Learning Model/s deployed in the system. For the current alamSYS, only one model is deployed, which is the DMD-LSTM model. A sample of which is presented in Figure 3.19.

Figure 3.18: Sample ML Models Info Collection from the alamSYS Database

(e) Stock Risks Profile - this collection contains the details about the risk profiles for each stock. A sample of which is presented in Figure 3.19.

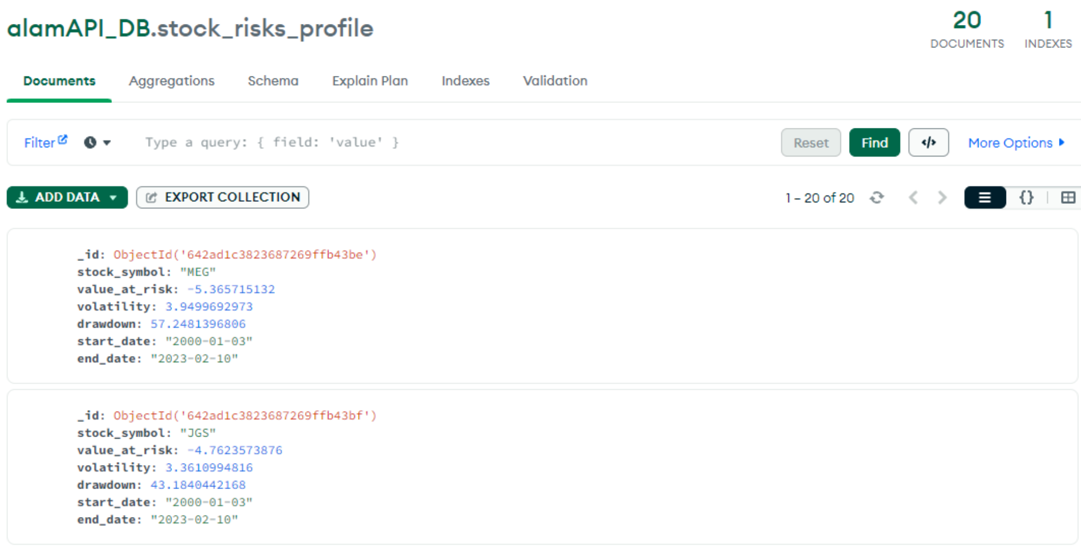

Figure 3.19: Sample Stock Risks Profile Collection from the alamSYS Database

Note that each collection are their own separate entities, hence the database is called non-relational, as the documents are not in any way related to each other.

3.2.5 Machine Learning Model Diagram

In this section, the process on how the machine learning model was developed is shown in Figure 3.20. Wherein, the process overview is based on the Fine-Tuned Support Vector Regression Model for Stock Predictions by Dash and Dash (2016).

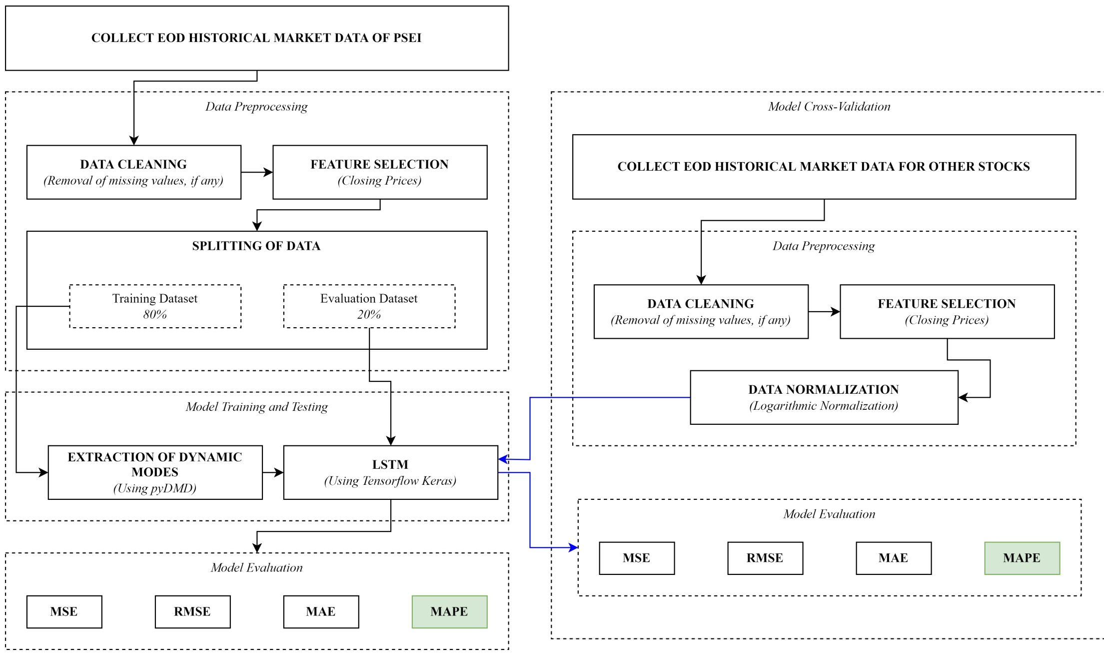

Figure 3.20: Machine Learning Model for the alamSYS

Data Collection

The market data used to develop the DMD-LSTM model was obtained through EODHD's end-of-day market data API. While PSEI market data was used for model training and testing, market data from other stocks was used for cross-validation of the DMD-LSTM model. The following are the specifics of the stocks gathered:

Table 3.1: Collected Market Data Details

Stock	Data Count	Start Date	End Date
AC	6809	June 27, 1994	February 10, 2023
ALI	6789	June 27. 1994	February 10, 2023
AP	3795	July 16, 2007	February 10, 2023
BDO	5041	May 22, 2002	February 10, 2023
BLOOM	3033	October 30, 2000	February 10, 2023
FGEN	4142	February 02, 2006	February 10, 2023
GLO	6707	January 03, 1995	February 10, 2023

Table 3.1 continued from previous page

Stock	Data Count	Start Date	End Date
ICT	6805	January 03, 1995	February 10, 2023
JGS	6525	June 27, 1994	February 10, 2023
LTG	3774	February 06, 1995	February 10, 2023
MEG	6751	January 03, 1995	February 10, 2023
MER	6799	June 27, 1994	February 10, 2023
MPI	3888	December 18, 2006	February 10, 2023
PGOLD	2762	October 06, 2011	February 10, 2023
PSEI	5675	January 03, 2000	February 10, 2023
RLC	5879	June 27, 1994	February 10, 2023
RRHI	2253	November 11, 2013	February 10, 2023
SMC	6799	June 27. 1994	February 10, 2023
TEL	6814	June 27, 1994	February 10, 2023
URC	6135	June 03, 1995	February 10, 2023

Data Preprocessing

Data preprocessing before model training and testing is composed of three main processes which are as follows:

(a) Data Cleaning - This was done to clean any missing values from the data.

(b) Feature Selection - Closing prices was selected as the main feature of the model.

(c) Splitting of Data - Data was split in the ratio of 80:20 for testing and training data, respectively.

Meanwhile for the data preprocessing for cross-validation, the following processes were done:

(a) Data Cleaning - This was done to clean any missing values from the data.

(b) Feature Selection - Closing prices was selected as the main feature of the model.

(c) Data Normalization - Using logarithmic normalization method, the data was normalized. Logarithmic normalization was utilized to help solved the problem with the data having extreme ranges, which affects the evaluation of the cross-validation. In essence it was used to enable data stability, and increase interpretability (Baeldung, 2022; Tuychiev, 2021; Andrew, 2019).

Model Training and Testing

Using pyDMD the dynamic modes was extracted from the training and testing data, these extracted values alongside the actual closing price data were utilized for the training of an LSTM model using Tensorflow Keras Library.

There are a total of eight model variations trained and tested, which are as follows: (1) Baseline LSTM with window size of 5; (2) Baseline LSTM with window size of 10; (3) Baseline LSTM with window size of 15; (4) Baseline LSTM with window size of 20; (5) DMD-LSTM with window size of 5; (6) DMD-LSTM with window size of 10; (7) DMD-LSTM with window size of 15; and (8) DMD-LSTM with window size of 20. Where the best performing model was used for the cross-validation and was deployed to the system.

Model Evaluation

The DMD-LSTM Model was evaluated using the following error metrics:

(a) Mean Squared Error (MSE) - MSE is a well-known metric for assessing regression models. It calculates the average of the squared differences between predicted and true values. MSE is useful because it penalizes large errors more severely than small errors, which is important in some applications. A lower MSE indicates that the model performed better (Glen, n.d.-a).

(b) Root Mean Squared Error (RMSE) - Another popular metric for evaluating regression models is the RMSE, which is the square root of the MSE. It, like MSE, computes the average of the differences between predicted and true values. Because it is in the same unit as the target variable, RMSE is easier to interpret. A lower RMSE indicates that the model is performing better (Glen, n.d.-b).

(c) Mean Absolute Error (MAE) - It calculates the average of the absolute differences between predicted and true values. MAE is advantageous because it is more resistant to outliers than MSE and RMSE. A lower MAE indicates that the model is performing better (Secret Data Scientist, 2023).

(d) Mean Absolute Percentage Error (MAPE) - It computes the average percentage difference between predicted and true values. MAPE is useful because it provides a relative measure of error, which makes comparing model performance across different target variable scales easier. A lower MAPE indicates that the model is performing better (Allwright, 2022). Furthermore, this is the primary error metric used to select the final model deployed to alamSYS.

Model Cross-Validation

The model selected for the cross-validation was the DMD-LSTM model with a window size of 5, due to it being the highest performing model based on having the lowest MAPE scores compared to the other models, this is further discussed on Chapter 4 of this paper.

The cross-validation was conducted using the stock market data from the other stocks aside from the training data from PSEI. Where cross-validation of an LSTM model must be done before deployment to assess the model's generalization performance. LSTM models are well-known for their ability to capture long-term dependencies in sequential data, but their performance varies greatly depending on the dataset and model hyperparameters used. And a correctly performed cross-validation helps to provide a more accurate estimate of the model's performance on unseen data, which is critical for ensuring that the model performs well in real-world scenarios (Mellema, 2020; Scherzinger, Roennau, & Dillmann, 2019).

Model Deployment

After determining that the DMD-LSTM model works well with non-training stock market data, it was deployed to the alamSYS as a '.keras' file.

3.2.6 Docker-Compose Layer Diagram

In this section, the different layers of the docker-compose containers are shown based on the way it will be used in the deployment of the system. Moreover, Figure 3.21 based on the provided diagram in the Docker documentation, regarding containers. Note that in the diagram shown below, the lowest level is the "Server Infrastructure" and the highest level are the three Docker instances.

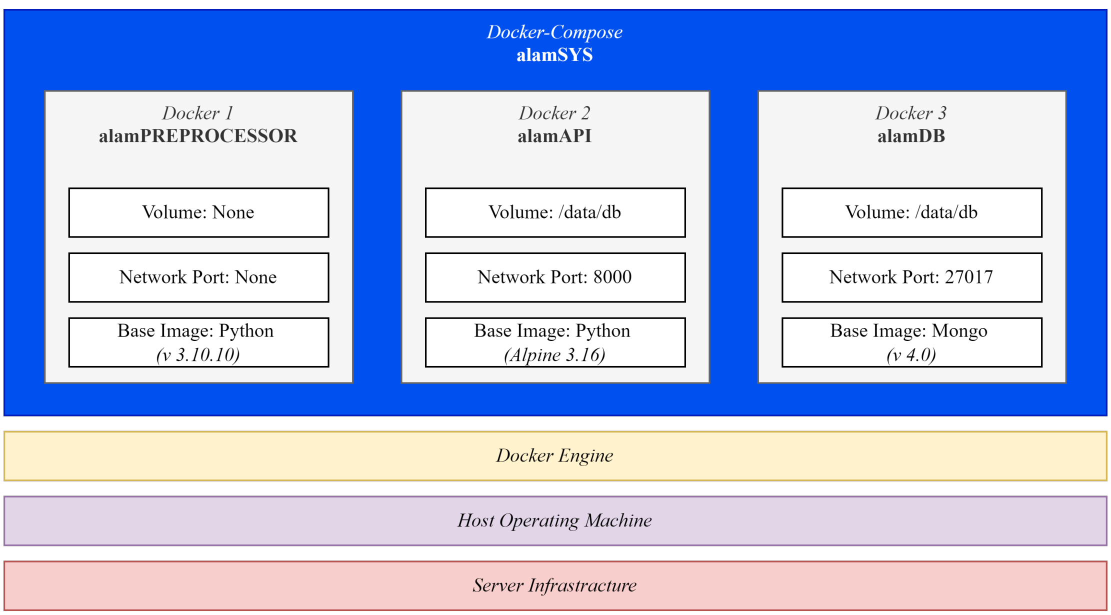

Figure 3.21: Docker-Compose Layer Diagram for the alamSYS

3.3 Hardware Requirements

In this section, the hardware requirements will be discussed.

3.3.1 For the Development of the alamSYS, and Training and Testing of the Machine Learning Model

To develop the alamSYS and the underlying system that it utilizes, the project developer would be needing a laptop with at least the following minimum requirements:

(a) A desktop class 4-core CPU running on 2GHz (minimum).

(b) 16 GB of Random Access Memory (RAM). This is to ensure that multiple instances of programs can run efficiently in the system.

(c) An up-to-date GPU with CUDA cores, this will be used specifically for faster training and testing of the machine learning model. Although if this is not available then a more powerful CPU maybe required.

3.3.2 For the Development of the Test Application and System Testing

(a) A device that can connect to a network such as a smartphone (preferably an Android smartphone: as the test application that will be developed will run on Android devices). Moreover, the device specifications does not matter, as long as it can run a browser or the developed Android test application.

3.4 Methodology

This section of the Chapter 3 will be divided into two sections:

(a) Software Development Process, wherein an Agile development will be discussed; and

(b) Procedures, wherein the general procedures of development will be tackled.

3.4.1 Software Development Process

Due to the expected heavy time constraints of the development of the system, the author of this paper decided to follow an Agile Software Development Process, primarily it will be using Agile Sprints for an efficient time management during the whole software development process. Wherein the following are the list of Sprints and sub-activities that will be followed are shown in the Table below:

Table 3.2: Summary of Sprints and Target Activities

Sprint Number	Target Activities	Allotted Time[1]
1	**Main Activity:** System Planning and Evaluation **Sub-Activities:** • Topic Proposal • Drafting of Chapters 1 to 3 for the Special Problem Proposal • System Architecture and User Requirement Analysis	**12 Weeks** Start: September 15, 2022 End: December 9, 2022

Table 3.2 continued from previous page

Sprint Number	Target Activities	Allotted Time[1]
2	**Main Activity:** System Prototyping **Sub-Activities:** • Build the different component of the alamSYS as indicated in the top-level overview diagram of the system, the following prototype will be developed: [1.] API endpoints [2.] Database [3.] Preprocessor • Testing of the build prototype. This also include creating unit test cases for each component. • Initial Documentations, this will be done inside the GitHub repository.	**12 Weeks** Start: September 30, 2022 End: April 3, 2023

Table 3.2 continued from previous page

Sprint Number	Target Activities	Allotted Time[1]
3	**Main Activity:** Machine Learning Model Training, Testing, and Evaluation **Sub-Activities:** • Collection of Historical Data, outside the Data Collector module of the system. As the full data will be needed for each stock for the training, rather the 200-day only historical data. Whereas the last date on the market data should be January 13, 2023. • Development of the Machine Learning Model. This includes data standardization, data splitting, and data training. • Machine Learning model testing and evaluation. • Revision of Chapter 1-3, in preparation for the final paper submission.	**10 Weeks** Start: January 15, 2023 End: March 30, 2023

Table 3.2 continued from previous page

Sprint Number	Target Activities	Allotted Time[1]
4	**Main Activity:** Integration of Machine Learning Model to the alamSYS and Additional Data Collection **Sub-Activities:** • Testing and Evaluation of alamSYS with the integration of the Machine Learning Model. • System Testing, this will be done to verify the functionality of the whole system, given a test deployment environment. Moreover, this will be done in a span of 4 weeks • Drafting of Chapter 4 and 5	**6 Weeks** Start: March 31, 2023 End: May 12, 2023

Table 3.2 continued from previous page

Sprint Number	Target Activities	Allotted Time[1]
5[2]	**Main Activity:** System Documentation **Sub-Activities:** • Updating and Finalization of Documentations included in the GitHub Repository. • Writing of the results, discussions, conclusions, and recommendations for Chapter 4 – 5 • Special problem paper revisions • Start the development of the test application (for showcasing of the system features)	**6 Weeks** Start: April 14, 2023 End: May 26, 2023

Table 3.2 continued from previous page

Sprint Number	Target Activities	Allotted Time[1]
6[2]	**Main Activity:** Preparation for Final Defense and System Presentation **Sub-Activities:** • Finalization of the mobile-based test application • Revisions and Finalization of the special problem paper. • Creation of presentation slide deck for the presentation of the special problem.	**3 Weeks** Start: May 27, 2023 End: June 17, 2023

1. Start and End Dates are based on the University's Academic Calendar and the Schedule provided by the Special Problem Adviser.
2. Sprints 5 and 6 are no longer part of the actual system development but is still included as a basis for the Gantt chart. Moreover, these activities can still be considered as part of the documentation process.

From Table 3.2, it is shown that there is a total of 39 weeks; from September 15, 2022, to June 17, 2023, however it must be noted that an additional 1 week was added to each sprint's allotted time to compensate for any unforeseen events during each sprint.

It should also be noted that Sprint 1 and Sprint 2 overlaps as the development of the prototype will start at Week 3, this will be possible as there will already be an initial system design to be followed, and any changes made during Sprint 1 can easily be adjusted to the creation of the prototype of the system in Sprint 2. This is also the case for Sprints 4 and 5, since their activities overlaps with each other, such that there are things in Sprint 4 that are unsupervised, hence, to better manage the time it is reasonable to start the activities of Sprint 5 along side the later parts of Sprint 4.

Moreover, the full details about the scheduling will be further discussed in the Gantt Chart of this chapter.

3.4.2 Procedures

In this section, the step-by-step procedures that will be followed in line with the development and testing of the system; alamSYS. Whereas the following are the procedures:

(a) Designing of the System Architecture for alamSYS

(b) Designing of Machine Learning Model

(c) Development of System Prototype

(d) Training, Testing, and Evaluation of the Machine Learning Model

(e) Integration of the Machine Learning Model to the alamSYS

(f) Initial testing for alamSYS, this shall also include any debugging, bug fixing, and code refactoring.

(g) Pre-deployment testing, this testing phase includes the following tests that will be done for a one-month continuous system operation:

   - Functional Testing, by monitoring the functionality of the alamSYS over 30 days and checking the success and error logs at the end of the given timeframe.

– Stress Testing, by creating ten-million artificial requests to the API everyday for 30 days.

(h) Logging and summarization of results from all the prior tests conducted

(i) Analysis and discussion of test data results.

(j) Code Documentation

(k) Maintenance, which will span beyond the time scope of the special problem.

3.5 Gantt Chart

Based on Table refsummary-sprints, the following figures for the Gantt Chart (created using TeamGantt) shows the software development schedule for the development of alamSYS. The Gantt Chart is divided into the different sprint to present the project scheduling. Moreover, a zoomed-out view of the whole Gantt Chart, will also be provided at the end of this section.

3.5.1 Gantt Chart for Sprint 1

Figure 3.22 shows the schedule of activities for Sprint 1. Wherein, it will start on September 15, 2022, and end on December 9, 2022.

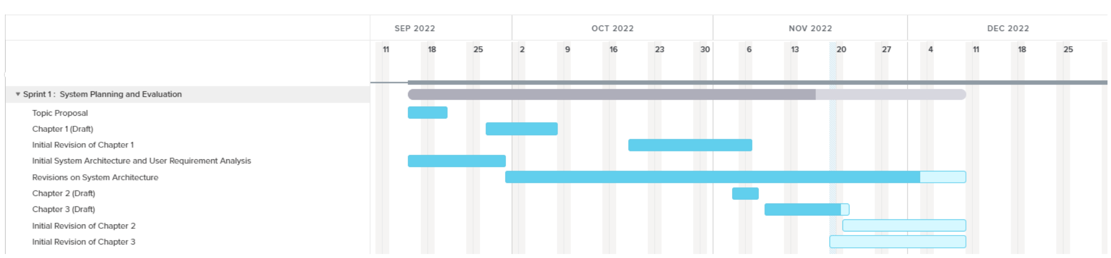

Figure 3.22: Gantt Chart for Sprint 1

3.5.2 Gantt Chart for Sprint 2

Figure 3.23 shows the schedule of activities for Sprint 2. Which will start on September 30, 2022, and end on January 5, 2023.

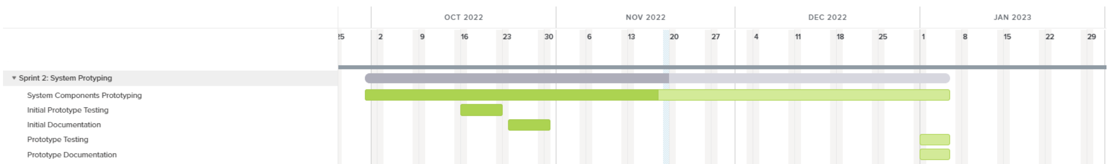

Figure 3.23: Gantt Chart for Sprint 2

3.5.3 Gantt Chart for Sprint 3

Figure 3.24 shows the schedule of activities for Sprint 3. Wherein, it will start on January 15, 2023, and end on March 30, 2023.

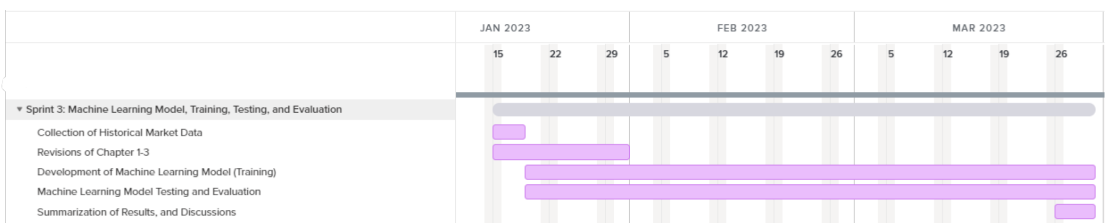

Figure 3.24: Gantt Chart for Sprint 3

3.5.4 Gantt Chart for Sprint 4

Figure 3.25 shows the schedule of activities for Sprint 4. Which will run from March 31, 2023, until May 12, 2023.

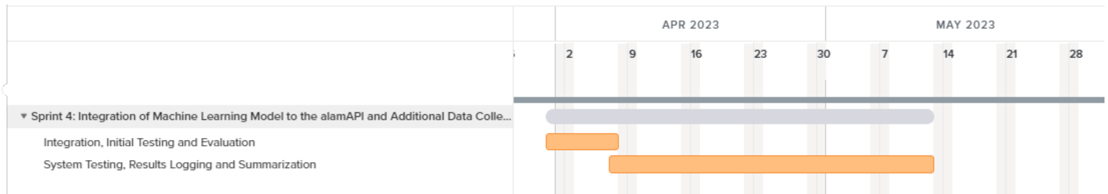

Figure 3.25: Gantt Chart for Sprint 4

3.5.5 Gantt Chart for Sprint 5

Figure 3.26 shows the schedule of activities for Sprint 5. Which will be from April 14, 2023, to May 12, 2023.

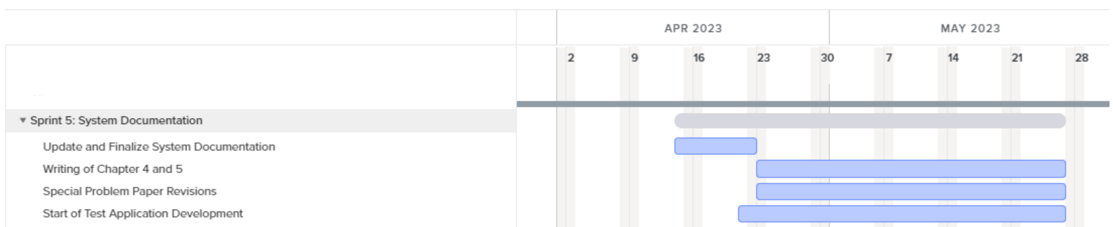

Figure 3.26: Gantt Chart for Sprint 5

3.5.6 Gantt Chart for Sprint 6

Figure 3.27 shows the schedule of activities for the final sprint for the development of alamSYS. Which will be done from May 27, 2023, until June 17, 2023. However, it should be noted that the end day may change, depending on the scheduled final defense.

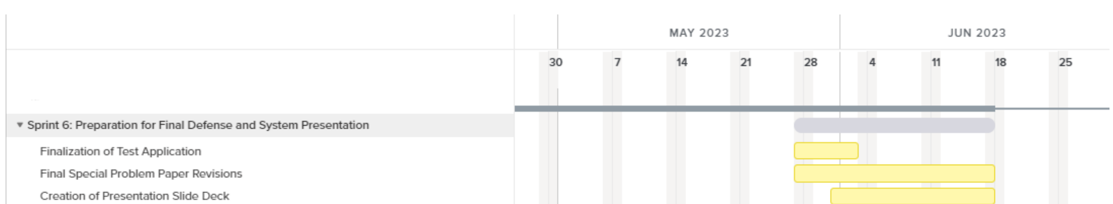

Figure 3.27: Gantt Chart for Sprint 6

3.5.7 Full Gantt Chart

To have an overview of the whole schedule of each Sprints, the full Gantt chart is shown in Figure 3.28.

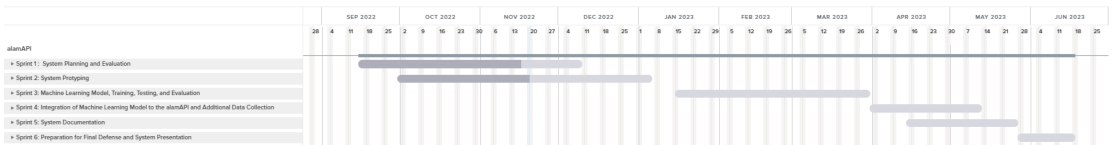

Figure 3.28: Full Gantt Chart

References

Adams, R. (2022, 7). *Is Investing in Stocks Gambling? No, Investing Isn't Zero Sum.* Retrieved October 07, 2022, from https://youngandtheinvested.com/is-investing-in-the-stock-market-gambling/

Agrawal, M., Khan, D. A. U., & Shukla, D. P. K. (2019, 7). Stock price prediction using technical indicators: A predictive model using optimal deep learning. *International Journal of Recent Technology and Engineering (IJRTE)*, *8*, 2297-2305. doi: 10.35940/ijrteB3048.078219

Alegado, S., Lopez, D. B., & Calonzo, A. (2022, 8). *Philippine economic growth slows to 7.4% as inflation bites.* Retrieved October 06, 2022, from https://www.bloomberg.com/news/articles/2022-08-09/philippine-economy-expands-7-4-last-quarter-below-estimate

Allwright, S. (2022). *What is a good mape score?* Retrieved April 15, 2023, from https://stephenallwright.com/good-mape-score/

Andrew. (2019). *You should (usually) log transform your positive data.* Retrieved April 15, 2023, from https://statmodeling.stat.columbia.edu/2019/08/21/you-should-usually-log-transform-your-positive-data/

Bae, K. H., & Kang, J. (2017). *Does the stock market benefit the economy?* Retrieved from https://www.efmaefm.org/0EFMSYMPOSIUM/2017/papers/Does%20the%20Stock%20Market%20Benefit%20the%20Economy%20-%20updated.pdf

Baeldung. (2022). *Normalization inputs for an artificial neural network.* Retrieved April 15, 2023, from https://www.baeldung.com/cs/normalizing-inputs-artificial-neural-network

Balaba, J. M. (2017). *Does the stock market drive the philippine economy?* Retrieved from https://www.dlsu.edu.ph/wp-content/uploads/pdf/conferences/research-congress-proceedings/2017/RVREBM/

RVREBM-I-003.pdf

Bangko Sentral ng Pilipinas. (n.d.). *Stock market.* Retrieved October 06, 2022, from https://www.bsp.gov.ph/Statistics/OtherRealSectorAccounts/stocks.pdf

Budiharto, W. (2021). Data science approach to stock prices forecasting in indonesia during covid-19 using long short-term memory (LSTM). *Journal of Big Data*, *8*(1). Retrieved from https://doi.org/10.1186%2Fs40537-021-00430-0 doi: 10.1186/s40537-021-00430-0

Bureau of the Treasury Bangko Sentral ng Pilipinas. (n.d.). *19 selected domestic interest rates weighted average in percent per annum for periods indicated.* Retrieved October 06, 2022, from www.bsp.gov.ph/Statistics/Financial%20System%20Accounts/tab19_dir.aspx

Business World. (2022, 5). *Psei sinks further as net foreign selling surges.* Retrieved October 06, 2022, from https://www.bworldonline.com/stock-market/2022/05/24/450559/psei-sinks-further-as-net-foreign-selling-surges/

Campbell, G. (2021, 5). *Does the stock market reflect the economy?* Retrieved October 07, 2022, from https://www.economicsobservatory.com/does-the-stock-market-reflect-the-economy

Canto, J., & Romano, K. (2022, 4). *Philippines economic outlook 2022.* Retrieved October 06, 2022, from https://www.mckinsey.com/featured-insights/asia-pacific/philippines-economic-outlook-2022

Chen, J. (2022, 7). *What is the stock market, what does it do, and how does it work?* Retrieved October 06, 2022, from https://www.investopedia.com/terms/s/stockmarket.asp

Commission on Population and Development. (2021). *Popcom: Number of filipinos in 2021 estimated at 110.8 million, sizes of families trending lower at 4 members.* Retrieved October 07, 2022, from https://popcom.gov.ph/popcom-number-of-filipinos-in-2021-estimated-at-110-8-million-sizes-of-families-trending-lower-at-4-members.

Cui, L. X., & Long, W. (2016, 11). Trading strategy based on dynamic mode decomposition: Tested in chinese stock market. *Physica A: Statistical Mechanics and its Applications*, *461*, 498-508. doi: 10.1016/j.physa.2016.06.046

Dash, R., & Dash, P. K. (2016, 3). A hybrid stock trading framework integrating technical analysis with machine learning techniques. *The Journal of Finance* *and Data Science*, *2*, 42-57. doi: 10.1016/j.jfds.2016.03.002

Docker. (n.d.). *Use containers to build, share and run your applications. docker + wasm.* Retrieved November 13, 2022, from https://www.docker.com/resources/what-container/

EdwardJones. (n.d.). *Why invest in stocks?* Retrieved September 19, 2022, from https://www.edwardjones.com/us-en/market-news-insights/guidance-perspective/benefits-investing-stock

EODHD. (n.d.). *Financial apis documentation. fundamental and historical data apis.* Retrieved November 13, 2022, from https://eodhistoricaldata.com/financial-apis/

Fayed, A. (2022, 4). *A list of 14 blue chip companies in the philippines.* Retrieved October 07, 2022, from https://adamfayed.com/a-list-of-14-blue-chip-companies-in-the-philippines/

Geeks for Geeks. (2022). *Python lists vs numpy arrays.* Retrieved January 13, 2023, from https://www.geeksforgeeks.org/python-lists-vs-numpy-arrays/

Glen, S. (n.d.-a). *Mean squared error: Definition and example.* Retrieved April 15, 2023, from https://www.statisticshowto.com/probability-and-statistics/statistics-definitions/mean-squared-error/

Glen, S. (n.d.-b). *Rmse: Root mean square error.* Retrieved April 15, 2023, from https://www.statisticshowto.com/probability-and-statistics/regression-analysis/rmse-root-mean-square-error/

Guo, Y. (2022). *Stock price prediction using machine learning.* Retrieved from https://www.diva-portal.org/smash/get/diva2:1672304/FULLTEXT01.pdf

Hall, M. (2022, 9). *How the stock market affects gdp.* Retrieved October 07, 2022, from https://www.investopedia.com/ask/answers/033015/how-does-stock-market-affect-gross-domestic-product-gdp.asp

Hua, J.-C., Roy, S., McCauley, J. L., & Gunaratne, G. H. (2016, 4). Using dynamic mode decomposition to extract cyclic behavior in the stock market. *Physica A: Statistical Mechanics and its Applications*, *448*, 172-180. doi: 10.1016/j.physa.2015.12.059

Kim, T. (2022, 7). *Alternative data for hedge funds: The competitive edge to investing.* Retrieved November 04, 2022, from https://www.similarweb.com/corp/blog/investor/asset-research/

hedge-funds-use-alternative-data

Kumbure, M. M., Lohrmann, C., Luukka, P., & Porras, J. (2022, 7). Machine learning techniques and data for stock market forecasting: A literature review. *Expert Systems with Applications*, *197*, 116659. doi: 10.1016/j.eswa.2022.116659

Kuttichira, D. P., Gopalakrishnan, E. A., Menon, V. K., & Soman, K. P. (2017, 9). Stock price prediction using dynamic mode decomposition. In (p. 55-60). IEEE. doi: 10.1109/ICACCI.2017.8125816

Liu, C., Yan, J., Guo, F., & Guo, M. (2022, 8). Forecasting the market with machine learning algorithms: An application of nmc-bert-lstm-dqn-x algorithm in quantitative trading. *ACM Transactions on Knowledge Discovery from Data*, *16*, 1-22. doi: 10.1145/3488378

Lu, H., & Tartakovsky, D. M. (2020, 1). Prediction accuracy of dynamic mode decomposition. *SIAM Journal on Scientific Computing*, *42*, A1639-A1662. doi: 10.1137/19M1259948

Mann, J., & Kutz, J. N. (2015, 8). Dynamic mode decomposition for financial trading strategies.

Mellema, G. R. (2020). Improved active sonar tracking in clutter using integrated feature data. *IEEE Journal of Oceanic Engineering*, *45*(1), 304-318. doi: 10.1109/JOE.2018.2870234

MongoEngine. (n.d.). *Mongoengine documentation.* Retrieved November 13, 2022, from https://docs.mongoengine.org/

Obthong, M., Tantisantiwong, N., Jeamwatthanachai, W., & Wills, G. (2020). A survey on machine learning for stock price prediction: Algorithms and techniques. In (p. 63-71). SCITEPRESS - Science and Technology Publications. doi: 10.5220/0009340700630071

Pang, X., Zhou, Y., Wang, P., Lin, W., & Chang, V. (2020, 3). An innovative neural network approach for stock market prediction. *The Journal of Supercomputing*, *76*, 2098-2118. doi: 10.1007/s11227-017-2228-y

Rea, L. (2020, 8). *Predicting future stock market trends with python and machine learning.* Retrieved October 05, 2022, from https://towardsdatascience.com/predicting-future-stock-market-trends-with-python-machine-learning-2bf3f1633b3c

Reuters. (2022, 9). *Philippine economy to grow more slowly than previously thought in 2022, imf says.* Retrieved October 06, 2022, from https://www.reuters.com/markets/asia/philippines-economy-seen-growing-65-2022-50-2023-imf-2022-09-26/

Royal Bank of Canada Direct Investing Inc. (n.d.). Key benefits of investing in stocks. *Royal Bank of Canada Direct Investing Inc.*. Retrieved from www6.royalbank.com/en/di/hubs/investing-academy/chapter/key-benefits-of-investing-in-stocks/jv7atg13/jv7atg1j

Savaș, M. C. (2017). *Algorithmic trading strategies using dynamic mode decomposition: Applied to turkish stock market.* Retrieved from https://etd.lib.metu.edu.tr/upload/12621107/index.pdf

Scherzinger, S., Roennau, A., & Dillmann, R. (2019). Contact skill imitation learning for robot-independent assembly programming. *CoRR*, *abs/1908.06272*. Retrieved from http://arxiv.org/abs/1908.06272

SCHMID, P. J. (2010, 8). Dynamic mode decomposition of numerical and experimental data. *Journal of Fluid Mechanics*, *656*, 5-28. doi: 10.1017/S0022112010001217

Schwab-Pomerantz, C. (2021, 2). *Is investing in the stock market gambling?* Retrieved October 07, 2022, from https://www.schwab.com/learn/story/is-investing-stock-market-gambling

Secret Data Scientist. (2023). *What is mae (mean absolute error)?* Retrieved April 15, 2023, from https://secretdatascientist.com/mae-mean-absolute-error/

Soni, P., Tewari, Y., & Krishnan, D. (2022, 1). Machine learning approaches in stock price prediction: A systematic review. *Journal of Physics: Conference Series*, *2161*, 012065. doi: 10.1088/1742-6596/2161/1/012065

Statista Research Department. (2022). *Number of stock market accounts philippines 2021.* Retrieved October 07, 2022, from www.statista.com/statistics/1194840/philippines-number-of-stock-market-accounts-by-type

Strader, T. J., Rozycki, J. J., Root, T. H., & Huang, Y.-H. (2020). *Machine learning stock market prediction studies: Review and research directions* (Vol. 28). Retrieved from https://scholarworks.lib.csusb.edu/jitim/vol28/iss4/3

Summers, B. D. (2022, 6). *Investing vs. gambling understanding risk-adjusted.* Retrieved October 07, 2022, from https://www.forbes.com/sites/forbesfinancecouncil/2022/06/01/investing-vs-gambling

-understanding-risk-adjusted-performance/?sh=1ca2a49957d9

The Economic Times. (n.d.). *What is 'stock market'.* Retrieved October 05, 2022, from https://economictimes.indiatimes.com/definition/stock-market

The Philippine Stock Exchange, Inc. (n.d.-a). *Company information.* Retrieved October 05, 2022, from https://edge.pse.com.ph/companyInformation/form.do?cmpy_id=478

The Philippine Stock Exchange, Inc. (n.d.-b). *Company list.* Retrieved October 05, 2022, from https://edge.pse.com.ph/companyDirectory/form.do

Tiangolo. (n.d.). *Fastapi documentation.* Retrieved November 13, 2022, from https://fastapi.tiangolo.com/

Trade Brains. (2022, 7). *How does the stock market affect the economy?* Retrieved October 07, 2022, from https://tradebrains.in/how-stock-market-affect-the-economy/

Trading Economics. (n.d.). *Philippines inflation rate.* Retrieved October 07, 2022, from https://tradingeconomics.com/philippines/inflation-cpi

Tuychiev, B. (2021). *How to differentiate between scaling, normalization, and log transformations.* Retrieved April 15, 2023, from https://towardsdatascience.com/how-to-differentiate-between-scaling-normalization-and-log-transformations-69873d365a94

Uchiyama, Y., & Nakagawa, K. (2021, 7). Improving momentum strategies using adaptive elastic dynamic mode decomposition. In (p. 388-393). IEEE. doi: 10.1109/IIAI-AAI53430.2021.00068

U.S. Securities and Exchange Commission. (n.d.). *What are stocks?* Retrieved October 07, 2022, from www.investor.gov/introduction-investing/investing-basics/investment-products/stocks

VisualParadigm. (n.d.). *Gane-sarson data flow diagram tutorial.* Retrieved November 15, 2022, from https://online.visual-paradigm.com/knowledge/software-design/gane-sarson-dfd-tutorial/

Williamson, J. G. (2015). *Handbook of the economics of international migration.* Retrieved December 15, 2022, from https://www.sciencedirect.com/topics/economics-econometrics-and-finance/global-economic-crisis

Appendix A

Code Snippets

```
def main():
    print("This is a test")
```