

# Real-Time Stock Market Trend Predictor & Visual Analytics Dashboard

```
# Real-Time Stock Market Trend Predictor & Visual Analytics Dashboard

# Import necessary libraries
import yfinance as yf
import pandas as pd
import numpy as np
import plotly.graph_objs as go
from prophet import Prophet
import streamlit as st

# Streamlit App Title
st.title('■ Real-Time Stock Market Trend Predictor & Dashboard')

# Input for Stock Symbol
symbol = st.text_input('Enter Stock Symbol (e.g. AAPL, TSLA, MSFT):', 'AAPL')

# Fetch live data
data = yf.download(symbol, period='6mo', interval='1d')

# Data Preprocessing
data.reset_index(inplace=True)
data['Date'] = pd.to_datetime(data['Date'])

# Prophet Model Preparation
df = data[['Date', 'Close']].rename(columns={'Date': 'ds', 'Close': 'y'})
model = Prophet(daily_seasonality=True)
model.fit(df)

# Forecast next 30 days
future = model.make_future_dataframe(periods=30)
forecast = model.predict(future)

# Visualization
fig = go.Figure()
fig.add_trace(go.Scatter(x=data['Date'], y=data['Close'], name='Actual Price'))
fig.add_trace(go.Scatter(x=forecast['ds'], y=forecast['yhat'], name='Predicted Price'))

st.plotly_chart(fig)

# Buy/Sell/Hold Signal
latest_price = data['Close'].iloc[-1]
predicted_price = forecast['yhat'].iloc[-1]

if predicted_price > latest_price * 1.02:
    recommendation = 'Buy ■'
elif predicted_price < latest_price * 0.98:
    recommendation = 'Sell ■'
else:
    recommendation = 'Hold ■■'

st.subheader('Recommendation: ')
st.write(recommendation)

# Additional Insights
st.write('### Summary Statistics')
st.dataframe(data.describe())
```