# VISVESVARAYA TECHNOLOGICAL UNIVERSITY

**Jnana Sangama, Belagavi – 590 018.**

**A PROJECT REPORT**

**on**

## "STOCK AND CRYPTOCURRENCY PREDICTION"

*Submitted in partial fulfillment of the requirement for the award of the 8th semester*

**Bachelor of Engineering**

***in***

**Computer Science and Engineering**

***by***


| | | |
|---|---|---|
| **MUSTAFA SHABBIR BHAVANAGARWALA** | **:** | **1VI18CS064** |
| **NAGARJUN K N** | **:** | **1VI18CS066** |
| **TANZIM ABBAS CHAROLIA** | **:** | **1VI18CS113** |
| **VISHAL M** | **:** | **1VI18CS122** |


***Under the supervision of***


**Mrs. Ashwini M**
**Assistant Professor**

**DEPT. OF COMPUTER SCIENCE AND ENGINEERING (NBA Accredited)**
**VEMANA INSTITUTE OF TECHNOLOGY**
**BENGALURU – 560 034**


**2021 – 22**

Karnataka ReddyJana Sangha®

**VEMANA INSTITUTE OF TECHNOLOGY**

**(Affiliated to Visvesvaraya Technological University, Belagavi)**

Koramangala, Bengaluru-34.

## Department of Computer Science and Engineering

# Certificate

Certified that the project work entitled "**STOCK AND CRYPTOCURRENCY PREDICTION"** carried out jointly by **MUSTAFA SHABBIR BHAVANAGARWALA (1VI18CS064), NAGARJUN K N (1VI18CS066), TANZIM ABBAS CHAROLIA (1VI18CS113)** and **VISHAL M (1VI18CS122),** are bonafide students **of Vemana Institute of Technology** in partial fulfillment for the award of **Bachelor of Engineering** in **Computer Science and Engineering** of the **Visvesvaraya Technological University, Belagavi** during the year 2021-22. It is certified that all corrections/suggestions indicated for internal assessment have been incorporated in the report. The project report has been approved as it satisfies the academic requirements in respect of the project work prescribed for the said degree.

| **Supervisor** | **HOD** | **Principal** |
|---|---|---|
| Mrs. Ashwini M | Dr. M. Ramakrishna | Dr. Vijayasimha Reddy. B. G |

**Submitted for the university examination (viva-voce) held on …….……**

## External Viva

**Internal Examiner** **External Examiner**

1.

2.

# DECLARATION

We the undersigned solemnly declare that the project entitled "**STOCK AND CRYPTOCURRENCY PREDICTION**" is based on our own work carried out during the course of our study under the supervision of **Mrs. Ashwini M**.

We assert the statements made and conclusions drawn are an outcome of our project work. We further certify that,

a. The work has not been submitted to any other Institution for any other degree/diploma/certificate in this university or any other University of India or abroad.

b. We have followed the guidelines provided by the university in writing the report.

c. Whenever we have used resources (data, theoretical analysis, and text) from various sources and given due credit to them in the text of the report and their details are provided in the references.

**Date:**

**Place: Bangalore**

**Project Associates:**


**1VI18CS064 MUSTAFA SHABBIR BHAVANAGARWALA**

**1VI18CS066 NAGARJUN KN**

**1VI18CS113 TANZIM ABBAS CHAROLIA**

**1VI18CS122 VISHAL M**

# ACKNOWLEDGEMENT

First we would like to thank our parents for their kind help, encouragement and moral support.

We thank **Dr. Vijayasimha Reddy. B. G,** Principal, Vemana Institute of Technology, Bengaluru for providing the necessary support.

We would like to place on record our regards to **Dr. M. Ramakrishna,** Professor and Head, Department of Computer Science and Engineering for his continued support.

We would like to thank our project coordinators **Mrs. Roopalakshmi S,** Assistant Professor and **Mrs. A. Rosline Mary,** Assistant Professor, Dept. of CSE for their support and coordination.

We would like to thank our project guide **Mrs. Ashwini M,** Assistant Professor, Dept. of CSE for her continuous support, valuable guidance and supervision towards successful completion of the project work.

We also thank all the Teaching and Non-teaching staff of Computer Science and Engineering Department, who have helped us to complete the project in time.

**Project Associates:**

**1VI18CS064 MUSTAFA SHABBIR BHAVANAGARWALA**
**1VI18CS066 NAGARJUN KN**
**1VI18CS113 TANZIM ABBAS CHAROLIA**
**1VI18CS122 VISHAL M**

# ABSTRACT

In our project, the aim is to predict the future value of the financial stocks of a company and crypto coins respectively with fine accuracy. The recent trend in stock and cryptocurrency prediction technologies, machine learning algorithms make predictions based on the values of current stock and cryptocurrency prices by being trained on their previous values.

Investors are familiar with the saying, “buy low, sell high” but this does not provide enough context to make proper investment decisions. Financial investors of today are facing this problem of trading as they do not properly understand as to which stocks or cryptocurrencies to buy or which to sell in order to get optimum profits. Predicting long term value is relatively easy than predicting on day-to-day basis as the price fluctuate rapidly every hour based on world events. So, our model predicts the price of stocks and cryptocurrencies, which helps the investors to invest in appropriate stocks and crypto coins.

The dataset we use for the proposed project is been taken from yahoo finance for stocks and twelve data for cryptocurrency using web scraping. But this data set is in raw format. The data set is a collection of valuation of stock market information about companies and cryptocurrencies. The initial step is to convert raw data into processed data. Which is done by feature extraction, since the raw data collected have multiple attributes but only close price is needed for the project. We split the data into training and test data. We use the training data and train the model, and then feed test data into the model and predict future price of stock and cryptocurrency. Then using predicted values, we plot the graphs for stock prediction and display predicted price for cryptocurrency. Now the user observes and interpret what’s the scenario and they can decide on the same to invest and get the best benefit out of it.

Keywords: Stock, Cryptocurrency, companies, good accuracy, efficient.

# CONTENTS

# LIST OF FIGURES

# LIST OF TABLES

# LIST OF ABBREVIATIONS AND SYMBOLS

| | |
|---|---|
| ANN | Artificial Neural Network |
| LSTM | Long Short-Term Memory |
| RNN | Recurrent Neural Network |

# CHAPTER 1

# INTRODUCTION

## 1.1 Introduction

### Stocks:

We all have heard the word stock one way or the other. Particularly stock is related with theassociates and companies which are commercialized and are to settling in the world of marketization. The other word used for stock is share which is prominently used in day-to-day life. People even term is as an investment plan and it's something people see as a long-term investment that secures and provides an abundant funds during the retirement age.

Buying a company stock is purchasing a small share of it. People invest on the same to get a long-term benefit which they think is less value for now but has to potential to grow with the time. It's an investment that provides the long time run and deals with long time goals with the fair objectives. The value of share you invest today has to give you a yield of best tomorrow but it's not the same.

Market is unpredictable so are the resources and the factors that are taken to drive it off or on the set. It's never been on the same level and the pattern of the same is still unpredictable till the time. Some closeness and prediction method had been derived and approximates values and the rough figures are generated hoping for the best but all of the resource can't be trusted and are still unpredictable in nature.

Knowing the market situation and researching on the same is the best way to find the reliability for which there are many agents who have taken the same as a profession and are making a fortune out of it. They predict and advise but the advisory cost and the chargeis higher and the stock evaluation is never less the same.

Stock in other term is defined as the fair share or the ownership representation explaining the security measures and the agreement between two parties which are an individual and the company. Stock is there from the start and due to its tendency of uncertainty it has been a word of fancy. People researching on the same and implementing on the daily basis had made a fortune out of it. There are various agents available in market for making you understand and invest on the same and the charges of the same are hectic and insanely expensive.

The main resources for the company is the fund to carry out the daily work and create a profit out of it. In time of need for an higher budget estimation and to overgrow from the resources they need the finance and undergoing a finance loan for approval, passing and having one is hectic and the banks are vultures for which the interest rate is higher than the other form of investment hence limiting the margin of the product.

Stock is another way for company to collect revenue and boost up the production for the upper yield and to gain the most out of the business plan for the bigger pictures. This is found to be an effective way to invest and grow in the commercial field and a better alternative to tackle the financial crisis during the requirement.

For an investor it's a risk phenomenon where they invest their saving and hope it brings back the return in higher yield. If the evaluation of the same increases, then the stock evaluation and its price increases causing the financial gain to both the parties. In Indian Society it is even consider as a side point business and people believe it as a hand of luck.

When an individual purchases a company stock then they're referred as a shareholder and they will get a share out of the same as they have invested in their profit or the gain. A investor can sell and buy the stock as per their needs. They can share their stock to their respective or the other individuals where as there are many stock brokers available out in the firm playing with the same.

## Cryptocurrency:

Time series prediction is not a new phenomenon. Prediction of most financial markets such as the stock market has been researched at large scale. Cryptocurrency presents an interesting parallel to this as it is a time series prediction problem in a market still in its beginning stage.

As a result, there is high volatility in the market and this provides an opportunity in terms of prediction. Due to the open nature of Cryptocurrency, it also poses another difficulty asopposed to traditional financial markets. It operates on a decentralized, peer-to-peer and trustless system in which all transactions are posted to an open ledger called the Blockchain. This type of transparency is not seen in other financial markets. Traditional time series prediction methods such as Holt- Winters exponential smoothing models rely on linear assumptions and require data that can be broken down into trend, seasonal and noise to be effective. This type of methodology is more suitable for a task such as predictingsales where seasonal effects are present.

Due to the lack of seasonality in the Cryptocurrency market and its high volatility, these methods are not very effective for this task. Given the complexity of the task, deep learning makes for an interesting technological solution based on its performance in similar areas.

## 1.2 Scope

Here we are predicting future stock values of various companies around the world as well as various cryptocurrency prices. Here we are fetching data through web scraping and using deep learning LSTM model for predictions. We are plotting the predictions as well.

## 1.3 Objectives

The main objective is to predict the future value of the financial stocks of various companies around the world and various crypto coins respectively with fine accuracy using machine learning model such as RNN LSTM model.

To implement the stock and cryptocurrency predictions and displaying on the website, so that investors can view the predictions and invest in best stocks and cryptocurrencies as per their requirements.

## 1.4 Organization of the project work

### 1.4.1 Plan of action for the project

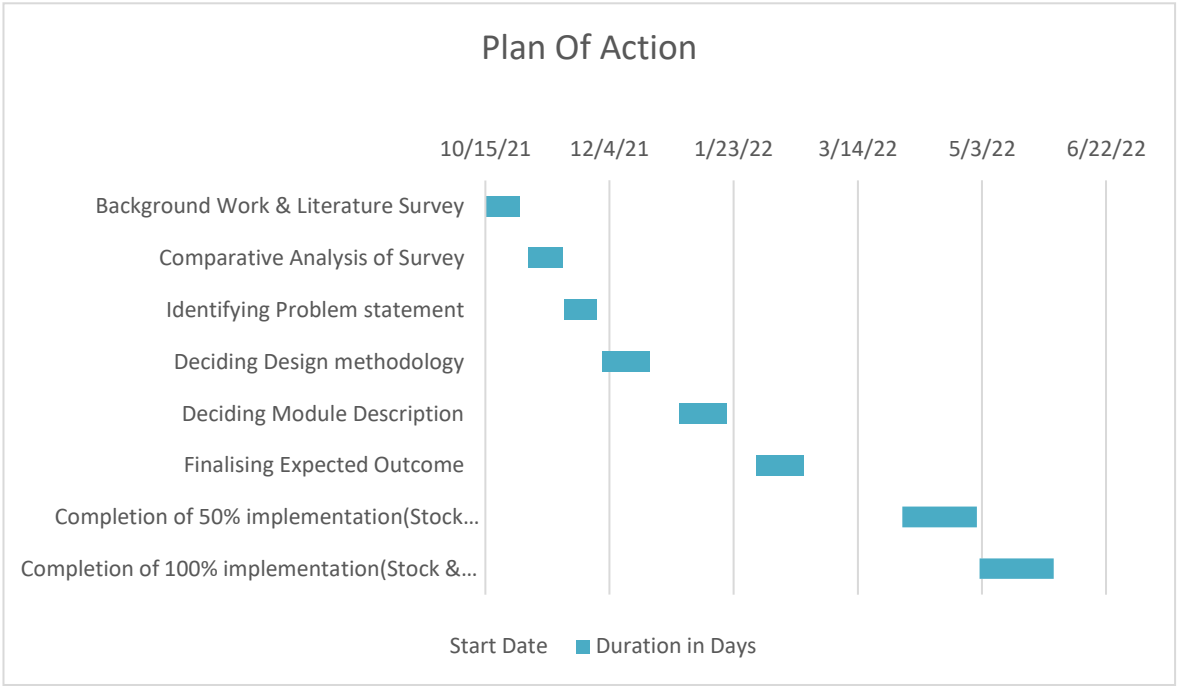



**Fig 1.4.1: Timeline of the project work**

Above Table 1.2.1 depicts the completion of both the current and proposed plan of action for the project. October 15th, the domain of the project, the project title were selected. Once these were approved by the project guide and the project coordinator, background work and literature surveys regarding the project were studied. Comparative analysis of the survey papers was carried out and from this it was concluded that RNN LSTM model was the best fit for the project. Then the problem statement and obstacles, design methodology was done from November to mid-January of 2022. After the design methodology, the project modules and expected outcome were identified. Later on April, stock prediction was done and on June cryptocurrency prediction and the website were completed.

### 1.4.2 Proposed plan for completion

- 4th week of April – Stock Prediction was done.
- 1st week of June – Cryptocurrency Prediction was completed.
- 3rd week of June – The model was implemented on a website.

### 1.4.3 Outline of the chapters

- **Chapter 2:** Explains about the literature survey which gives the brief account of information we gain from each paper and journals referred.
- **Chapter 3**: Describes about the existing system, their drawbacks and the proposed system with feasibility study.
- **Chapter 4**: Software and Hardware requirements.
- **Chapter 5**: Explains the details about Project description, problem definition along with description about the modules which provides the block diagram and description.
- **Chapter 6**: Testing the system with the possible test cases and their results.
- **Chapter 7:** Explains about the results and snapshots of the project.
- **Chapter 8:** Conclusion and future enhancement.

# CHAPTER 2

# LITERATURE SURVEY

## • 2.1 Survey 1: Stocks

- **Title:** Stock Market Prediction Using Linear Regression and SVM.
- **Author:** Bhawna Panwar, Gaurav Dhuriya, Prashant Johri, Sudeept Singh Yadav, Nitin Gaur.
- **Methodology:** Linear Regression and SVM.
- **Description:** Several machine learning algorithms have shown that these stock prices can be predicted and these algorithms can be implemented using the approach of supervised learning. In Supervised Learning, we have test data using this we train the models. In this paper, the first task is to use web scrapping to collect datasets from stock data. Then we plot the data on the graph, from the graph we can analyze the stock prices going high or low. After this, we will predict stock prices using SVM and Linear Regression, that Linear Regression for stock market analysis is better than the SVM for the same.

  From this paper, we have analyzed two of such algorithms on their comparisons analyzing the more accurate one for the stock market prediction. The machine learning algorithms are Linear Regression and SVM. The dataset is obtained by web scraping the data from the yahoo finance site. The data is then formulated in the form of a table to the user for plotting the graphs of companies like Apple, Amazon, Microsoft, Google, etc. The graphs can be analyzed for visually analyzing the stock going up and down.

  On observing the accuracy of both the algorithms on same the stock data it was observed linear regression predicted more accuracy as compared to the SVM model. Thus, it is proved that Linear Regression for stock data or stock market analysis is better than the SVM for the same
- **Advantage:** Forecasting prices of Amazon Stock with high accuracy of 94%.
- **Limitation:** Predicting only for Amazon stock with limited dataset.

## • 2.2 Survey 2: Cryptocurrency

- **Title:** Crypto-Currency price prediction using Decision Tree and Regression techniques.
- **Author:** Karunya Rathan, Somarouthu Venkat Sai, Tubati Sai Manikanta.
- **Methodology:** Decision Tree and Regression techniques.
- **Description:** Cryptocurrency such as Bitcoin is more popular these days among investors. In the proposed work, it is studied to forecast the Bitcoin price precisely considering different parameters that influence the Bitcoin price. This study first handles, it is identified the price trend on day by day changes in the Bitcoin price while it gives knowledge about Bitcoin price trends. The dataset till current date is taken with open, high, low and close price details of Bitcoin value.

  Exploiting the dataset machine learning module is introduced for prediction of price values. The aim of this work is to derive the accuracy of Bitcoin prediction using different machine learning algorithm and compare their accuracy. Experiment results are compared for decision tree and regression model.

  The Bitcoin's simply and similar like a stock but in different way. There are various algorithm using machine learning are utilized on price prediction on stock value. The features influencing Bitcoin are unique. For investors it is mandatory to predict Bitcoin prices. Bitcoin price do not affect by business announcements or government announcements and it is not at all like securities exchange. Thus, we exploit machine learning techniques to foresee the cost of Bitcoin.

  The objective of the proposed study is for price prediction of bitcoin by feature selection of different machine learning techniques.

- **Advantage:** Forecasting prices of popular cryptocurrency (Bitcoin) with high accuracy of 95%.
- **Limitation:** Predicting only Bitcoin prices.

## • 2.3 Survey 3: Cryptocurrency

- **Title:** Bitcoin Price Predictive Modeling Using Expert Correction.
- **Author:** Bohdan M. Pavlyshenko.
- **Methodology:** Bayesian Regression model.
- **Description:** The linear model for Bitcoin price which includes regression features based on Bitcoin currency statistics, mining processes, Google search trends and Wikipedia pages visits. The pattern of deviation of regression model prediction from real prices is simpler comparing to price time series. It is assumed that this pattern can be predicted by an experienced expert. In such a way, using the combination of the regression model and expert correction, one can receive better results than with either regression model or expert opinion only.

  It is shown that Bayesian approach makes it possible to utilize the probabilistic approach using distributions with fat tails and take into account the outliers in Bitcoin price time series.

  One of the main goals in the Bitcoin analytics is price forecasting. There are many factors which influence the price dynamics. The most important factors are: interaction between supply and demand, attractiveness for investors, financial and macroeconomic indicators, technical indicators such as difficulty, the number of blocks created recently, etc. A very important impact on the cryptocurrency price has trends in social networks and search engines. Using these factors, one can create a regression model with good fitting of bitcoin price on the historical data.

  In the proposed approach, the expert has to define time pivot points which describe the deviation of regression model based on the historical data comparing to real price time series. With Bayesian inference, one can utilize the probabilistic approach using distributions with fat tails and take into account outliers in Bitcoin price time series.

- **Advantage:** Pattern of deviation of real price is simpler compared to price time series.
- **Limitation:** Accuracy is unknown and predicting only for bitcoin.

## • 2.4 Survey 4: Stocks

- **Title:** Stock Market Analysis Using Linear Regression and Decision Tree Regression.
- **Author:** Rezaul Karim, Md Khorshed Alam, Md Rezaul Hossain.
- **Methodology:** Linear Regression and Decision TreeRegression.
- **Description:** The Stock market or Share market is a more perplexing and sophisticated way to do business. Every business owner wants to reduce the risk and make an immense profit using an effective way. The bank sector, brokerage corporations, small ownerships, all depends on this very body to earn profit and reduce risks. However, using the machine learning algorithm of this paper to predict the future stock price and shuffle by using subsist algorithms and open source libraries to assist in inventing this unsure format of business to a bit more predictable. The proposed system of this paper works in two methods – Linear Regression and Decision Tree Regression. Two models like Linear Regression and Decision Tree Regression are applied for different sizes of a dataset for revealing the stock price forecast prediction accuracy. Moreover, the authors of this paper have revealed some development that could be the club to acquire better validity in theseapproaches. This is the prediction about stock market and we can solve it by using classification algorithm. So, in this research we use linear regression and Decision tree regression as a classification for prediction stock market.

  The objective of this paper is to get a better decision using two supervised regression machine learning algorithms and the use of statistic formula gives us better accuracy of stock price predict.

  In the proposed work, Bitcoin dataset is considered from 2011 to till date price and applied machine learning models such as Decision Tree and Linear regression models. Also the price forecast for five days is done using Decision Tree and Linear regression models. The proposed learning method suggest the best algorithm to choose and adopt for crypto currency prediction problem. The experimental study results show that linear regression outperforms the other by high accuracy on price prediction
- **Advantage:** High accuracy 99%.
- **Limitation:** Considering dataset of only 1 year.

## • 2.5 Survey 5: Stocks

- **Title:** Stock Price prediction using LSTM and SVR.
- **Author:** Gourav Bathla.
- **Methodology:** LSTM and SVR.
- **Description:** Stock price movement is non-linear and complex. Traditional approaches such as Linear Regression and Support Vector Regression were used but accuracy was not adequate. Artificial Neural Network was deployed to predict stock prices but as stock prices are time-series based, recurrent neural network was applied to further improve prediction accuracy. In RNN, there is limitation of not able to store high dependencies and also vanishing gradient descent issue exists. Therefore, data scientists and analysts applied LSTM to predict stock price movement. Stock market is one of the most important institutions of any economy. In the past, several researchers have proposed innovative approaches to predict stock prices. Researchers have used Linear regression, Support Vector Machines, ARIMA and traditional machine learning techniques for stock price prediction. The limitations of these approaches are that accuracy is not adequate. There is need of better approachwhich can predict high variations prices with significant accuracy. In this research paper, deep learning is applied on different stock index datasets to predict stock prices. Experiment analysis have proved that accuracy is better as compared to existing approaches.

  Here in this paper, Support Vector Regression is applied to predict stock prices. SVM is used by researchers to achieve high precision for financial market prediction.

  In this research work, deep learning is applied to improve prediction accuracy. LSTM is used with adam optimizer and sigmoid activation function. Mean AbsolutePercentage Error is used as evaluation metric. Experiment is conducted on different stock index such as S&P 500, NYSE, NSE, BSE, Dow Jones Industrial Average and NASDAQ. Experiment analysis proves that LSTM outperforms SVR and provides better prediction accuracy.
- **Advantage:** Model is trained using one dataset and deployed on other dataset and Statistical model is proposed to predict future stock prices.
- **Limitation:** There is a need of more datasets for experimental analysis.

## • 2.6 Survey 6: Cryptocurrency

- **Title:** Time Series Analysis of Cryptocurrencies Using Deep Learning & Fbprophet.
- **Author:** Yash Indulkar.
- **Methodology:** LSTM and Fbprophet.
- **Description:** cryptocurrency prediction and analysis using different algorithms, themajor cryptocurrency took into account for analysis and prediction are Bitcoin (BTC), Ethereum (ETH), Chainlink (LINK), Bitcoin Cash (BTC), XRP (XRP). Nowadays, investing in cryptocurrency has become a major deal, with huge cash flow and billions of industries which has taken over the small industry that was over the past. With this investment, it is important to understand the high & low of a particular cryptocurrency and what output will be generated with such decisions. The machine learning industry has advanced to a great extent and it would further do, this advancement has led us to a bigger problem-solving technique, that is prediction of data or analysis of trend which can be in any format. The format in this paper is a time series analysis of the daily high low-close of digital currency. The algorithms used for such analysis is LSTM (Long Short-Term Memory) which is part of Deep Learning and further Fbprophet which is an Auto Machine Learning for prediction is used. The metric used for the analysis of the algorithm is MAE (Mean Absolute Error). The programming language used is Python, which solves the majority of use cases.

  The cryptocurrency used is Bitcoin (BTC), Ethereum (ETH), Chainlink (LINK), Bitcoin Cash (BTC), XRP (XRP). These currencies have different values based on the popularity of each cryptocurrency; estimation of these values may vary depending on each one of them. The deep learning algorithm used for the analysis of such digital currency is LSTM, it stands for Long Short-Term Memory which is a special RNN (Recurrent Neural Network). Unlike the feed-forward neural network, the LSTM has a feedback connection which makes it special. The other automatic forecasting procedure used is Fbprophet which is an auto machine learning for prediction of data based on events, which fits the datasets based on a linear model, that can shift the model to nonlinear or logistics growth with tweaking of arguments.
- **Advantage:** The mean absolute error is very less.
- **Limitation:** Here the prediction is done for only for 4 different cryptocurrency.

## 2.7 Comparative Analysis

| Reference | Algorithm/ Technique | Platform used | Performance Metrics | Advantage | Drawback |
|---|---|---|---|---|---|
| [1] | Linear Regression and SVM | Jupyter Notebook | Accuracy: 94% | Forecasting prices of Amazon Stock Market | Predicting onlyfor Amazon stock with limited dataset |
| [2] | Decision tree and Regression Techniques | Jupyter Notebook | Accuracy: 95% | Forecasting prices of popular cryptocurrency i.e., Bitcoin | Predicting onlyBitcoin prices |
| [3] | Bayesian Regression model | Jupyter Notebook | Unknown | Pattern of deviation of real price is simpler compared to price time series | Accuracy is unknown and predicting onlyfor bitcoin |
| [4] | Linear Regression and Decision Tree Regression | Jupyter Notebook | Accuracy: 99% | High accuracy of 99% | Considering dataset of only 1year |
| [5] | LSTM and SVR | Jupyter Notebook | Unknown | Statistical model is proposed to predict future stock prices | Less number of datasets are used |
| [6] | LSTM and Fbprophet | Jupyter Notebook | Unknown | Mean absolute error is very less | Prediction is done for only for 4 different cryptocurrency |

**Table 2.1: Comparative Analysis**

# Chapter 3

# SYSTEM ANALYSIS

## 3.1 Existing System

Fundamental analysis, which looks at a stock's and cryptocurrency's past performance, identifying patterns in stock and cryptocurrency price variation, is one of the traditional approaches to stock and cryptocurrency price prediction. The latter is often accomplished with the help of Genetic Algorithms (GA) or Artificial Neural Networks (ANN), although these fail to capture long-term temporal dependencies in stock and cryptocurrency values. Another important concern with utilizing simple ANNs for stock and cryptocurrency prediction is the exploding / vanishing gradient phenomenon, which occurs when the weights of a big network become either too large or too tiny (respectively), dramatically reducing their convergence to the ideal value. This is usually caused by two factors weights are initialized randomly, and the weights closer to the end of the network also tend to change a lot more than those at the beginning.

Another approach is to reduce the dimensionality of the input data and use feature selectionalgorithms to shortlist a core set of features that have the greatest impact on stock and cryptocurrency prices or currency exchange rates across markets. However, because it doesnot take into account the entire history of trends, this method does not consider long-term trading strategies; additionally, there is no provision for outlier detection.

### 3.1.1 Drawbacks

- Predicting only for 1 or 2 stocks and cryptocurrencies previously was not suitable for current real time decision making to help investors.
- Artificial Neural Networks algorithms used previously to predict stocks and cryptocurrencies fail to predict exact future prices of current stocks and cryptocurrencies.
- The dataset considered for previously stocks and cryptocurrencies have variations and many external factors effecting the stocks and cryptocurrencies on the daily basis.
- The dataset considered previously was far lesser in comparison to dataset generated in the current time.

## 3.1 Proposed System

In the proposed system, we are predicting future stock values of various companies around the world as well as future cryptocurrency prices for various cryptocurrencies. By fetching data from yahoo finance for stocks and twelve data websites for cryptocurrencies through web scraping. In the data retrieved, the close prices are extracted and converted between 0 to 1 range using MinMaxScaler for ease of plotting the line graphs. Using LSTM model, wetrain and predict future stocks and cryptocurrencies prices. Later, we plot and display forecasted prices for the searched stock and cryptocurrency by the user respectively.

## 3.2 Feasibility Study

The feasibility analysis of the project is mentioned below stating why the project is better than the other existing system in terms of technical, economical and operational feasibility.

### 3.2.1 Technical Feasibility

In the proposed system, a system with windows 11 operating system, 16 GB of RAM, minimum of 100MB of hard disk space, and a graphics card with minimum of 2 GB VRAM is considered for stock and cryptocurrency predictions. The software used here to code machine learning model is jupyter notebook, and visual studio for developing the streamlit web application which integrates the main LSTM model and related code. Libraries such asmatplotlib, pandas, streamlit, tensorflow is considered for calculating and displaying the predictions of stock and cryptocurrency.

### 3.2.2 Operational Feasibility

The system helps investors and new users in understanding the stock and cryptocurrency market by showing them the predicted future stocks and cryptocurrencies with fine accuracy. It guides the users to make decisions whether to buy or sell the stock and cryptocurrency based on the predictions and future prices shown in the website.

### 3.2.3 Economical Feasibility

The system uses free and open source softwares such as jupyter notebook of anaconda distribution for coding LSTM Model, and for coding web application, visual studio is used.For fetching data sets, free websites such as yahoo finance and twelve data websites are used.

# CHAPTER 4

# SYSTEM REQUIREMENTS

## 4.1 Hardware Requirements

- **Processor** **:** Intel i5 or Amd 4000 series or Apple M1 or above.
- **RAM** **:** Minimum 8GB or more.
- **Hard Disk** **:** Minimum 2 GB of space Hard Disk.
- **Display** **:** 1920 x 1080 px , 16:9 aspect ratio.

## 4.2 Software Requirements

- **Operating System** **:** Windows 10 and above.
- **IDE** **:** Visual Studio and Jupiter Notebook.
- **Dataset Format** **: .**csv file.
- **Visualization** **:** matplotlib.

# CHAPTER 5
# PROJECT DESCRIPTION

## 5.1 Problem Definition

- Investors are familiar with the saying, “buy low, sell high” but this does not provide enough context to make proper investment decisions. Before an investor invests in any stock or cryptocurrency, he needs to be aware how the market behaves.
- Investing in a good stock or cryptocurrency but at a bad time can have disastrous results, while investment in a mediocre stock or cryptocurrency at the right time can bear profits.
- Financial investors of today are facing this problem of trading as they do not properly understand as to which stocks or cryptocurrencies to buy or which to sell in order to get optimum profits.
- Predicting long term value is relatively easy than predicting on day-to-day basis as the price fluctuate rapidly every hour based on world events.
- So, our model predicts the price of stocks and cryptocurrencies, which helps the investors to invest in appropriate stocks and crypto coins.

## 5.2 Overview of The Project

The proposed project predicts future price of stocks and cryptocurrencies using machine learning model and displays the results as graphs in the website. The data is fetched from yahoo finance for stocks and twelve data website for cryptocurrency using web scraping. The machine learning model used is RNN LSTM model which takes the close price of stocks and cryptocurrencies and trains the model, later the model is tested using the test data sets. The predicted graphs and prices are displayed on the website which is built using python stream lit framework.

In the website the user as to enter the stock or cryptocurrency ticker. The model calculates the predictions and the results are displayed as line graphs and predicted price value on thewebsite.

## 5.3 System Architecture

The dataset we use for the proposed project is been taken from yahoo finance for stock and twelve data for cryptocurrency using web scraping. But, this data set is in raw format. The data set is a collection of valuation of stock market information about companies and cryptocurrencies. The initial step is to convert raw data into processed data, which is done by feature extraction. Since the raw data collected have multiple attributes but only some of those attributes are needed for prediction. We split the data into training and test data. We use the training data and train the model, and then feed test data into the model and predict future price of stock and cryptocurrency. Then using predicted prices, we plot the graph for stocks and display for cryptocurrencies.

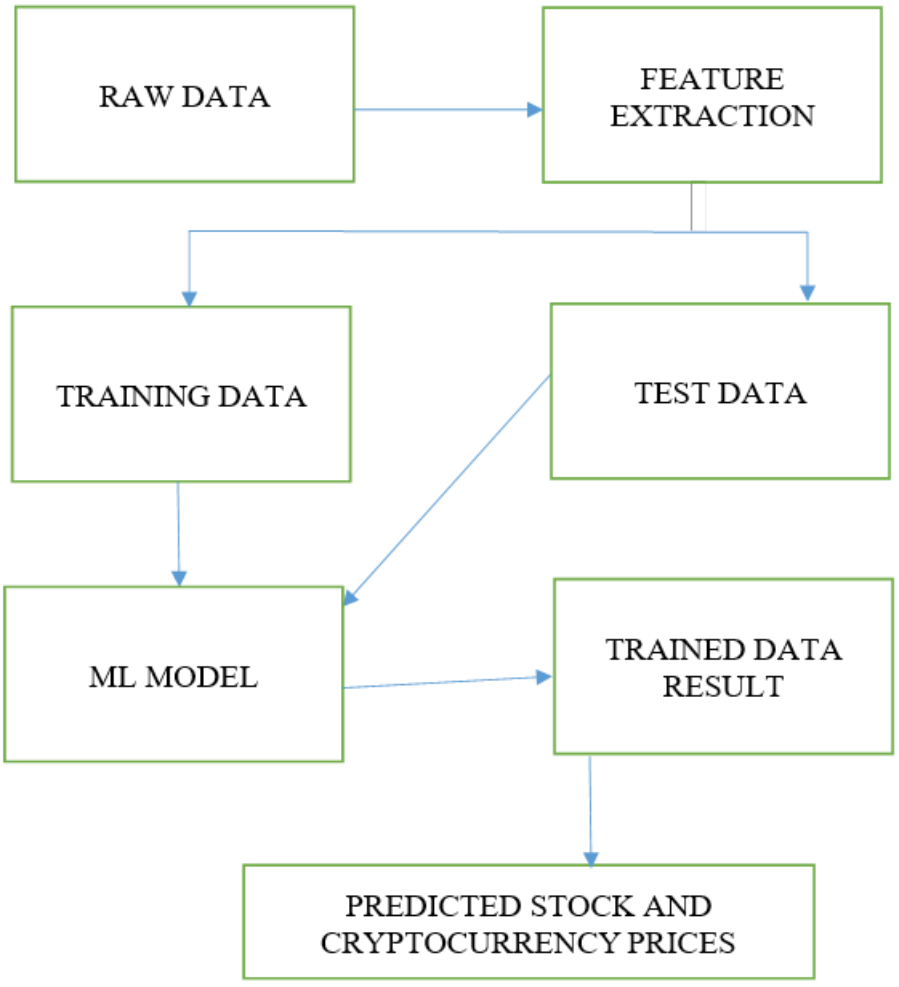



**Fig 5.3: System Architecture**

## 5.4 Dataflow Diagram

We are taking a company or cryptocurrency for fetching the data from the panda's data reader library then we are feeding data into model. Then we train the data to predict the stock and cryptocurrency for certain number of days. Then we plot the graph for prediction.

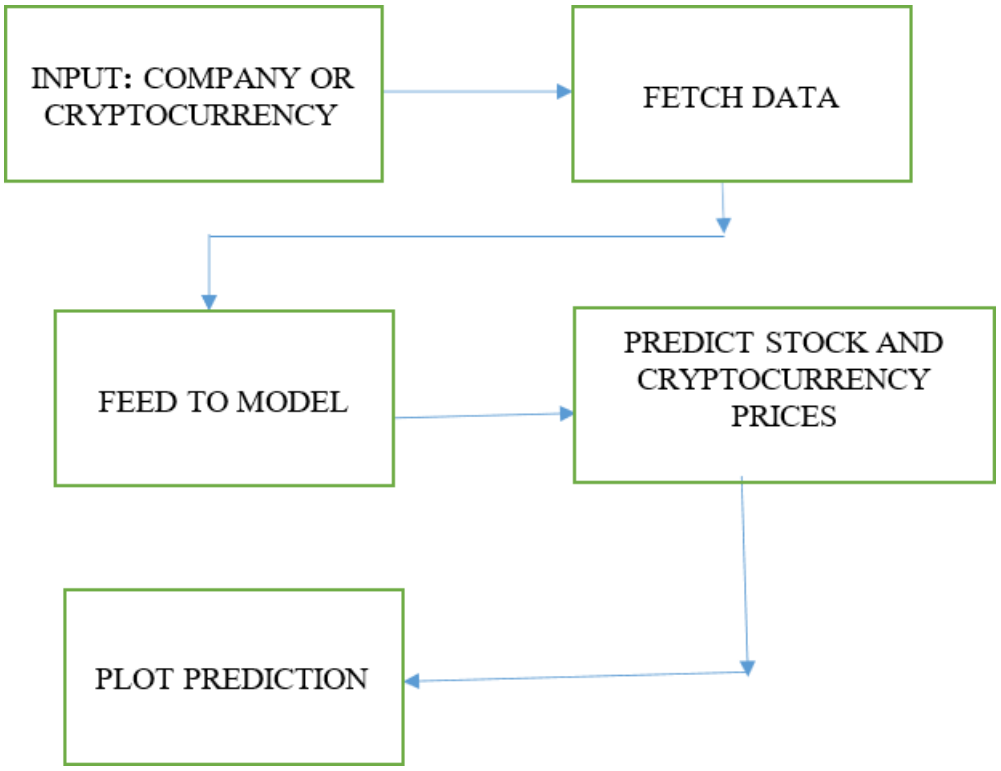



**Fig 5.4: Data flow diagram**

## 5.5 Use Case diagram

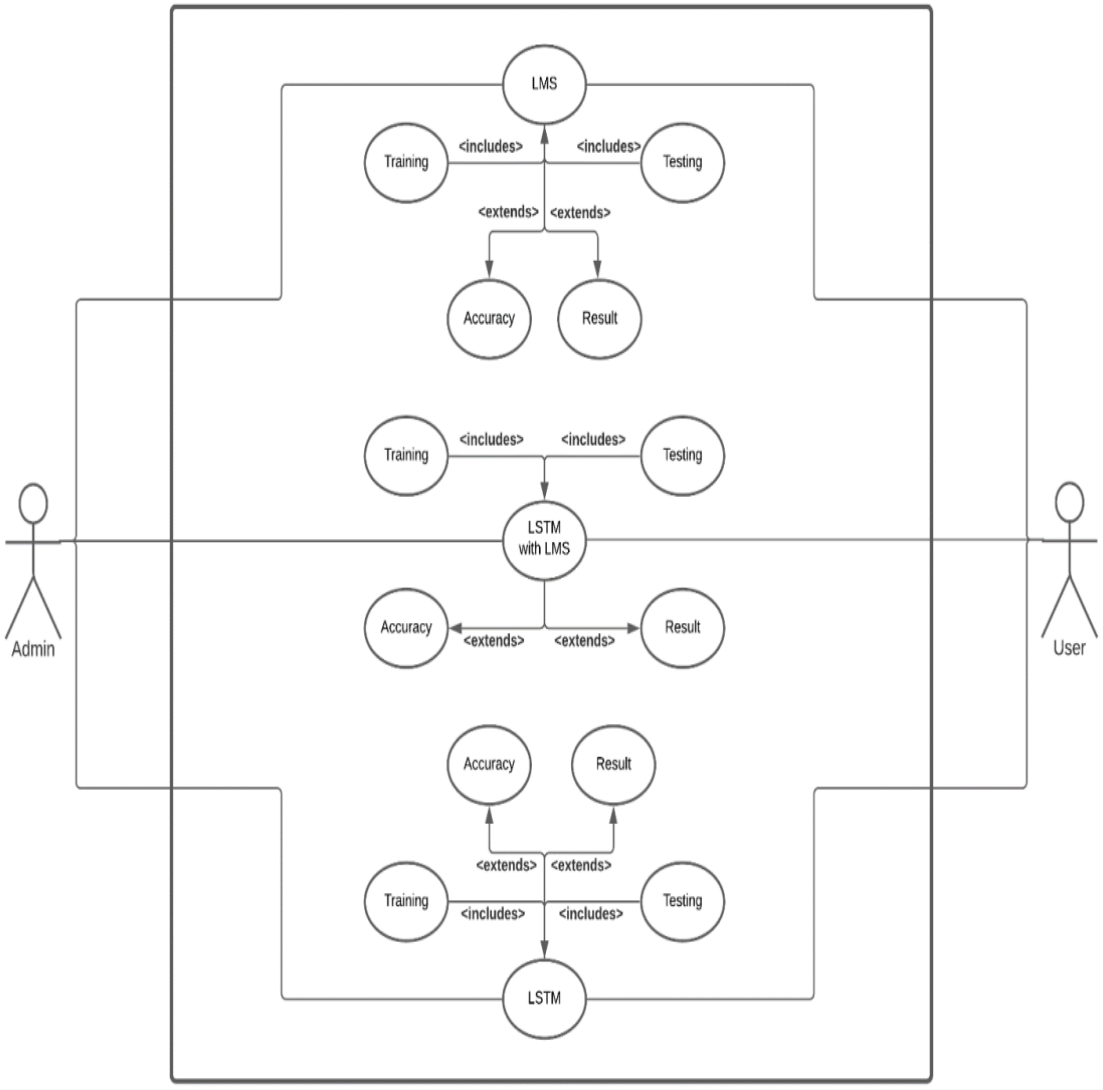



**Fig 5.5 Use case diagram.**

In the Unified Modelling Language (UML), a use case diagram can summarize the details of your system's users (also known as actors) and their interactions with the system. To build one, you'll use a set of specialized symbols and connectors. An effective use case diagram can help your team discuss and represent:

• Scenarios in which your system or application interacts with people, organizations, or external systems.

• Goals that your system or application helps those entities (known as actors) achieve.

• The scope of your system

## 5.6 Module Description

### 5.6.1 Module 1: Collecting data from user

**Description:**

For stocks, search bar is provided in the website to enter the company symbol were, prediction is done for a company based on what is entered in the search bar.

For cryptocurrency, search bar is provided in the website to enter the cryptocurrency symbol were, prediction is done for a cryptocurrency based on what is entered in the search bar.

**Relevant diagram:**

- **For Stocks the search bar.**

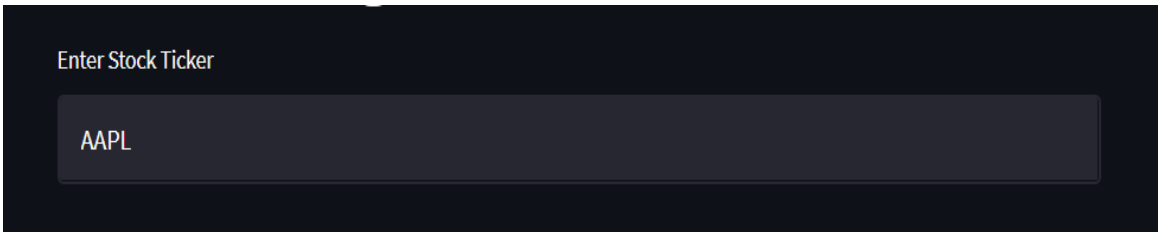



**Fig 5.6.1.1: Search bar for stocks on the webpage.**

- **For Cryptocurrency the search bar.**

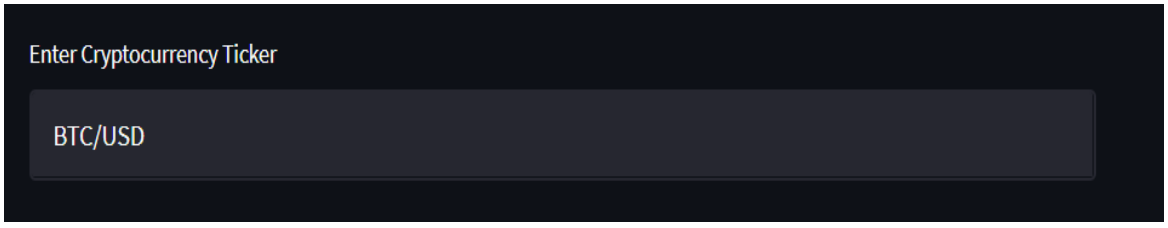



**Fig 5.6.1.2: Search bar for cryptocurrency on the webpage.**

**Source code:**

- **For stocks**

```
user_input= st.text_input('Enter Stock Ticker','AAPL’)
```

- **For cryptocurrency**

```
user_input = st.text_input('Enter Cryptocurrency Ticker','BTC/USD')
```

- ## 5.6.2 Module 2: Data Retrieval and Data Preprocessing

**Description:**

For stocks, data is retrieved using web scraping from yahoo finance using Pandas Data Reader library. In preprocessing of the data, we retrieve close column from the dataframe and scale it between 0 to 1 using MinMaxScaler from sklearn library.

For cryptocurrency, data is retrieved from a URL in json format using the api key provided by twelve data website. Later we convert it to a dataframe and retrieve the close column and scale it between 0 to 1.

**Source code:**

- **For Stocks:**

```
start= datetime.strftime(datetime.now() - timedelta(3650),'%Y-%m-%d %H:%M:%S')
today= date.today()
end=today.strftime("%Y-%m-%d")
user_input= st.text_input('Enter Stock Ticker','AAPL’)

if len(user_input)==0:
   st.warning("Enter stock symbol")return
finance= 'yahoo'
try:
      df=data.DataReader(user_input,finance,start,end)
except:
st.warning("Invalid Stock Name")return
if len(df.index)<365:
st.warning("OOPS! Insufficient data to predict, Stock should be minimum 1 year old")
       return
scaler=MinMaxScaler(feature_range=(0,1))
df1=scaler.fit_transform(np.array(df1).reshape(-1,1))
```

- **For cryptocurrency:**

```
api_key = '2f0fdd5d49254c06b7dc54cc45a4b628'interval = '5min'
order = 'asc'
start_date = datetime.strftime(datetime.now() - timedelta(20),'%Y-%m-%d %H:%M:%S')
end_date = datetime.strftime(datetime.now() - timedelta(3),'%Y-%m-%d %H:%M:%S')
st.title('Cryptocurrency Price Prediction and Forecasting')
```

```
start_date = datetime.strftime(datetime.now() - timedelta(20),'%Y-%m-%d %H:%M:%S') end_date = datetime.strftime(datetime.now() - timedelta(3),'%Y-%m-%d %H:%M:%S') st.title('Cryptocurrency Price Prediction and Forecasting')
user_input = st.text_input('Enter Cryptocurrency Ticker','BTC/USD')
if len(user_input)==0:
    st.warning("Enter cryptocurrency symbol")return
try:
    api_url = f'https://api.twelvedata.com/time_series?
    symbol={user_input}&start_date={start_date}&end_date={end_date}&interval={interval} &order={order}&apikey={api_key}' data = requests.get(api_url).json()
    data_final = pd.DataFrame(data['values'])
except:
    st.warning("Invalid Cryptocurrency Symbol")
    return
if len(data_final.index)<21:
   st.warning("OOPS! Insufficient data to predict,Cryptocurrency should be minimun 21 days old")
   return
scaler = MinMaxScaler(feature_range=(0,1))
scaled_data = scaler.fit_transform(data_final['close'].values.reshape(-1,1))
```

- ### 5.6.3 Module 3: Splitting the data into training and test sets

**Description:**

For stocks, basically available 10 years of data is splitted into 65% training and 35% testing sets using the length of the dataframe. Both sets are reshaped into 3 dimensional array.

For cryptocurrency, data retrieved from the twelve data website is splitted into 17 days for the training set and 3 days for the testing set. Both sets are reshaped into 3 dimensional array.

**Source Code:**

- **For stocks:**

```
training_size=int(len(df1)*0.65)
test_size=len(df1)-training_size
train_data,test_data=df1[0:training_size,:],df1[training_size:len(df1),:1]
def create_dataset(dataset, time_step=1):
    dataX, dataY = [], []
    for i in range(len(dataset)-time_step-1):
        a = dataset[i:(i+time_step), 0] ###i=0, 0,1,2,3-----99 100
        dataX.append(a)
        dataY.append(dataset[i + time_step, 0])
    return np.array(dataX), np.array(dataY)
```

- **For Cryptocurrency:**

```
time_intervals_to_train = 24
prediction_interval = 12
x_train = []
y_train = []
for i in range(time_intervals_to_train, len(scaled_data) - prediction_interval):
    x_train.append(scaled_data[i - time_intervals_to_train: i, 0])
    y_train.append(scaled_data[i + prediction_interval, 0])
x_train = np.array(x_train)
y_train = np.array(y_train)
x_test= []
for x in range(time_intervals_to_train, len(model_inputs)):
   x_test.append(model_inputs[x - time_intervals_to_train: x, 0])
x_test = np.array(x_test)
x_test = np.reshape(x_test, (x_test.shape[0], x_test.shape[1], 1))
```

- ## 5.6.4 Module 4: Building Model

**Description:**

For stocks, LSTM model is built using 3 LSTM layers and a dense layer from keras api. In our analysis, we consider the mean square error loss and adam optimizer.

For cryptocurrency, LSTM model is built using 3 LSTM layers and a dense layer from keras api. The first layer as dropout of 40%, second layer as 20% and the third layer as 10%. As previously we consider the mean square error loss and adam optimizer.

**Relevant Diagram:**

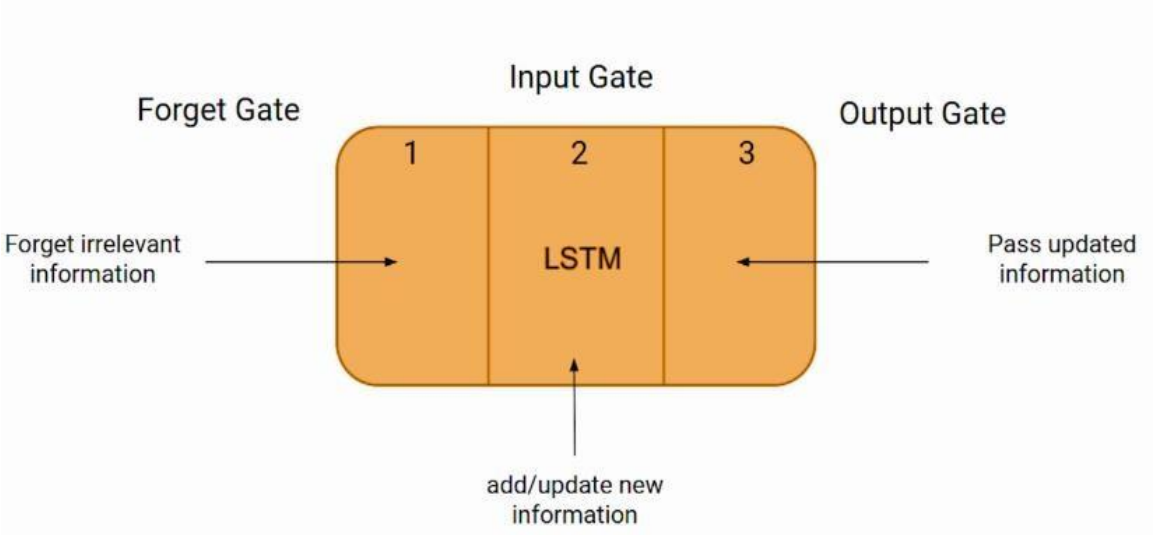



**Fig 5.6.4: LSTM Model**

**Source code:**

- **For Stocks:**

```
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense
from tensorflow.keras.layers import LSTM
model=Sequential()
model.add(LSTM(50,return_sequences=True,input_shape=(100,1)))
model.add(LSTM(50,return_sequences=True))
model.add(LSTM(50))
model.add(Dense(1))
model.compile(loss='mean_squared_error',optimizer='adam’)
```

- **For cryptocurrency:**

```
from tensorflow.keras.layers import Dense, Dropout, LSTM
from tensorflow.keras.models import Sequential
```

```
model = Sequential()
model.add(LSTM(128, return_sequences= True, input_shape = (x_train.shape[1], 1), activation='relu'))
model.add(Dropout(0.4))
model.add(LSTM(64, return_sequences= True, activation ='relu'))
model.add(Dropout(0.3))
model.add(LSTM(32, activation ='relu'))
model.add(Dropout(0.2))
model.add(Dense(1, activation ='sigmoid'))
model.compile(loss='mean_squared_error', optimizer='adam', metrics= ['accuracy'])
```

- **Module 5.6.5: Prediction**

**Description:**

For stocks, the predicted series of values by machine learning model are displayed in visual representation using matplotlib library in the form of line plot graph.

For cryptocurrency, the predicted value by machine learning model is displayed.

**Source Code:**

**For Stocks:**

```
from numpy import array
lst_output=[]
n_steps=100
i=0
while(i<30):
    if(len(temp_input)>100):
        x_input=np.array(temp_input[1:])
        print("{} day inpu{}".format(i,x_input))
        x_input=x_input.reshape(1,-1)
        x_input = x_input.reshape((1, n_steps, 1))
        yhat = model.predict(x_input, verbose=0)
        print("{} day output {}".format(i,yhat))
        temp_input.extend(yhat[0].tolist())
        temp_input=temp_input[1:]
        lst_output.extend(yhat.tolist())
    i=i+1
```

```
else:
    x_input = x_input.reshape((1,
    n_steps,1))yhat =
    model.predict(x_input, verbose=0)
    print(yhat[0])
    temp_input.extend(yhat[0].tolist())
    print(len(temp_input))
    lst_output.extend(yhat.tolist())
    i=i+1
```

- **For Cryptocurrency:**

```
last_data = model_inputs[len(model_inputs) - time_intervals_to_train: len(model_inputs), 0]
last_data = np.array(last_data)
prediction = model.predict(last_data)
```

# CHAPTER 6

# SYSTEM TESTING

## 6.1 Introduction

As System Testing refers to testing the system as a whole, all the modules/components are integrated in order to verify if the system works as expected or not. This system testing phase tests the design and behavior of the system and also the expected outcome in various scenarios. In the present situation, there is a need to deliver better and faster services. to provide an efficient and reliable device, the device was put through various testing such as:

- **Performance Testing**

Performance testing is the process of determining the speed, responsiveness and stability of a computer, network, software program or device under a workload.

- **Integration Testing**

System Integration Testing is defined as a type of software testing carried out in an integrated hardware and software environment to verify the behavior of the complete system.

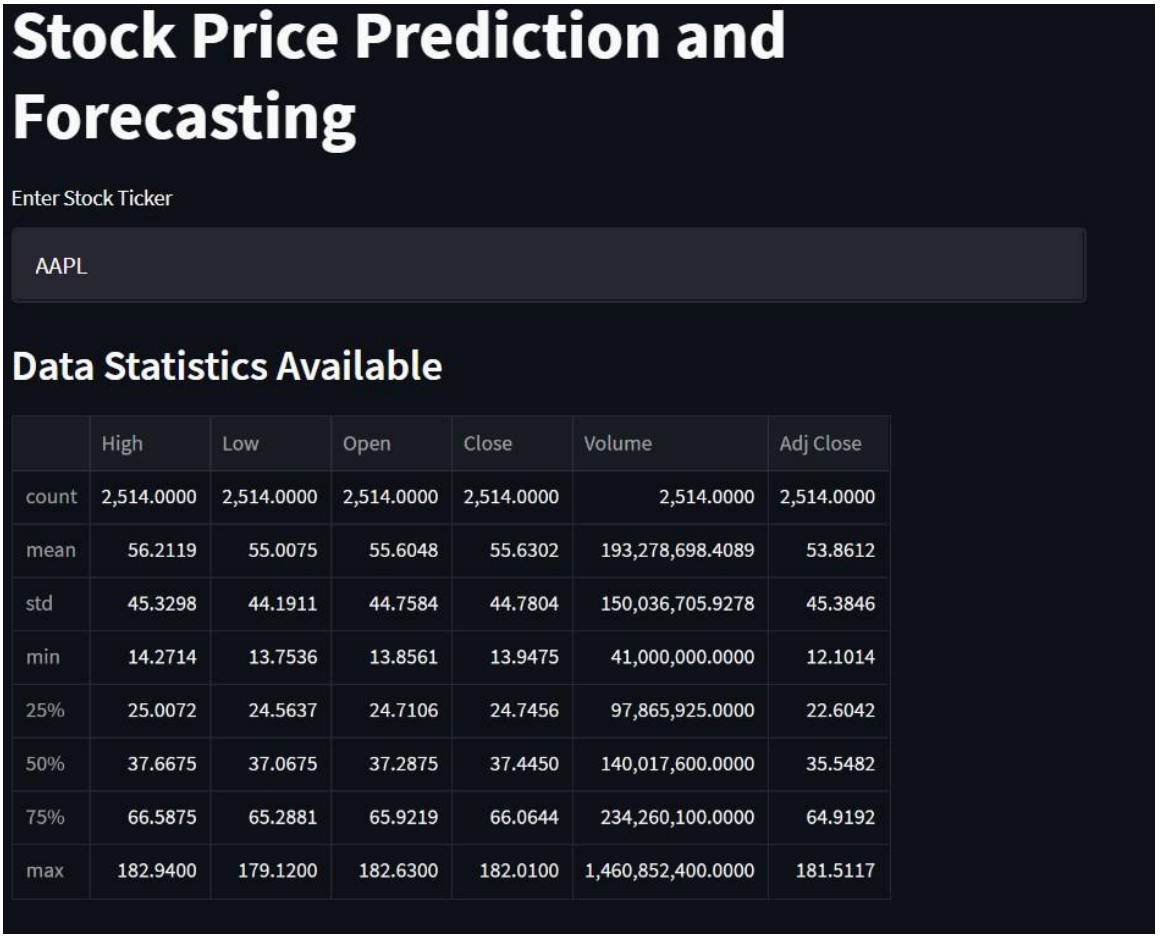
- **Reliability and Scalability Testing**

Reliability and Scalability is important for building stock and cryptocurrency test environment which involves simulation of model.

- **Usability Testing**

There are so many data of different shape and form factors that are used by the users. Moreover, the perception also varies from one user to other.

## 6.2 Test Cases




| | High | Low | Open | Close | Volume | Adj Close |
|---|---|---|---|---|---|---|
| count | 2,514.0000 | 2,514.0000 | 2,514.0000 | 2,514.0000 | 2,514.0000 | 2,514.0000 |
| mean | 56.2119 | 55.0075 | 55.6048 | 55.6302 | 193,278,698.4089 | 53.8612 |
| std | 45.3298 | 44.1911 | 44.7584 | 44.7804 | 150,036,705.9278 | 45.3846 |
| min | 14.2714 | 13.7536 | 13.8561 | 13.9475 | 41,000,000.0000 | 12.1014 |
| 25% | 25.0072 | 24.5637 | 24.7106 | 24.7456 | 97,865,925.0000 | 22.6042 |
| 50% | 37.6675 | 37.0675 | 37.2875 | 37.4450 | 140,017,600.0000 | 35.5482 |
| 75% | 66.5875 | 65.2881 | 65.9219 | 66.0644 | 234,260,100.0000 | 64.9192 |
| max | 182.9400 | 179.1200 | 182.6300 | 182.0100 | 1,460,852,400.0000 | 181.5117 |

**Fig 6.2.1: User enters valid stock symbol.**

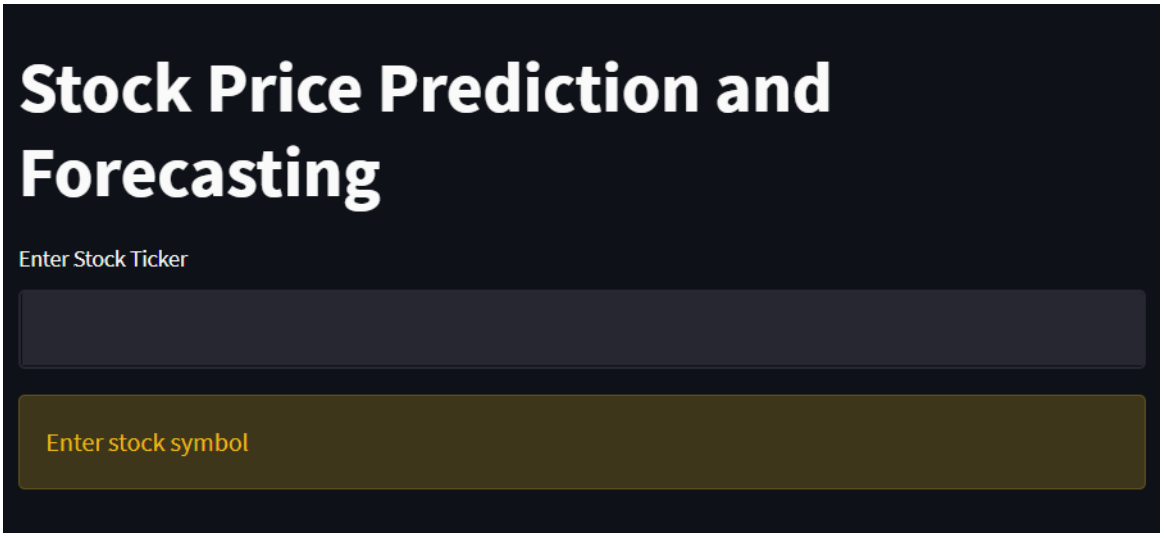



**Fig 6.2.2: User does not enter stock symbol.**

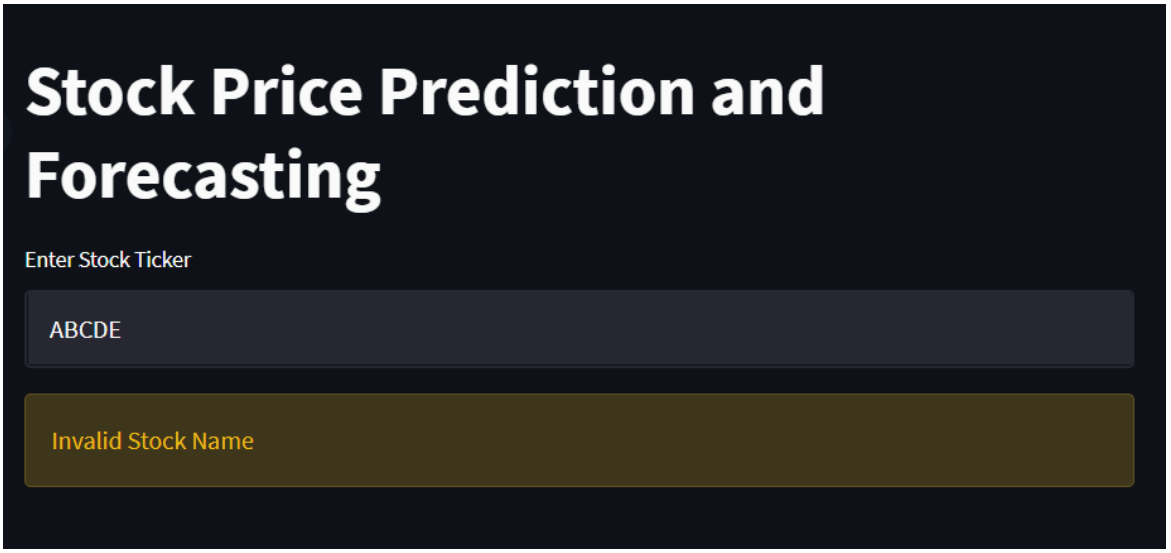



**Fig 6.2.3: User enters invalid stock symbol.**

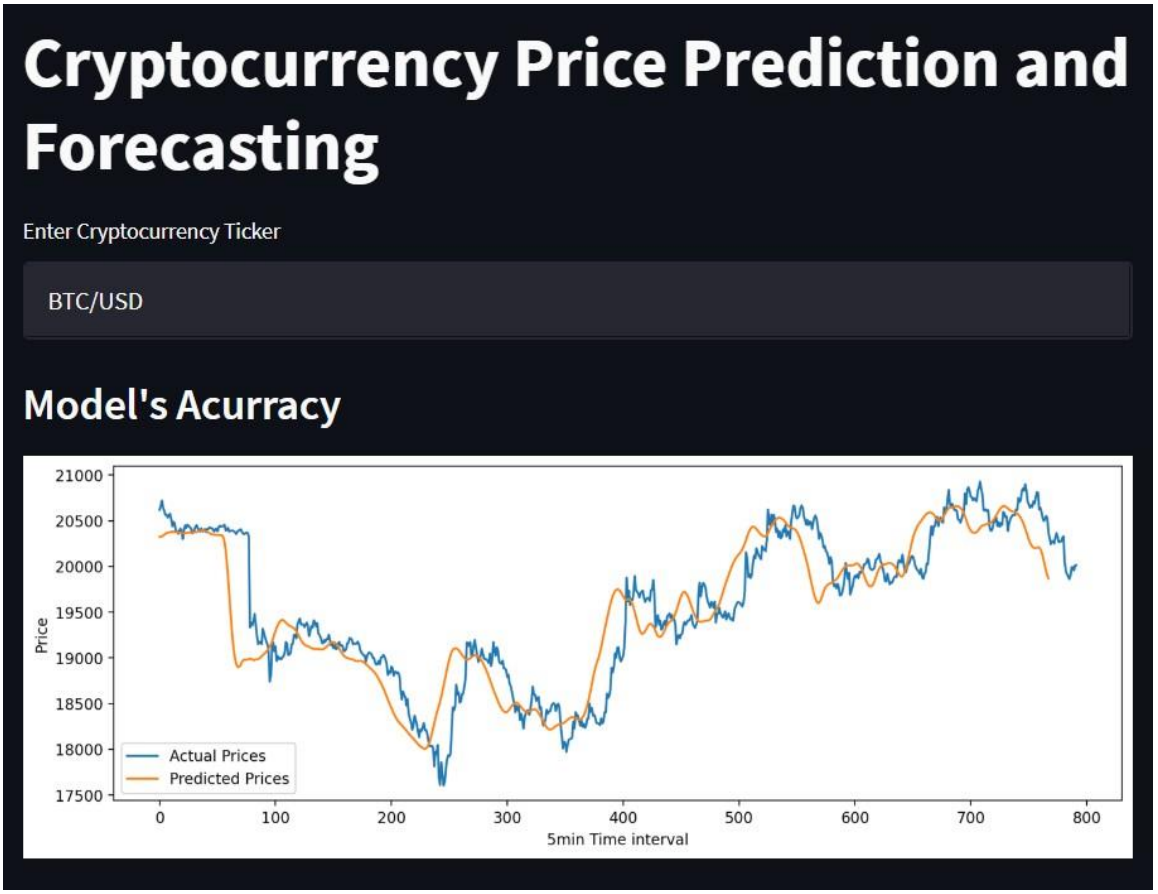



**Fig 6.2.4: User enters valid cryptocurrency symbol.**

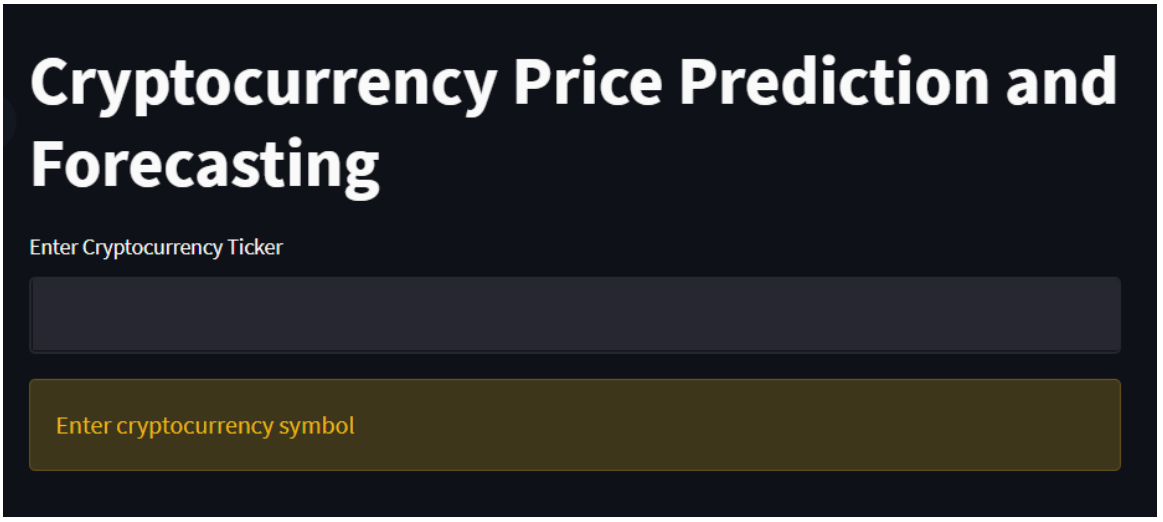



**Fig 6.2.5: User does not enter cryptocurrency symbol.**

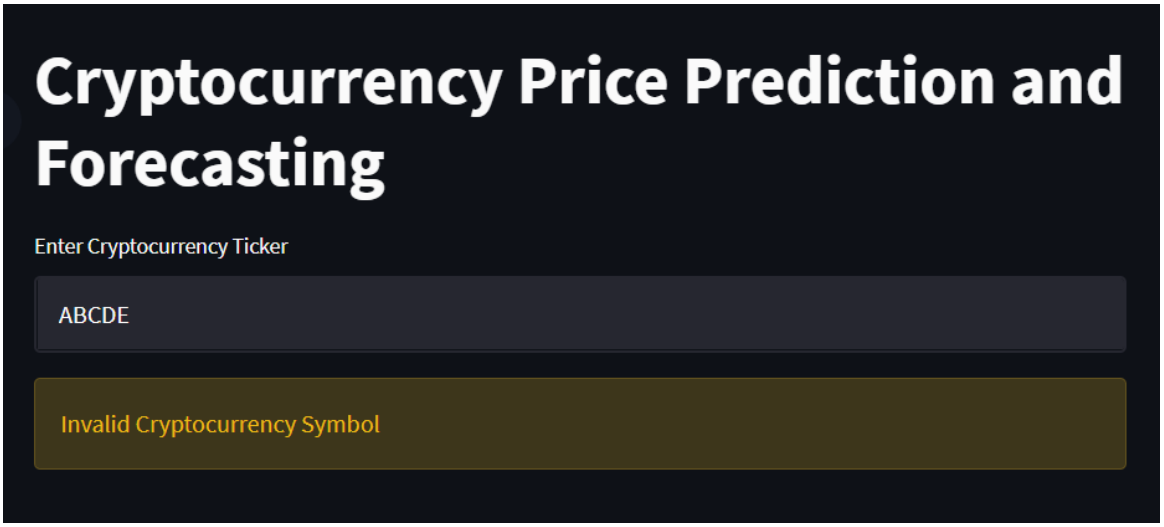



**Fig 6.2.6: User enters invalid cryptocurrency symbol.**

# Chapter 7

# RESULTS AND DISCUSSION

## 7.1 Introduction

The proposed system was developed using jupyter notebook and visual studio IDE. The data is taken from yahoo finance website for stocks and twelve data website for cryptocurrencies. The LSTM model trained and tested using the collected datasets. Then the predictions are displayed on the website as graphs and predicted prices.

## 7.2 Results

### For stocks:


**Fig 7.2.1: Default option for stock.**

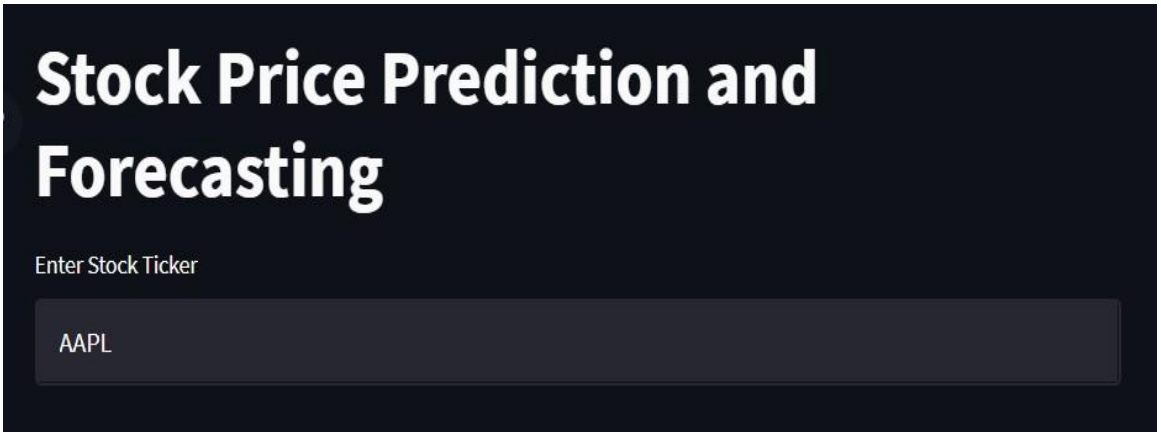



**Fig 7.2.2: Search bar for entering stock symbol.**

## Data Statistics Available

| | High | Low | Open | Close | Volume | Adj Close |
|---|---|---|---|---|---|---|
| count | 2,514.0000 | 2,514.0000 | 2,514.0000 | 2,514.0000 | 2,514.0000 | 2,514.0000 |
| mean | 56.2119 | 55.0075 | 55.6048 | 55.6302 | 193,278,698.4089 | 53.8612 |
| std | 45.3298 | 44.1911 | 44.7584 | 44.7804 | 150,036,705.9278 | 45.3846 |
| min | 14.2714 | 13.7536 | 13.8561 | 13.9475 | 41,000,000.0000 | 12.1014 |
| 25% | 25.0072 | 24.5637 | 24.7106 | 24.7456 | 97,865,925.0000 | 22.6042 |
| 50% | 37.6675 | 37.0675 | 37.2875 | 37.4450 | 140,017,600.0000 | 35.5482 |
| 75% | 66.5875 | 65.2881 | 65.9219 | 66.0644 | 234,260,100.0000 | 64.9192 |
| max | 182.9400 | 179.1200 | 182.6300 | 182.0100 | 1,460,852,400.0000 | 181.5117 |

**Fig 7.2.3: Statistical data of company's stock.**


**Fig 7.2.4: Line graph of closing price vs time chart.**

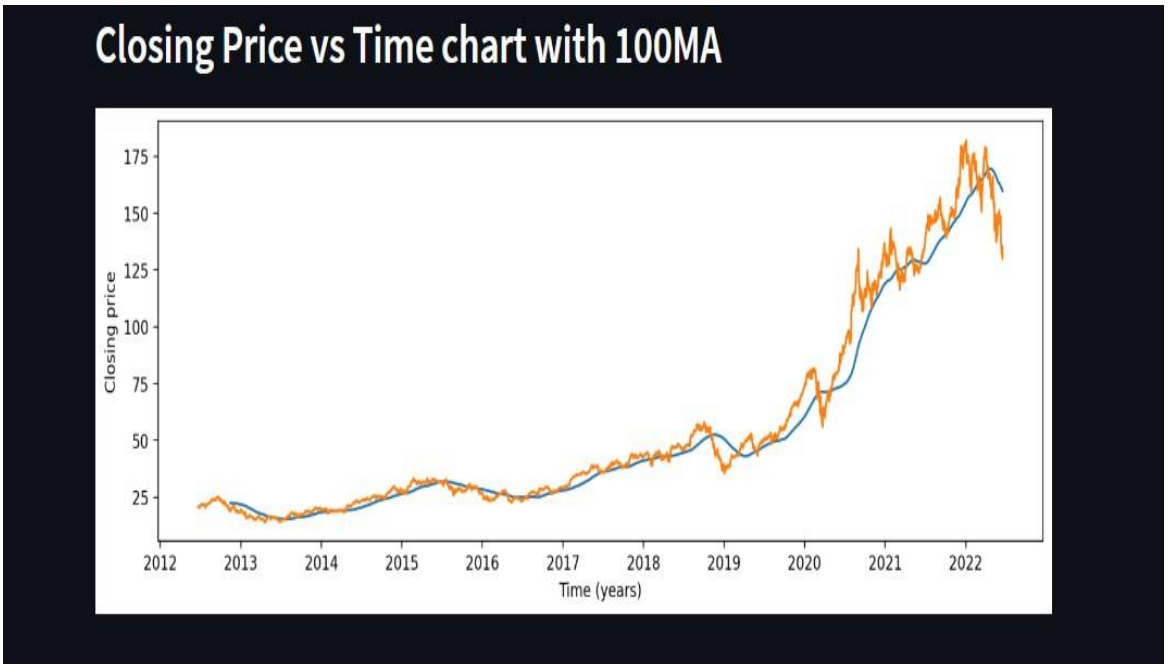



**Fig 7.2.5: Line graph of closing price vs time chart with 100 days moving average.**

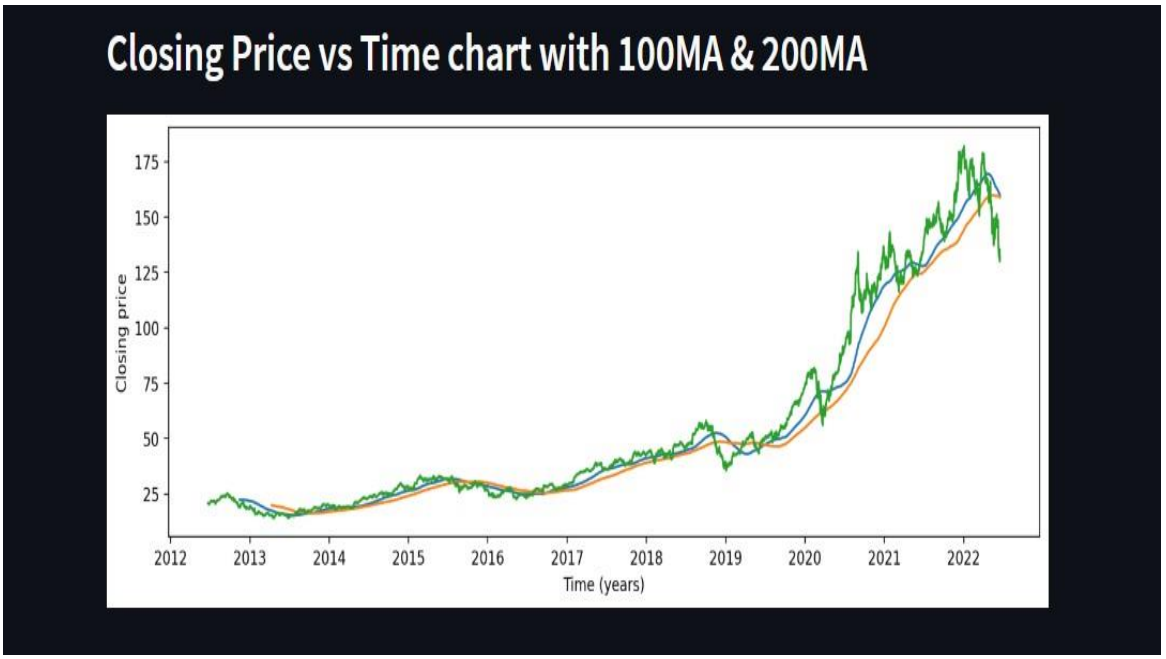



**Fig 7.2.6: Line graph of closing price vs time chart with 100 days and 200 days moving average.**

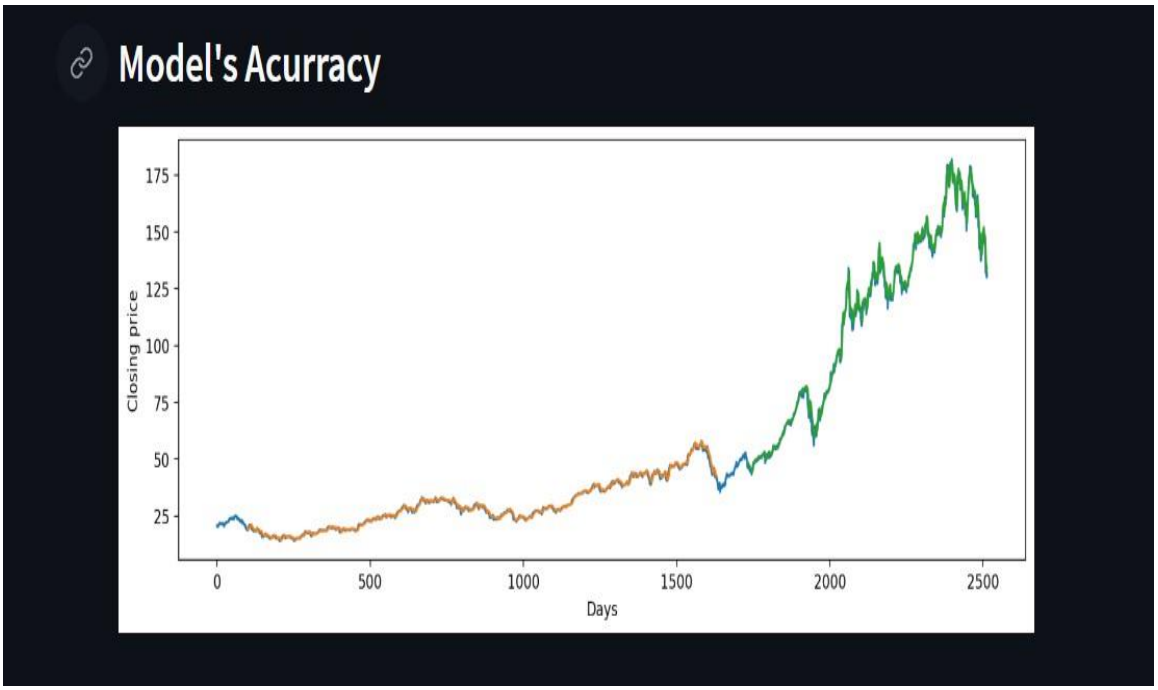



**Fig 7.2.7: LSTM model accuracy for stock prediction.**

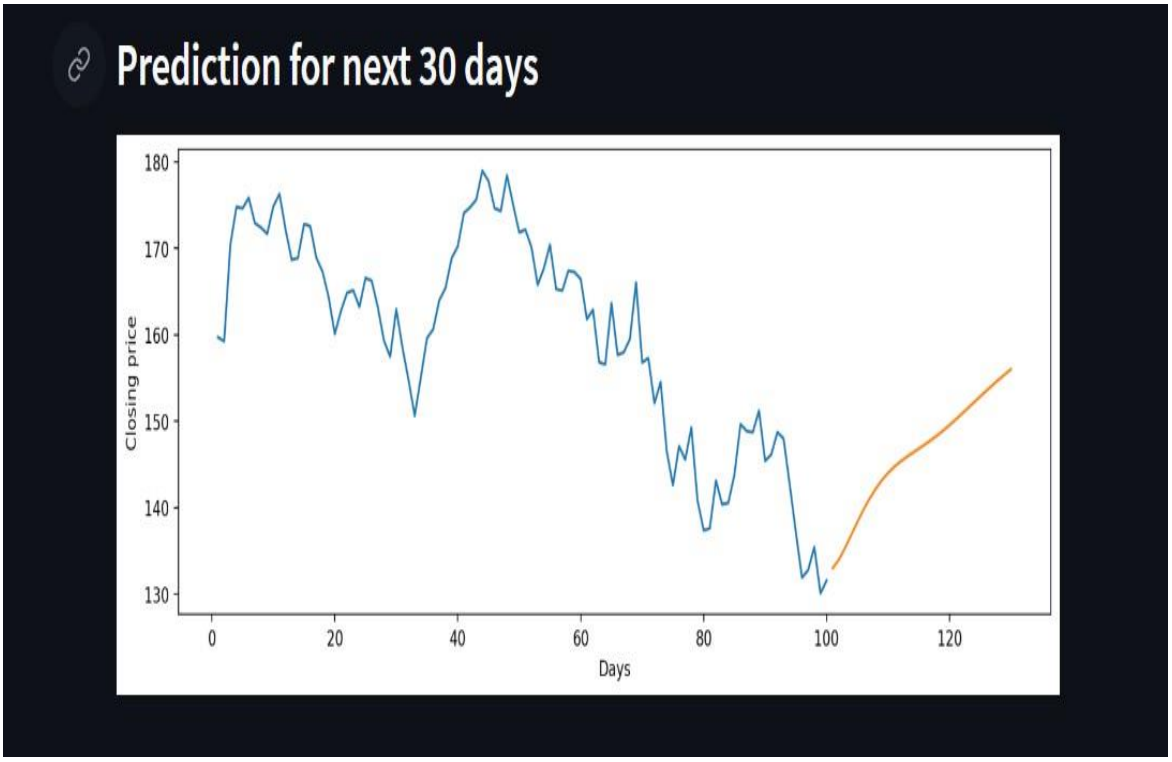



**Fig 7.2.8: Stock prediction for next 30 days.**


**Fig 7.2.9: Forecasted stock prices.**

## For Cryptocurrency:

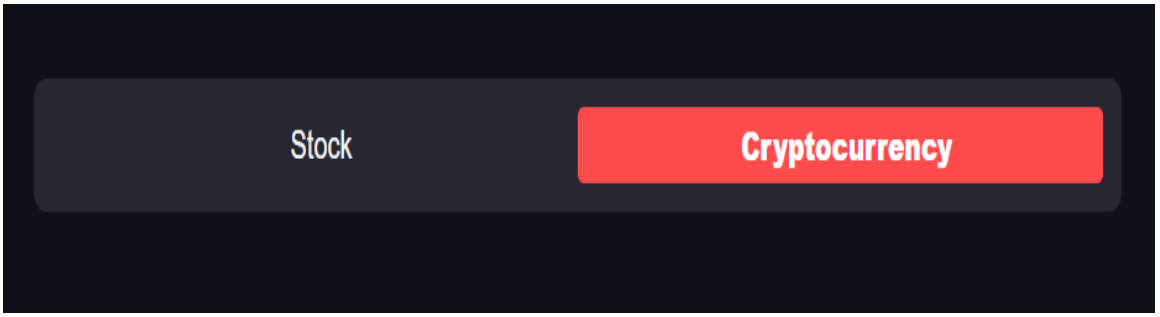



**Fig 7.2.10: When cryptocurrency is selected on the webpage.**


**Fig 7.2.11: Search bar for entering cryptocurrency symbol.**

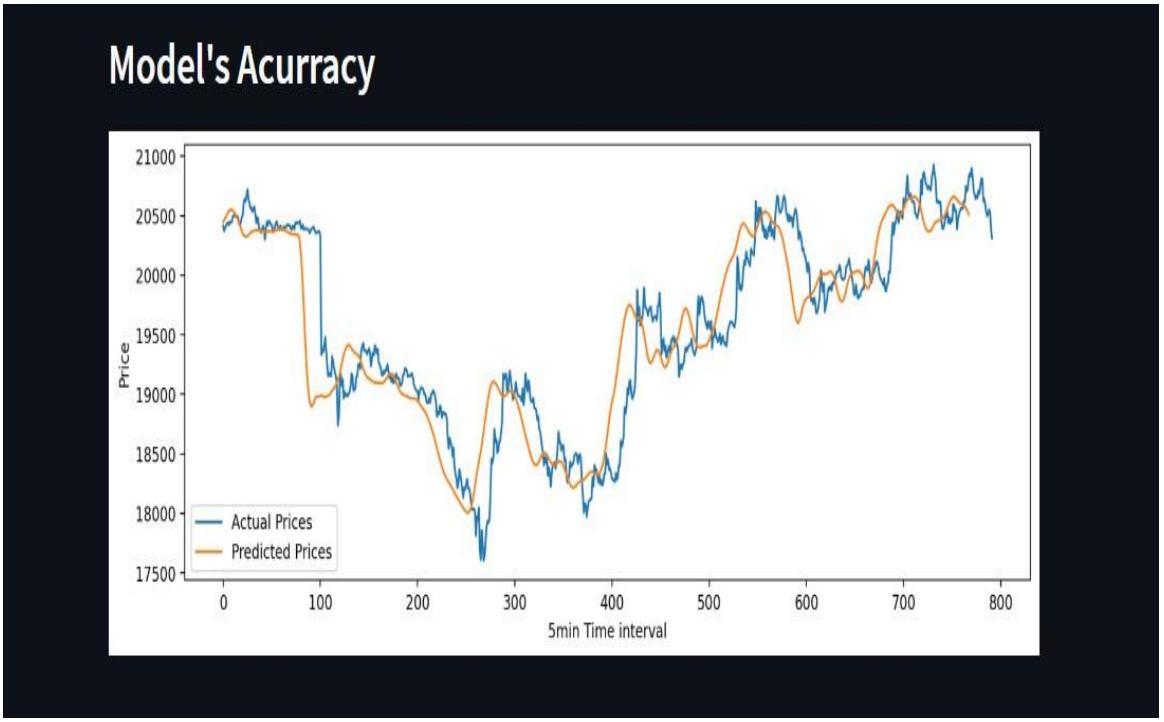



**Fig 7.2.12: LSTM Model accuracy for cryptocurrency.**

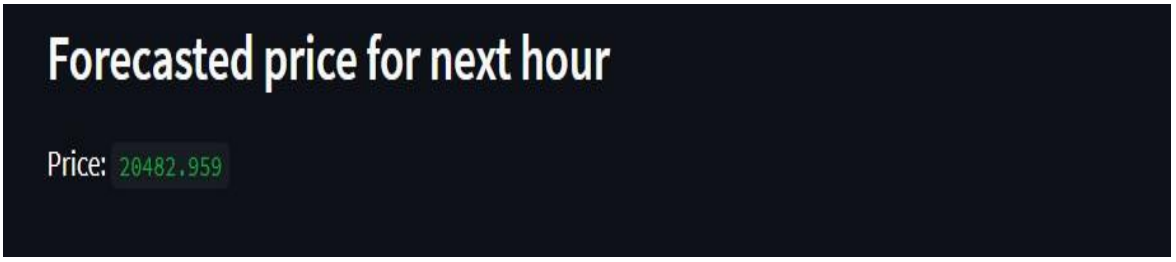



**Fig 7.2.13: Forecasted price for next hour of the searched cryptocurrency.**

# CONCLUSION AND FUTURE WORK

## Conclusion

For stock prediction data is retrieved from yahoo finance and fed into the LSTM model where it is trained and tested and even predicting the future price for given period of days. For cryptocurrency data is retrieved from twelve data website in json format and converted into dataframe and fed into the LSTM model. The future price for upcoming hour is also predicted. The website was developed using python Streamlit framework for stock and cryptocurrency prediction. From this project it is concluded that the prediction of stock and cryptocurrency prices using RNN LSTM model provides best accuracy than other models. Thus, it provides a best guidance for investors to buys and sell stocks and cryptocurrencies.

## Future work

Providing a user interface walkthrough guide and improving the user interface. Enabling user interaction using voice-based commands for blind and visually impaired people. Providing a recommendation system for investors, thus providing end to end guidance and maximizing their profit.

# REFERENCES

[1] Bhawna Panwar, Gaurav Dhuriya, Prashant Johri, Sudeept Singh Yadav, Nitin Gaur "Stock Market Prediction Using Linear Regression and SVM", 2021.

[2] Karunya Rathan, Somarouthu Venkat Sai, Tubati Sai Manikanta "Crypto-Currency price prediction using Decision Tree and Regression techniques",2019.

[3] Bohdan M. Pavlyshenko "Bitcoin Price Predictive Modeling Using Expert Correction",2019.

[4] Rezaul Karim, Md Khorshed Alam, Md Rezaul Hossain "Stock Market Analysis Using Linear Regression and Decision Tree Regression",2021.

[5] Gourav Bathla "Stock Price prediction using LSTM and SVR", 2020.

**[6]** Yash Indulkar "Time Series Analysis of Cryptocurrencies Using Deep Learning & Fbprophet",2021**.**

# APPENDIX A

## A.1 SOURCE CODE

```
import  webbrowser
import numpy as np
import pandas as pd
import  matplotlib.pyplot  as  plt
import pandas_datareader as data
from keras.models import load_model
import streamlit as st
import tensorflow as tf
import math
from streamlit_option_menu import option_menu
from datetime import date, datetime, timedelta
def crypto():
    import requests, pandas as pd, numpy as np, matplotlib.pyplot as plt
    from sklearn.preprocessing import MinMaxScaler
    from tensorflow.keras.layers import Dense, Dropout, LSTM
    from tensorflow.keras.models import Sequential
    from keras.models import load_model
    import streamlit as st
    api_key = '2f0fdd5d49254c06b7dc54cc45a4b628'
    interval = '5min'
    order = 'asc'
    start_date =  datetime.strftime(datetime.now() - timedelta(20),'%Y-%m-%d
    %H:%M:%S')
```

```
end_date = datetime.strftime(datetime.now() - timedelta(3),'%Y-%m-%d %H:%M:%S')
st.title('Cryptocurrency Price Prediction and Forecasting')
user_input = st.text_input('Enter Cryptocurrency Ticker','BTC/USD')
if len(user_input)==0:
    st.warning("Enter cryptocurrency symbol")
    return
try:
    api_url = f'https://api.twelvedata.com/time_series?symbol={user_input}
    &start_date={start_date}&end_date={end_date}&interval={interval}
    &order={order}&apikey={api_key}'
    data = requests.get(api_url).json()
    data_final = pd.DataFrame(data['values'])
except:
    st.warning("Invalid Cryptocurrency Symbol")
    return
if len(data_final.index)<6048:
    st.warning("OOPS! Insufficient data to predict,Cryptocurrency should be minimun
21 days old")
    return
#RANGE CONVERSION
scaler = MinMaxScaler(feature_range=(0,1))
scaled_data = scaler.fit_transform(data_final['close'].values.reshape(-1,1))
time_intervals_to_train = 24
prediction_interval = 12
x_train = []
y_train = []
```

```
    #data prepration
    for i in range(time_intervals_to_train, len(scaled_data) - prediction_interval):
        x_train.append(scaled_data[i - time_intervals_to_train: i, 0])
        y_train.append(scaled_data[i + prediction_interval, 0])
    x_train = np.array(x_train)
    y_train = np.array(y_train)
    x_train = np.reshape(x_train, (x_train.shape[0], x_train.shape[1],1))
#Load crypto model
    model = load_model('Cryptoc_predictor.h5')
    test_start =datetime.strftime(datetime.now() - timedelta(3),'%Y-%m-%d %H:%M:%S')
    now= datetime.now()
    test_end = now.strftime("%Y-%m-%d %H:%M:%S")
    test_api_url = f'https://api.twelvedata.com/time_series?symbol={user_input}
    &start_date={test_start}&end_date={test_end}&interval={interval}
    &order={order}&apikey={api_key}'
    test_data = requests.get(test_api_url).json()
    test_data_final = pd.DataFrame(test_data['values'])
    #test data preparation
    bitcoin_prices =pd.to_numeric(test_data_final['close'], errors ='coerce').values
    test_inputs = test_data_final['close'].values
    test_inputs = test_inputs.reshape(-1,1)
    model_inputs = scaler.fit_transform(test_inputs)
    x_test= []
    for x in range(time_intervals_to_train, len(model_inputs)):
        x_test.append(model_inputs[x - time_intervals_to_train: x, 0])
```

```
    x_test = np.array(x_test)
    x_test = np.reshape(x_test, (x_test.shape[0], x_test.shape[1], 1))
    prediction_prices = model.predict(x_test)
    prediction_prices = scaler.inverse_transform(prediction_prices)
     #model acurracy
    st.subheader('Model\'s Acurracy')
    fig = plt.figure(figsize =(12,4))
    plt.plot(bitcoin_prices, label ='Actual Prices')
    plt.plot(prediction_prices, label = 'Predicted Prices')
    plt.xlabel('5min Time interval')
    plt.ylabel('Price')
    plt.legend()
    st.pyplot(fig)
#forecast price for the next hour
    last_data = model_inputs[len(model_inputs) - time_intervals_to_train:
   len(model_inputs), 0]
    last_data = np.array(last_data)
    last_data = np.reshape(last_data, (1, last_data.shape[0], 1))
    prediction = model.predict(last_data)
    prediction = scaler.inverse_transform(prediction)
    st.subheader('Forecasted price for next hour')
    st.write("Price: ",prediction[0][0])
def stocks():
   start= datetime.strftime(datetime.now() - timedelta(3650),'%Y-%m-%d %H:%M:%S')
   today= date.today()
   end= today.strftime("%Y-%m-%d")
```

```
user_input= st.text_input('Enter Stock Ticker','AAPL')
if len(user_input)==0:
      st.warning("Enter stock symbol")
      return
finance= 'yahoo'
try:
    df=data.DataReader(user_input,finance,start,end)
except:
    st.warning("Invalid Stock Name")
    return
if len(df.index)<365:
       st.warning("OOPS! Insufficient data to predict,Stock should be minimun 1 year old")
    return
#describing data
st.subheader('Data Statistics Available')
st.write(df.describe())
#visulisations
st.subheader('Closing Price vs Time chart')
fig = plt.figure(figsize =(12,4))
plt.plot(df.Close)
plt.xlabel('Time  (years)')
plt.ylabel('Closing price')
st.pyplot(fig)
st.subheader('Closing Price vs Time chart with 100MA')
\
```

```
    ma100 = df.Close.rolling(100).mean()
    fig = plt.figure(figsize =(12,4))
    plt.plot(ma100)
    plt.plot(df.Close)
    plt.xlabel('Time (years)')
    plt.ylabel('Closing price')
    st.pyplot(fig)
    st.subheader('Closing Price vs Time chart with 100MA & 200MA')
    ma100 = df.Close.rolling(100).mean()
    ma200 = df.Close.rolling(200).mean()
    fig = plt.figure(figsize =(12,4))
    plt.plot(ma100)
    plt.plot(ma200)
    plt.plot(df.Close)
    plt.xlabel('Time (years)')
    plt.ylabel('Closing price')
    st.pyplot(fig)
 # end of streamlit_L
    df1=df.reset_index()['Close']
 #minmaxscaler range
    from sklearn.preprocessing import MinMaxScaler
    scaler=MinMaxScaler(feature_range=(0,1))
    df1=scaler.fit_transform(np.array(df1).reshape(-1,1))
    training_size=int(len(df1)*0.65)
 #splitting dataset into train and test split
```

```
    test_size=len(df1)-training_size
    train_data,test_data=df1[0:training_size,:],df1[training_size:len(df1),:1]


# convert an array of values into a dataset matrix
# convert an array of values into a dataset matrix
    def create_dataset(dataset, time_step=1):
        dataX, dataY = [], []
        for i in range(len(dataset)-time_step-1):
            a = dataset[i:(i+time_step), 0] ###i=0, 0,1,2,3-----99 100
            dataX.append(a)
            dataY.append(dataset[i + time_step, 0])
        return np.array(dataX), np.array(dataY)
# reshape into X=t,t+1,t+2,t+3 and Y=t+4
    time_step = 100
    X_train, y_train = create_dataset(train_data, time_step)
    X_test, ytest = create_dataset(test_data, time_step)
#Load stock model
    model = load_model('Stockp_predictor.h5')
### Lets Do the prediction and check performance metrics
    train_predict=model.predict(X_train)
    test_predict=model.predict(X_test)
##Transform back to original form
    train_predict=scaler.inverse_transform(train_predict)
    test_predict=scaler.inverse_transform(test_predict)
## Calculate RMSE performance metrics
```

```
    import math
    from sklearn.metrics import mean_squared_error
    math.sqrt(mean_squared_error(y_train,train_predict))
## Test Data RMSE
    math.sqrt(mean_squared_error(ytest,test_predict))


## Plotting
# shift train predictions for plotting
    look_back=100
    trainPredictPlot = np.empty_like(df1)
    trainPredictPlot[:, :] = np.nan
    trainPredictPlot[look_back:len(train_predict)+look_back, :] = train_predict
# shift test predictions for plotting
    testPredictPlot = np.empty_like(df1)
    testPredictPlot[:, :] = np.nan
    testPredictPlot[len(train_predict)+(look_back*2)+1:len(df1)-1, :] = test_predict
# plot baseline and predictions
    st.subheader('Model\'s Acurracy')
    fig = plt.figure(figsize =(12,4))
    plt.plot(scaler.inverse_transform(df1))
    plt.plot(trainPredictPlot)
    plt.plot(testPredictPlot)
    plt.xlabel('Days')
    plt.ylabel('Closing price')
    st.pyplot(fig)
```

```
    x_input=test_data[len(test_data)-100:].reshape(1,-1)
    temp_input=list(x_input)
    temp_input=temp_input[0].tolist()
# demonstrate prediction for next 30 days
    from numpy  import  array
    lst_output=[]
    n_steps=100
    i=0
    while(i<30):
        if(len(temp_input)>100):
        #print(temp_input)
            x_input=np.array(temp_input[1:])
            print("{} day input {}".format(i,x_input))
            x_input=x_input.reshape(1,-1)
            x_input = x_input.reshape((1, n_steps, 1))
        #print(x_input)
            yhat = model.predict(x_input, verbose=0)
            print("{} day output {}".format(i,yhat))
            temp_input.extend(yhat[0].tolist())
            temp_input=temp_input[1:]
        #print(temp_input)
            lst_output.extend(yhat.tolist())
            i=i+1
        else:
             x_input = x_input.reshape((1, n_steps,1))
```

```
            yhat = model.predict(x_input, verbose=0)
            print(yhat[0])
            temp_input.extend(yhat[0].tolist())
            print(len(temp_input))
            lst_output.extend(yhat.tolist())
            i=i+1
    day_new=np.arange(1,101)
    day_pred=np.arange(101,131)
    st.subheader('Prediction for next 30 days')
    fig = plt.figure(figsize =(12,4))
    plt.plot(day_new,scaler.inverse_transform(df1[len(df1)-100:]))
    plt.plot(day_pred,scaler.inverse_transform(lst_output))
    plt.xlabel('Days')
    plt.ylabel('Closing price')
    st.pyplot(fig)
    st.subheader('Forecasted Stock prices')
    fig = plt.figure(figsize =(12,4))
    df3=df1.tolist()
    df3.extend(lst_output)
    df3=scaler.inverse_transform(df3).tolist()
    plt.plot(df3)
    plt.xlabel('Days')
    plt.ylabel('Closing price')
    st.pyplot(fig)
#navigation bar
```

```
selected = option_menu(
    menu_title=None,
    options= ["Stock", "Cryptocurrency"],
    icons=["Stock","Cryptocurrency"],
    menu_icon="cast",
    default_index= 0,
    orientation = "horizontal",
)

if selected =="Stock":
    st.title('Stock Price Prediction and Forecasting')
    stocks()
if selected =="Cryptocurrency":
    crypto()
```

**Stocks Model:** For Stockp_predictor.h5

```
model=Sequential()
model.add(LSTM(50,return_sequences=True,input_shape=(100,1)))
model.add(LSTM(50,return_sequences=True))
model.add(LSTM(50))
model.add(Dense(1))
model.compile(loss='mean_squared_error',optimizer='adam')
model.fit(X_train,y_train,validation_data=(X_test,ytest),epochs=100,batch_size=64,verbo
    se=1)
model.save('Stockp_predictor.h5')
```

**Cryptocurrency Model:** For Cryptocurrency_predictor.h5

```
model = Sequential()
model.add(LSTM(128, return_sequences= True, input_shape = (x_train.shape[1], 1),
    activation='relu'))
model.add(Dropout(0.4))
model.add(LSTM(64, return_sequences= True, activation ='relu'))
model.add(Dropout(0.3))
model.add(LSTM(32, activation ='relu'))
model.add(Dropout(0.2))
model.add(Dense(1, activation ='sigmoid'))
model.compile(loss='mean_squared_error', optimizer='adam', metrics= ['accuracy'])
model.fit(x_train, y_train, epochs=10, batch_size=64)
model.save('Cryptocurrency_predictor.h5')
```

# A.2 INSTALLATION PROCEDURE

How to Install Anaconda on Windows?

Anaconda is an open-source software that contains Jupyter, spyder, etc that are used for large data processing, data analytics, heavy scientific computing. Anaconda works for R and python programming language. Spyder(sub-application of Anaconda) is used for python. Opencv for python will work in spyder. Package versions are managed by the package management system called conda.

To begin working with Anaconda, one must get it installed first. Follow the below instructions to Download and install Anaconda on your system:

Download and install Anaconda:

Head over to anaconda.com and install the latest version of Anaconda. Make sure to download the "Python 3.7 Version" for the appropriate architecture.

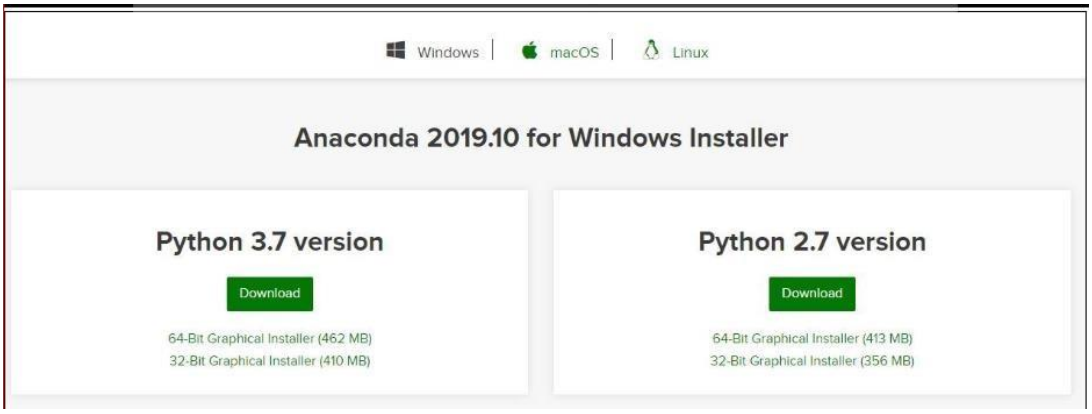



**Fig A.2.1: Downloading anaconda on the official website.**

Begin with the installation process:

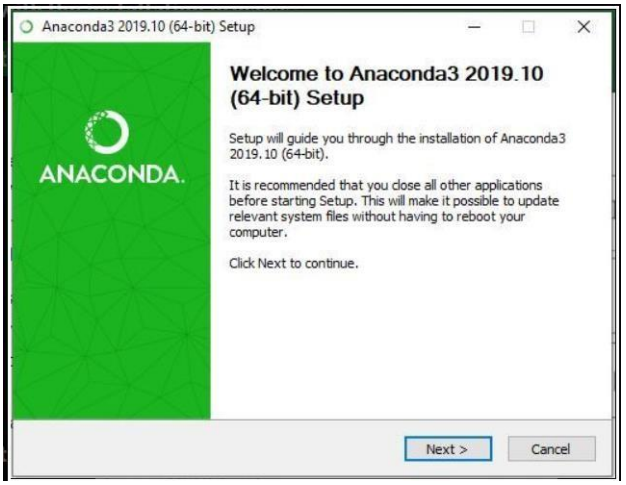



**Fig A.2.2: Getting Started**

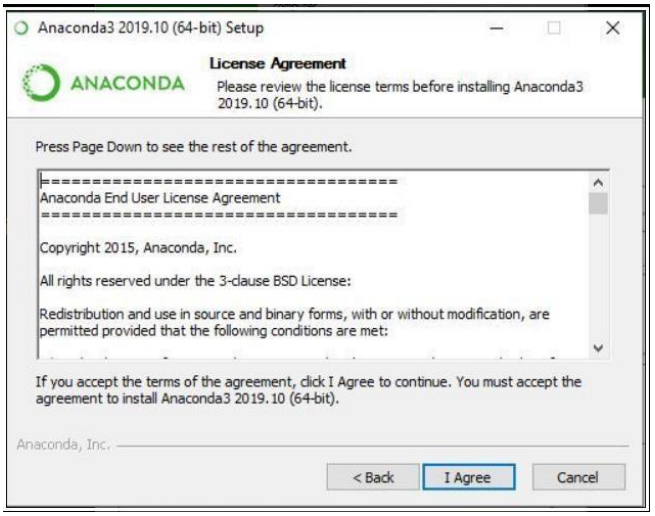



**Fig A.2.3: Getting through the License Agreement**

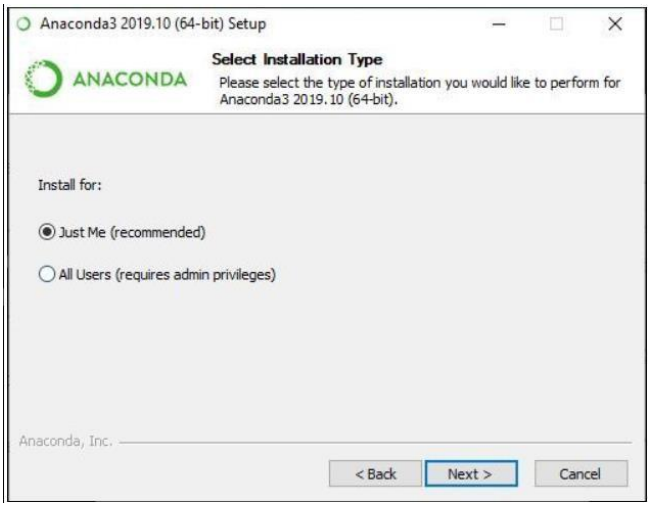



**Fig A.2.4: Select Installation Type: Select Just Me if you want the software to be used by a single User.**

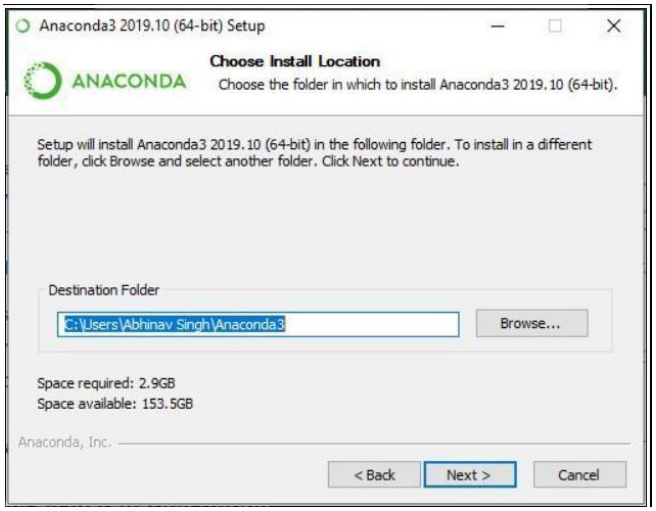



**Fig A.2.5: Choose Installation Location**

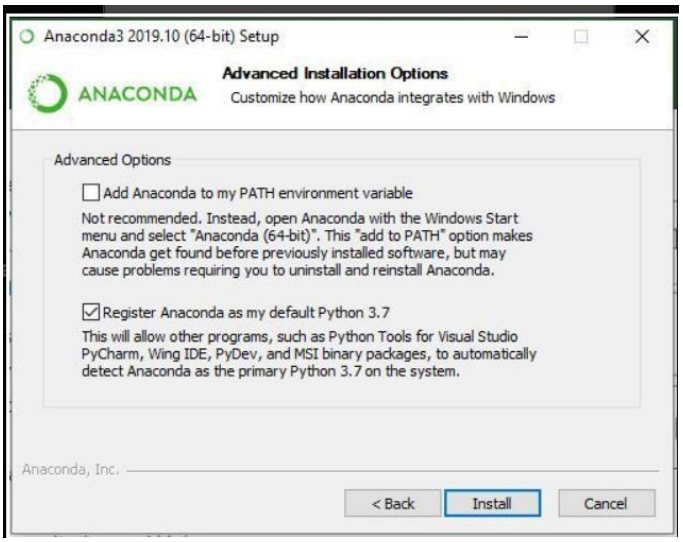



**Fig A.2.6: Advanced Installation Option.**

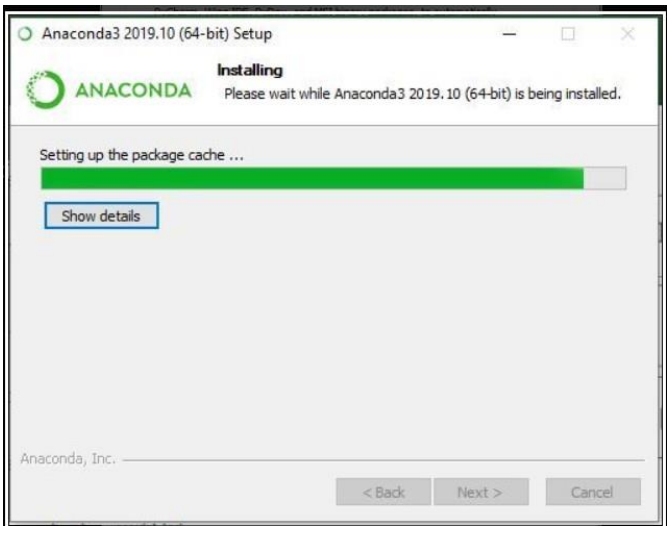



**Fig A.2.7: Getting through the Installation Process.**


**Fig A.2.8: Recommendation to Install Pycharm.**


**Fig A.2.9: Finishing up the Installation.**

Working With Anaconda:

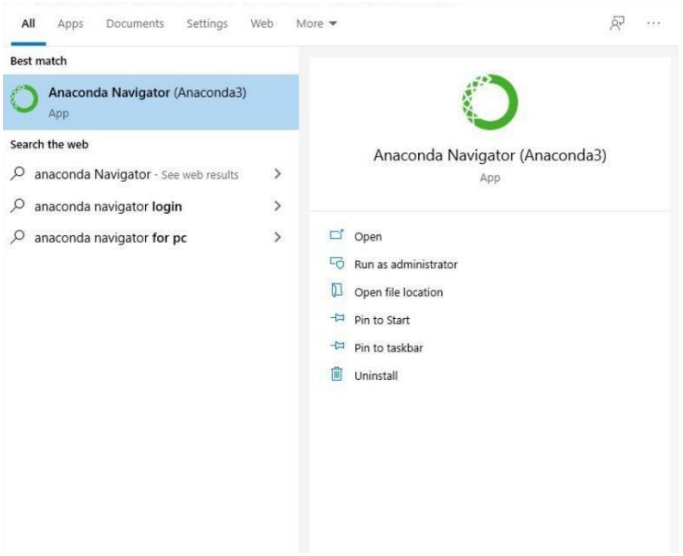



**Fig A.2.10: Searching navigator.**

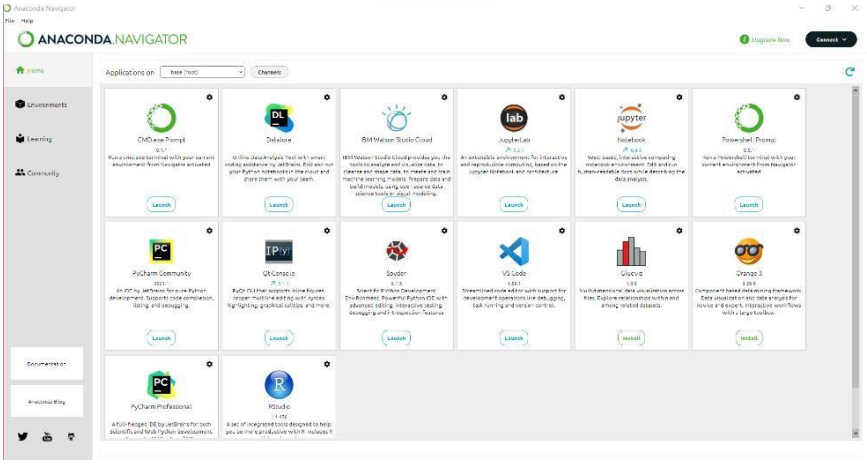



**Fig A.2.11: Main menu of navigator.**