# STOCK PRICE PREDICTION

# **PHASE :3**

**PROCESS AND DATA;**

1. **Data Collection:** Gather historical stock prices, trading volumes, and relevant financial data. Sources could include financial databases, APIs, or web scraping tools.
2. **Data Pre-processing**: Clean the data, handle missing values, and perform feature engineering. This step might involve normalization, scaling, or transforming the data to make it suitable for the chosen model.
3. **Feature Selection:** Choose relevant features that might affect stock prices, such as historical prices, trading volumes, news sentiment, economic indicators, and company-specific information.
4. **Model Selection:** Select an appropriate machine learning algorithm such as linear regression, decision trees, random forests, or deep learning models like recurrent neural networks (RNNs) or long short-term memory networks (LSTMs).
5. **Training the Model:** Use a portion of the data to train the model, adjusting parameters and hyper parameters to optimize performance.
6. **Model Evaluation:** Assess the model's performance using metrics like Mean Absolute Error (MAE), Mean Squared Error (MSE), or Root Mean Squared Error (RMSE) on a validation dataset.
7. **Testing the Model:** Apply the trained model to a separate test dataset to evaluate its predictive power.
8. **Iterate and Refine:** Fine-tune the model by iterating on the pre-processing steps, feature selection, and model selection to improve predictive accuracy.

It’s important to note that stock market prediction is inherently complex and subject to a variety of external factors, including market sentiment, geopolitical events, and economic indicators, which might not be fully captured by historical data alone. Therefore, it's essential to exercise caution and understand the limitations of any predictive model.

**USED SENSORS:**

Predicting stock prices using sensor data is an emerging application of data science, combining the fields of finance, data analysis, and sensor technology. By leveraging sensor data, such as market sentiment, economic indicators, or even physical sensors monitoring relevant parameters, data scientists can develop more nuanced and accurate models for stock price forecasting.

To achieve this, the process typically involves collecting and analysing a diverse range of data, applying machine learning algorithms, and employing techniques like time series analysis to identify patterns and correlations. However, it's important to note that stock market prediction, despite technological advancements, remains inherently challenging due to its dynamic nature and the influence of various external factors.

**PYTHON SCRIPT;**

import serial

import time

import requests

import random

# Define your Thing Speak API key and channel URL

spikey = "G1RSE98QHVUFT626"

url = f"https://api.thingspeak.com/update?api\_key={api\_key}"

# Initialize the serial connection to Arduino

Arduino port = '/dev/ttyACM0' # Update with your Arduino's port

seer = serial. Serial(Arduino port, 9600)

# Function to read data from Arduino

def. read\_arduino\_data():

data = sertraline().decode().strip()

return data

# Simulated sensor data (replace with actual sensor readings)

def. read\_sensor\_data():

temperature = random. Uniform(20.0, 30.0)

water level = random. Uniform(0, 100)

return temperature, water level

**while True:**

**try**:

# Read data from Arduino

Arduino data = read\_arduino\_data()

# Read sensor data

temperature, water level = read\_sensor\_data()

# Prepare data to send to Thing Speak

data = {

'field1': temperature,

'field2': water level,

'field3': Arduino data

}

# Send data to Thing Speak

response = requests. Post(url, data=data)

print("Data Sent to Thing Speak")

except Exception as e:

print(error: {e}")

# Set the data upload interval

time. Sleep(300) # Upload data every 5 minutes (adjust as needed)