Smart Internz

Time Series Analysis For Bitcoin Price Prediction Using Prophet

EXTERNSHIP PROJECT: VIT BHOPAL CAMPUS

TITLE : Time Series Analysis For Bitcoin Price Prediction Using Prophet

GROUP : AAVUGARI VIVEK VARDHAN (21BAI10029)
NAGULA RACHIT (21BAI10260)
NIHAL SANJAY JASTI (21BCE11039)
ANUDEEP VENIGALLA (21BAI10466)

Table of Contents

1. INTRODUCTION

1.1 Overview:

This project focuses on utilizing the FbProphet model to predict the price of Bitcoin, the world's most valuable cryptocurrency. Bitcoin, created in January 2009, offers a unique opportunity for price forecasting due to its high volatility, which surpasses that of traditional currencies. With over 40 exchanges worldwide accepting more than 30 different currencies, Bitcoin has gained popularity among investors for its anonymity and transparency within the system.

The objective of this project is to develop a reliable prediction system for Bitcoin's price using the FbProphet model. By analyzing various factors that influence Bitcoin's price, we aim to define the future price trend of this cryptocurrency. Recognizing the highly volatile nature of the crypto market, our project aims to leverage FbProphet's capabilities to provide accurate and actionable predictions for Bitcoin's price. By considering historical data and relevant market indicators, we aim to generate forecasts that can assist investors and stakeholders in making informed decisions regarding Bitcoin.

1.2 Purpose:

The purpose of this project is to outline the utilization of FbProphet for Bitcoin price prediction. It aims to identify trends, seasonality, and outliers in Bitcoin prices while analyzing the impact of external factors. Additionally, the project focuses on generating accurate predictions and visualizing the data to aid decision-making in the cryptocurrency market.

Identifying Trends and Seasonality: This project utilizes the FbProphet model to identify long-term trends, weekly and daily seasonality, and holiday effects within Bitcoin prices. By analyzing historical data, the project aims to uncover patterns and recurring trends that can contribute to more accurate price forecasting.

Visualizing Data: The project leverages FbProphet's visualization capabilities to create informative visual representations of Bitcoin price data. These visualizations aid in understanding the data, identifying trends, and presenting predictions in a clear and concise manner.

Making Predictions: FbProphet is leveraged to make predictions about future Bitcoin prices. By analyzing past price movements and considering relevant market indicators, the project aims to generate forecasts that assist in decision-making for investors and stakeholders in the cryptocurrency market.

2. LITERATURE SURVEY

2.1 Existing Problem:

The existing problem in the field of Bitcoin price prediction lies in the high volatility and non-linear nature of cryptocurrency markets. Bitcoin prices are influenced by various factors such as market demand, regulatory developments, technological advancements, and investor sentiment. These factors create challenges for accurately predicting future price movements, leading to potential financial risks for investors and traders. this strategy is not without its problems and difficulties.

2.2 References:

High Volatility: Bitcoin prices are known for their high volatility, characterized by large price fluctuations within short periods. This volatility makes it challenging to accurately predict future price movements, as small changes in market conditions can lead to significant price swings. Addressing this problem requires robust modeling techniques that can capture and account for the inherent volatility in Bitcoin price data.

Non-Linearity: Bitcoin price data often exhibits non-linear patterns and relationships. Traditional linear regression models may not be able to capture these complex dynamics effectively. Non-linear modeling approaches, such as machine learning algorithms or time series methods like Prophet, are needed to capture the non-linear nature of Bitcoin price data and improve prediction accuracy.

Limited Historical Data: Bitcoin is a relatively new asset, and historical price data is limited compared to traditional financial assets. The availability of a smaller dataset can pose challenges in building accurate forecasting models. Dealing with limited historical data requires careful selection of relevant features, incorporating external data sources, or utilizing techniques like transfer learning to leverage information from related financial assets.

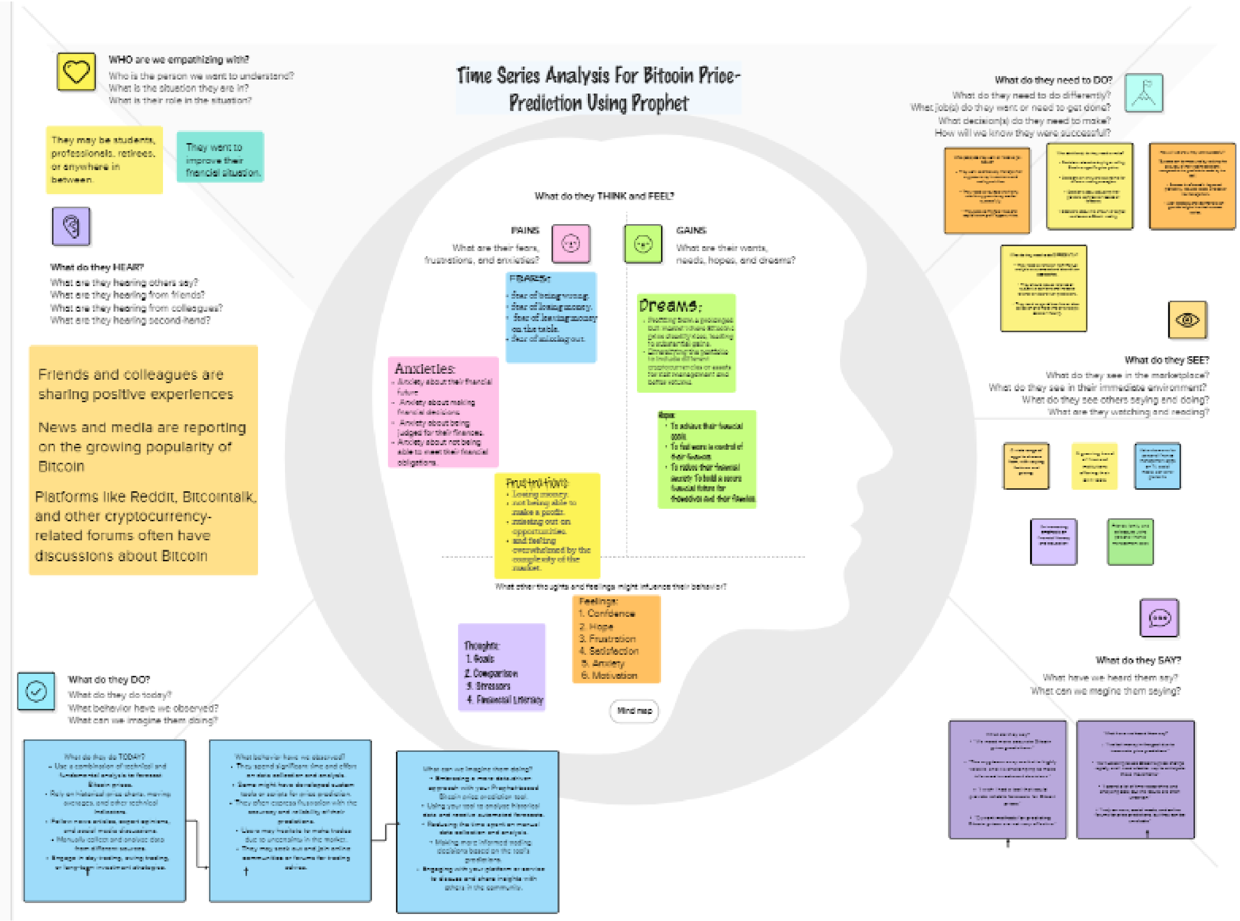
Since cryptocurrency prices can be highly volatile, it's essential to update your model regularly to capture new trends and patterns.

2.3 Problem Statement Definition:

Bitcoin being the most prominent and widely traded among them. However, the extreme volatility and non-linear nature of Bitcoin prices present a significant challenge for accurate prediction. The problem is to develop an accurate and reliable time series forecasting model for predicting Bitcoin's future prices using the FbProphet algorithm.

3. IDEATION AND PROPOSED SOLUTION

3.1 Empathy Map Canvas:



Time Series Analysis For Bitcoin Price-Prediction Using Prophet

WHO are we empathizing with?
Who is the person we want to understand?
What is the situation they are in?
What is their role in the situation?

They may be students, professionals, retirees, or anywhere in between.

They want to improve their financial situation.

What do they HEAR?
What are they hearing others say?
What are they hearing from friends?
What are they hearing from colleagues?
What are they hearing second-hand?

Friends and colleagues are sharing positive experiences

News and media are reporting on the growing popularity of Bitcoin

Platforms like Reddit, Bitcointalk, and other cryptocurrency-related forums often have discussions about Bitcoin

What do they THINK and FEEL?

PAINS
What are their fears, frustrations, and anxieties?

GAINS
What are their wants, needs, hopes, and dreams?

FEARS:
- fear of being wrong.
- fear of losing money.
- fear of leaving money on the table.
- fear of missing out.

Anxieties:
- Anxiety about their financial future
- Anxiety about making financial decisions
- Anxiety about being judged for their finances.
- Anxiety about not being able to meet their financial obligations.

Frustrations:
- Losing money.
- not being able to make a profit.
- missing out on opportunities.
- and feeling overwhelmed by the complexity of the market.

Dreams:
- Predicting a bull market where Bitcoin's price steadily rises, leading to substantial gains.
- Diversifying the portfolio to include different cryptocurrencies or assets for risk management and better returns.

Hope:
- To achieve their financial goals.
- To feel more in control of their finances.
- To raise their financial security To build a secure financial future for themselves and their families.

What other thoughts and feelings might influence their behavior?

Thoughts:
1. Goals
2. Comparison
3. Stressors
4. Financial Literacy

Feelings:
1. Confidence
2. Hope
3. Frustration
4. Satisfaction
5. Anxiety
6. Motivation

Mind map

What do they need to DO?
What do they need to do differently?
What job(s) do they want or need to get done?
What decision(s) do they need to make?
How will we know they were successful?

What do they SEE?
What do they see in the marketplace?
What do they see in their immediate environment?
What do they see others saying and doing?
What are they watching and reading?

What do they SAY?
What have we heard them say?
What can we imagine them saying?

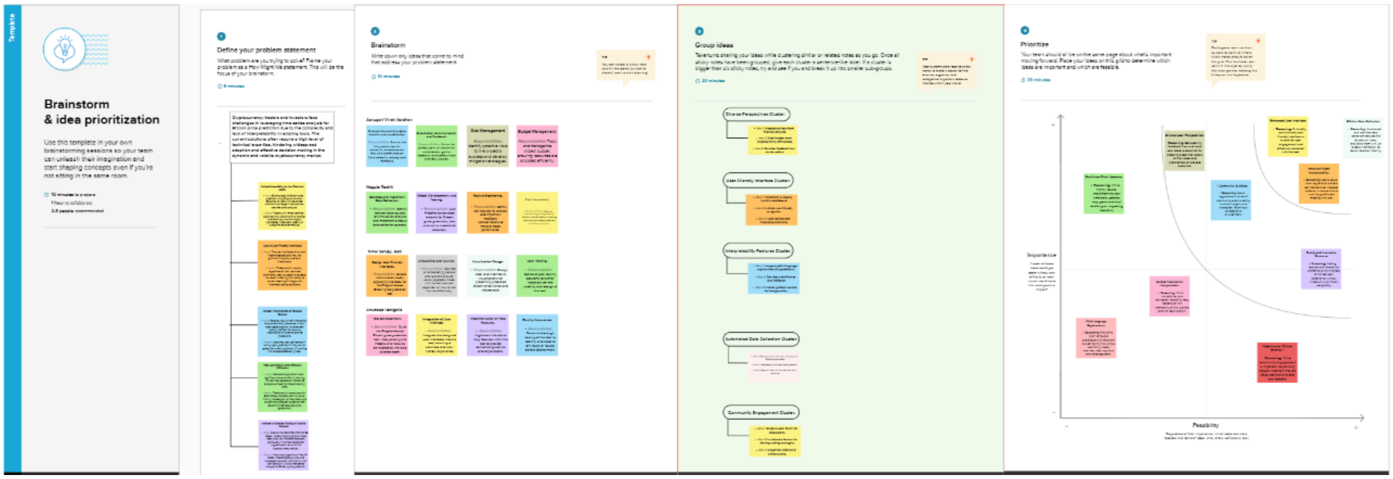
What do they DO?
What do they do today?
What behavior have we observed?
What can we imagine them doing?

What do they do TODAY?
- Use a combination of technical and fundamental analysis to forecast Bitcoin prices.
- Rely on historical price charts, moving averages, and other technical indicators.
- Follow news articles, expert opinions, and social media discussions.
- Manually collect and analyze data from different sources.
- Engage in day trading, swing trading, or long-term investment strategies.

What behavior have we observed?
- They spend significant time and effort on data collection and analysis.
- Some might have developed custom tools or scripts for price prediction.
- They often express frustration with the accuracy and reliability of their predictions.
- Users may hesitate to make trades due to uncertainty in the market.
- They may seek out and join online communities or forums for trading advice.

What can we imagine them doing?
- Embracing a more data-driven approach with your Prophet-based Bitcoin price prediction tool.
- Using your tool to analyze historical data and receive automated forecasts.
- Reducing the time spent on manual data collection and analysis.
- Making more informed trading decisions based on the tool's predictions.
- Engaging with your platform or service to discuss and share insights with others in the community.

3.2 Ideation and Brainstorming:



4. REQUIREMENT ANALYSIS

4.1 Functional Requirement

Python: Python is the primary programming language for this project. Ensure that Python is installed on your system. It is recommended to use the Python 3.x version.

Prophet Library: Install the Prophet library using the pip package manager. Prophet can be installed by running the command: pip install prophet. The Prophet library is developed by Facebook's Core Data Science team and provides an efficient implementation of the time series forecasting algorithm.

Data Retrieval: The project utilizes the Yahoo Finance library, which is an inbuilt library in Python, to retrieve the Bitcoin price data. This library provides convenient functions for accessing historical financial data from Yahoo Finance's API. You can install it using the command: **pip install yfinance**.

Streamlit: Streamlit is a popular Python library used for building interactive web applications and dashboards. It allows for easy integration of data analysis code with a user-friendly frontend. Install Streamlit using the command: **pip install streamlit.** Streamlit will be used to create a frontend interface to display the results and visualizations of the Bitcoin price prediction analysis.

Data Analysis and Visualization Libraries: Install essential Python libraries for data analysis and visualization. These may include Pandas, NumPy, Matplotlib, and Seaborn.

4.2 Non Functional Requirement

Since Prophet is a lightweight model, it can be run on most CPUs without requiring highend or specialized hardware. However, if the dataset is large or if there is a need for computationally intensive operations, having a more powerful CPU can help expedite the processing time. Additionally, having an adequate amount of RAM is crucial to ensure smooth execution, especially when working with larger datasets or running multiple models simultaneously. Sufficient storage space is also necessary to store the Bitcoin price data, intermediate results, and any additional datasets used for analysis.

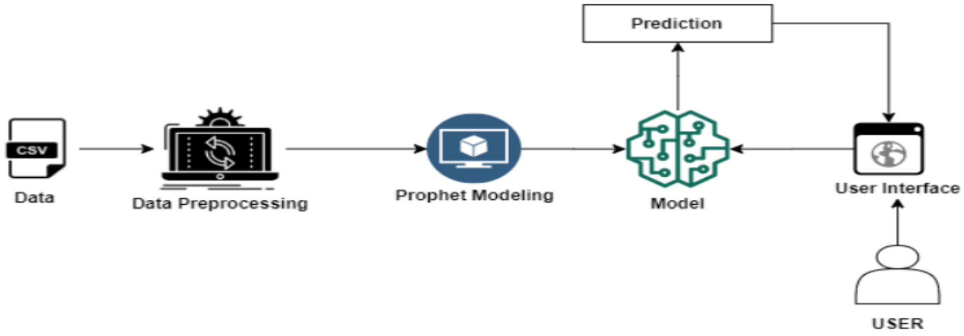
Note: The Functional and Non functional requirements mentioned above are general recommendations. The specific requirements may vary depending on the size of the dataset, complexity of the model, and available computing resources.

5. PROJECT DESIGN

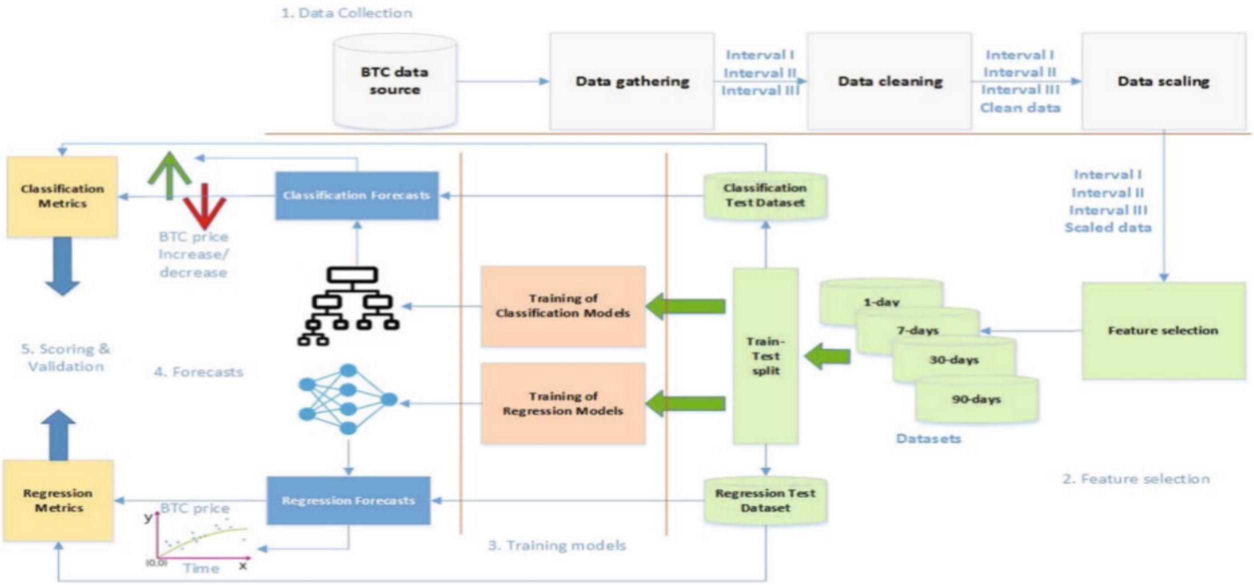
5.1 Data Flow Diagrams & User Stories

A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It shows how data enters and leaves the system, what changes the information, and where data is stored.

General Architecture:



Data Flow Diagram:



User Stories

User Type	Functional Requirement (EPIC)	User Story Number	User Story / Task	Acceptance Criteria	Priority	Release
Customer (Mobile user)	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	I can access my account / dashboard	High	Sprint -1
		USN-2	As a user, I will receive confirmation email once I have registered for the application	I can receive confirmation email & click confirm	High	Sprint -1
		USN-3	As a user, I can register for the Website through Gmail		Medium	Sprint -1
	Login	USN-4	As a user, I can log into the application by entering email & password		High	Sprint -1
	Dashboard	USN-5	As a user, I should be able to see a dashboard after login	See 2 Buttons View/ predict the price	High	Sprint -1
	Prediction	USN-6	As a user, I should be able to predict the price by selecting the date and getting output	Submit button	High	Sprint -2
	Security	USN-7	As a user, I want to run it securely		Medium	Sprint-3
	Deployment	USN-8	I Want the Website to publicly Visible and able to run	Through cloud based services like heroku etc.	High	Sprint -4
Customer Care Executive	–	–	—	——	———	———
Administrator	—	—-	—-	—-	—-	—-

5.2 Solution Architecture

Solution architecture is a complex process – with many sub-processes – that bridges the gap between business problems and technology solutions.

- **Solution:**

The user interacts with the UI (User Interface) to select the date as input. Selected Date input values are analyzed by the model which is integrated. Once the model is analyzed the input prediction is showcased on the UI. We will be Create or Collect the dataset based on this project, we will be creating an HTML File and building a web , for data processing we will build a Python Code.

- **Structure, Characteristics, Behavior:**

It consists of a dataset where Import necessary libraries, load Bitcoin price data, and prepare it for analysis. Use Facebook Prophet to model the time series data with appropriate characteristics (seasonality, holidays, etc.). Generate future price predictions and visualize the results. Create an HTML report to display the prediction charts, metrics, and insights, using libraries like Flask or Dash for web development.

- **Features, Development Phases and Solution Requirements:**

A) Features:

1) **Historical Bitcoin Price Data**: A dataset containing historical Bitcoin price data, timestamps and values.
2) **Notifications:** Send reminders for upcoming bitcoin and up to date prices.
3) **User Authentication:** Secure web page and updating every date with prices.
4) **Exogenous Variables:** Optional additional data like news sentiment, trading volumes, or other indicators that can improve the accuracy of predictions.

B) Development Phase:

1) **Data Collection:** Gather historical Bitcoin price data and any relevant variables.
2) **Data Preprocessing:** Clean, transform, and preprocess the data. Handle missing values and outliers.
3) **Model Building**: Implement the Prophet model in a programming language like Python (using libraries like Fbprophet). Configure the model with appropriate hyperparameters.
4) **Train,Testing And Evaluation**: Train the Prophet model using the training data. The model will automatically detect patterns and seasonality in the data. Use the testing dataset to evaluate the model's performance.
5) **Visualization and Reporting:** Create visualizations of the historical Bitcoin price data and the model's forecasts.
6) **Deployment:** We use the model for real-time predictions, deploy it to a cloud platform. And Ensure That it can handle new data as it becomes available.

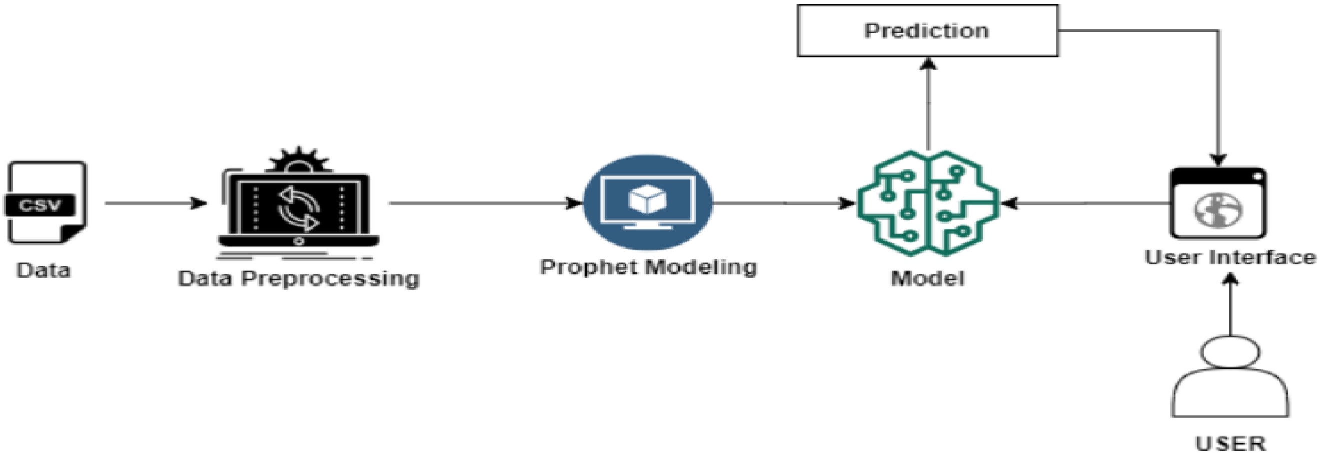
C) Requirements:

1) **Data**: Access to a reliable and up-to-date dataset of Bitcoin price data.
2) **Hardware/Cloud Resources**: Sufficient computing resources for model training and deployment if required.
3) **Monitoring**: If used in a production environment, implement a system to monitor model performance and retrain it periodically to adapt to changing market conditions.
4) **Security:** If handling sensitive data or deploying in a production environment, ensure data security and access control measures are in place.
5) **Data Backup**: Provide a backup Solution to prevent data loss.

- Solutions Delivered Via:

1) Prophet - Python
2) Machine Learning Algorithms - Random Forest, SVM
3) Data Analysis - Filter
4) Code will be done through - Jupyter Notebook or Google Colab
5) Model Integration - Flask

Example - Solution Architecture Diagram



6. PROJECT PLANNING AND SCHEDULING

6.1 Technical Architecture

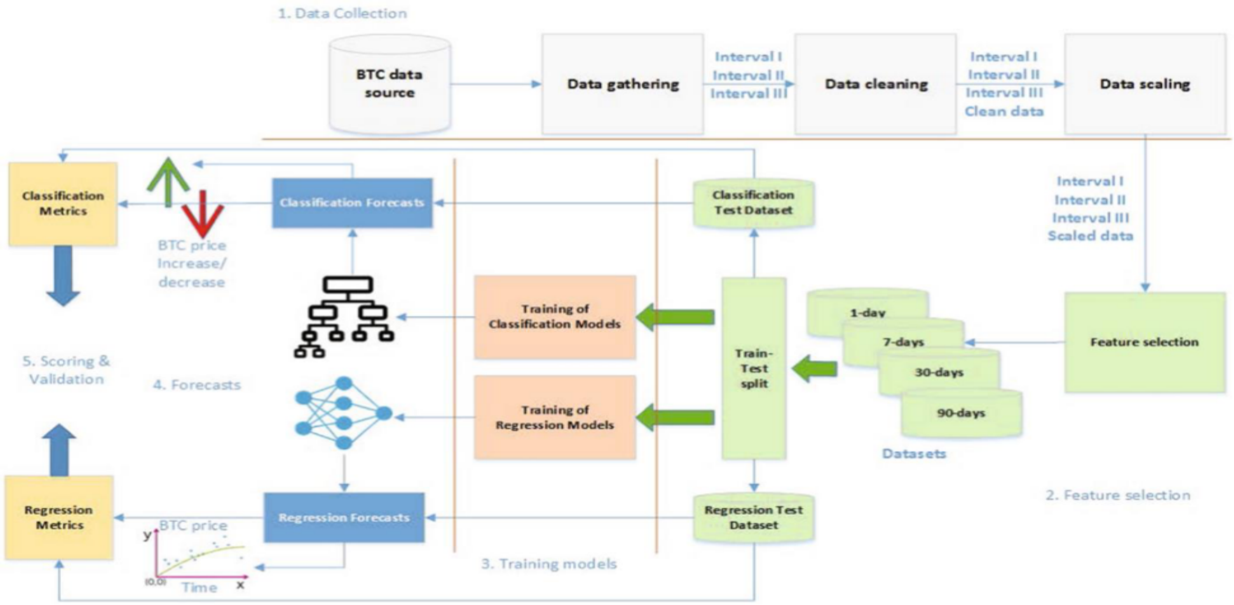


Table-1 Components and Technologies:

S.No	Component	Description	Technology
1	User Interface	How user interacts with application e.g. Web UI	HTML, CSS, prophet
2	Application Logic-1	Building a Python code for the price prediction	Python
3	Database	Data Type, Configurations etc.	From Kaggle (Dataset), SQL
4	File Storage	File storage requirements	SQLite etc.
5	Machine Learning Model	Purpose of Machine Learning Model	Bitcoin Price prediction
6	Infrastructure (Server / Cloud)	Application Deployment on Local System / Cloud	Streamlit, heroku etc.
7	Code File	A Python code for outputting values by using colab or jupyter	Colab, jupyter notebook etc.

Table-2 Application Characteristics:

S.No	Characteristics	Description	Technology
1	Open-Source Frameworks	Utilization of Open Source Frameworks in web	HTML, prophet (pkl), python
2	Security Implementations	Implementations of security measures and access Controls	Encryption etc.
3	Scalable Architecture	Design considerations for the code scalability	Using colab and Load balancing
4	Availability	Measures taken to ensure high availability of the web	Load Balancers etc.
5	Performance	considerations to ensure optimal performance and accuracy	Content Delivery Networks (CDN)

6.2 Sprint Planning & Estimation

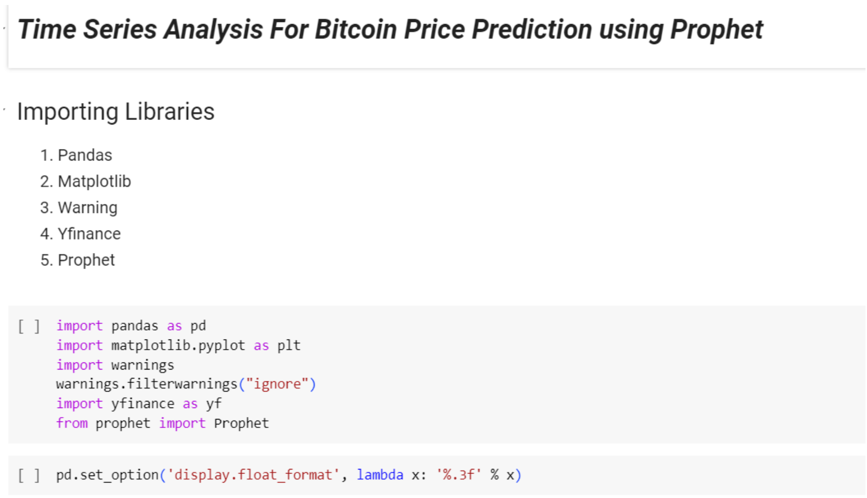
User Type	Functional Requirement (EPIC)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint -1	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	2	High	Vivek
Sprint -1		USN-2	As a user, I will receive confirmation email once I have registered for the application	1	High	Rachit
Sprint -2		USN-3	As a user, I can register for the Website through Gmail	2	Medium	Rachit
Sprint -2	Login	USN-4	As a user, I can log into the application by entering email & password	2	High	Nihal
Sprint -2	Dashboard	USN-5	As a user, I should be able to see a dashboard after login	2	High	Vivek

Sprint -3	Prediction	USN-6	As a user, I should be able to predict the price by selecting the date and getting output	3	High	Anudeep
Sprint -4	Security	USN-7	As a user, I want to run it securely	2	Medium	Nihal
Sprint- 4	Deployment	USN-8	I want the website to Publicly visible and able to run	3	High	Anudeep

6.3 Sprint Delivery Schedule

Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint -1	20	6 Days	24 Oct 2023	29 Oct 2023	20	13 Oct 2022
Sprint -2	20	6 Days	31 Oct 2023	05 Nov 2023		
Sprint -3	20	6 Days	07 Nov 2023	12 Nov 2023		
Sprint -4	20	6 Days	14 Nov 2023	22 Nov 2023		

7. CODING & SOLUTIONING (Explain the features added in the project along with code)



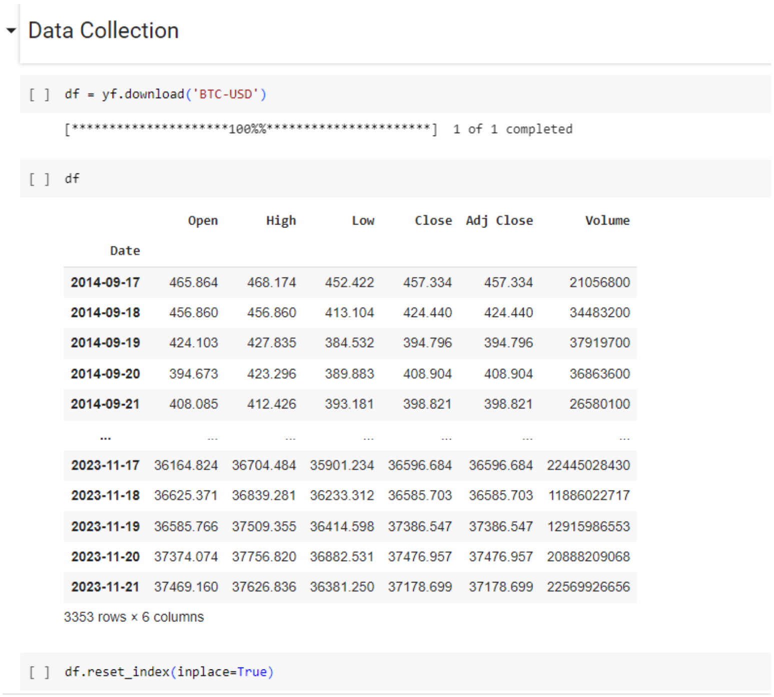
Time Series Analysis For Bitcoin Price Prediction using Prophet

Importing Libraries

1. Pandas
2. Matplotlib
3. Warning
4. Yfinance
5. Prophet

```
[ ] import pandas as pd
    import matplotlib.pyplot as plt
    import warnings
    warnings.filterwarnings("ignore")
    import yfinance as yf
    from prophet import Prophet
```

```
[ ] pd.set_option('display.float_format', lambda x: '%.3f' % x)
```



Data Collection

```
[ ] df = yf.download('BTC-USD')
```

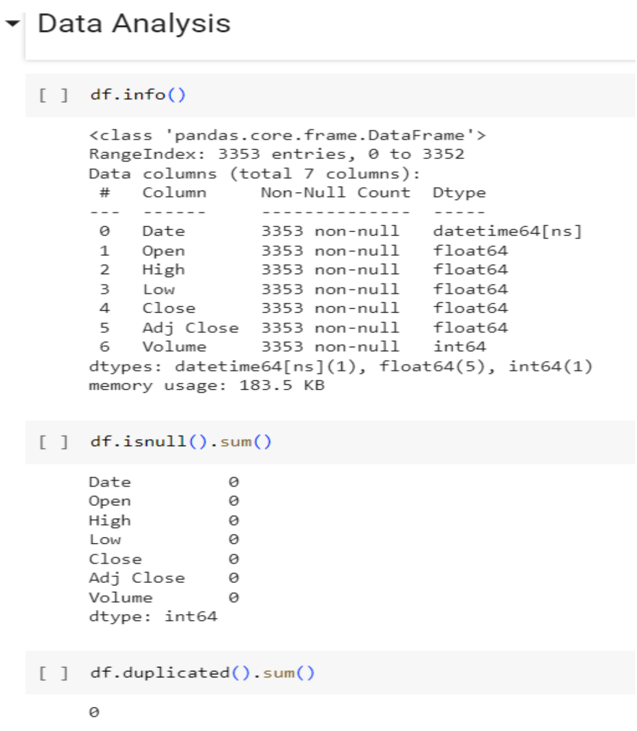
```
[*********************100%%**********************]  1 of 1 completed
```

```
[ ] df
```

	Open	High	Low	Close	Adj Close	Volume
Date						
2014-09-17	465.864	468.174	452.422	457.334	457.334	21056800
2014-09-18	456.860	456.860	413.104	424.440	424.440	34483200
2014-09-19	424.103	427.835	384.532	394.796	394.796	37919700
2014-09-20	394.673	423.296	389.883	408.904	408.904	36863600
2014-09-21	408.085	412.426	393.181	398.821	398.821	26580100
...	...	...	...	...	...	...
2023-11-17	36164.824	36704.484	35901.234	36596.684	36596.684	22445028430
2023-11-18	36625.371	36839.281	36233.312	36585.703	36585.703	11886022717
2023-11-19	36585.766	37509.355	36414.598	37386.547	37386.547	12915986553
2023-11-20	37374.074	37756.820	36882.531	37476.957	37476.957	20888209068
2023-11-21	37469.160	37626.836	36381.250	37178.699	37178.699	22569926656

3353 rows × 6 columns

```
[ ] df.reset_index(inplace=True)
```



Data Analysis

```
[ ]  df.info()
```

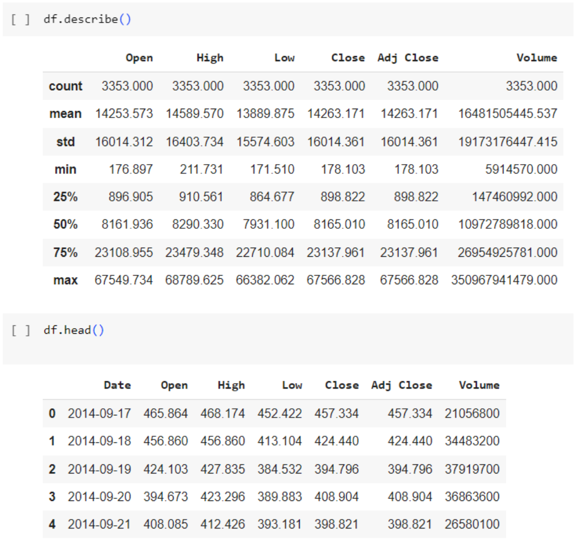
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3353 entries, 0 to 3352
Data columns (total 7 columns):
 #   Column     Non-Null Count  Dtype
---  ------     --------------  -----
 0   Date       3353 non-null   datetime64[ns]
 1   Open       3353 non-null   float64
 2   High       3353 non-null   float64
 3   Low        3353 non-null   float64
 4   Close      3353 non-null   float64
 5   Adj Close  3353 non-null   float64
 6   Volume     3353 non-null   int64
dtypes: datetime64[ns](1), float64(5), int64(1)
memory usage: 183.5 KB
```

```
[ ]  df.isnull().sum()
```

```
Date         0
Open         0
High         0
Low          0
Close        0
Adj Close    0
Volume       0
dtype: int64
```

```
[ ]  df.duplicated().sum()
```

```
0
```

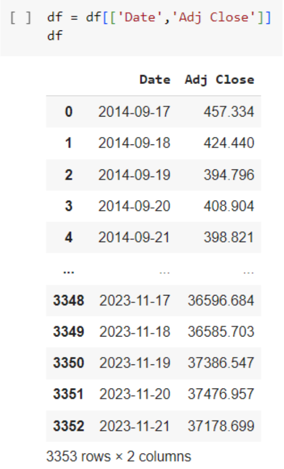


```
[ ]  df.describe()
```

	Open	High	Low	Close	Adj Close	Volume
count	3353.000	3353.000	3353.000	3353.000	3353.000	3353.000
mean	14253.573	14589.570	13889.875	14263.171	14263.171	16481505445.537
std	16014.312	16403.734	15574.603	16014.361	16014.361	19173176447.415
min	176.897	211.731	171.510	178.103	178.103	5914570.000
25%	896.905	910.561	864.677	898.822	898.822	147460992.000
50%	8161.936	8290.330	7931.100	8165.010	8165.010	10972789818.000
75%	23108.955	23479.348	22710.084	23137.961	23137.961	26954925781.000
max	67549.734	68789.625	66382.062	67566.828	67566.828	350967941479.000

```
[ ]  df.head()
```

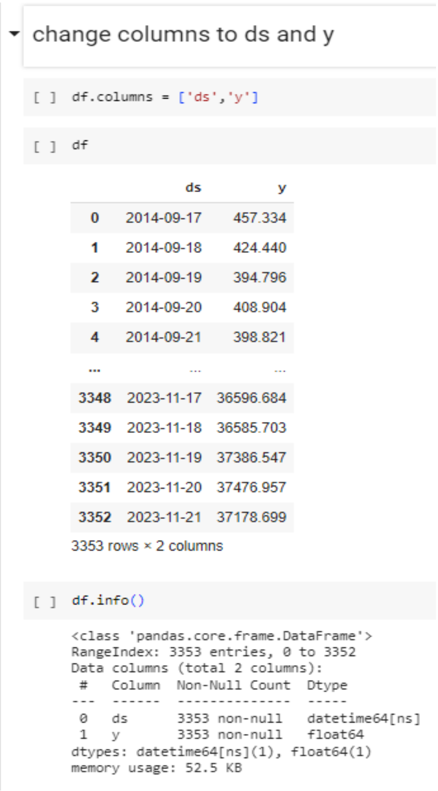
	Date	Open	High	Low	Close	Adj Close	Volume
0	2014-09-17	465.864	468.174	452.422	457.334	457.334	21056800
1	2014-09-18	456.860	456.860	413.104	424.440	424.440	34483200
2	2014-09-19	424.103	427.835	384.532	394.796	394.796	37919700
3	2014-09-20	394.673	423.296	389.883	408.904	408.904	36863600
4	2014-09-21	408.085	412.426	393.181	398.821	398.821	26580100



```
[ ]  df = df[['Date','Adj Close']]
     df
```

	Date	Adj Close
0	2014-09-17	457.334
1	2014-09-18	424.440
2	2014-09-19	394.796
3	2014-09-20	408.904
4	2014-09-21	398.821
...	...	...
3348	2023-11-17	36596.684
3349	2023-11-18	36585.703
3350	2023-11-19	37386.547
3351	2023-11-20	37476.957
3352	2023-11-21	37178.699

3353 rows × 2 columns



change columns to ds and y

```
[ ] df.columns = ['ds','y']
```

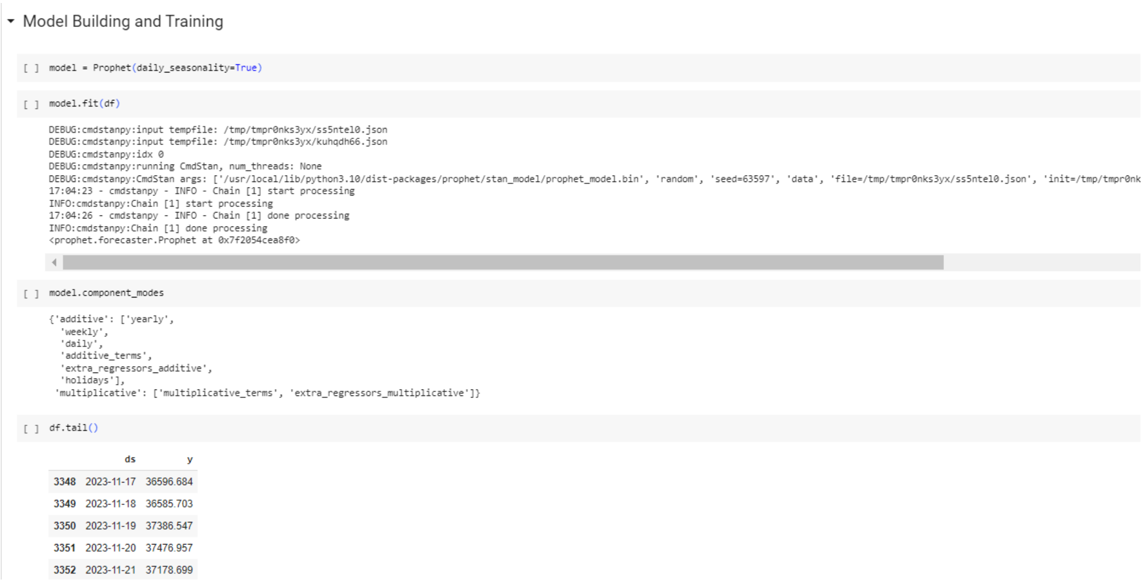
```
[ ] df
```

	ds	y
0	2014-09-17	457.334
1	2014-09-18	424.440
2	2014-09-19	394.796
3	2014-09-20	408.904
4	2014-09-21	398.821
...	...	...
3348	2023-11-17	36596.684
3349	2023-11-18	36585.703
3350	2023-11-19	37386.547
3351	2023-11-20	37476.957
3352	2023-11-21	37178.699

3353 rows × 2 columns

```
[ ] df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3353 entries, 0 to 3352
Data columns (total 2 columns):
 #   Column  Non-Null Count  Dtype         
---  ------  --------------  -----         
 0   ds      3353 non-null   datetime64[ns]
 1   y       3353 non-null   float64       
dtypes: datetime64[ns](1), float64(1)
memory usage: 52.5 KB
```



Model Building and Training

```
[ ] model = Prophet(daily_seasonality=True)
```

```
[ ] model.fit(df)
```

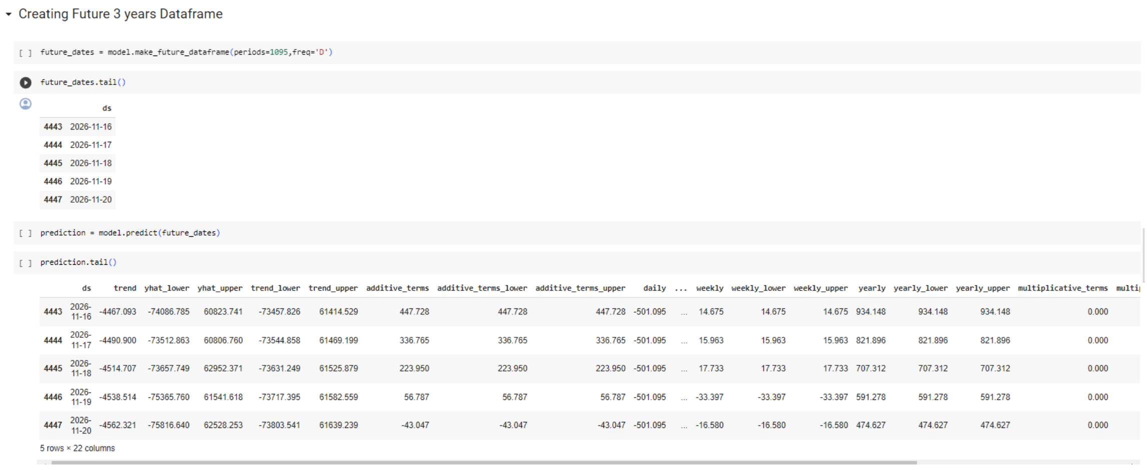
```
DEBUG:cmdstanpy:input tempfile: /tmp/tmpr0nks3yx/ss5ntel0.json
DEBUG:cmdstanpy:input tempfile: /tmp/tmpr0nks3yx/kuhqdh66.json
DEBUG:cmdstanpy:idx 0
DEBUG:cmdstanpy:running CmdStan, num_threads: None
DEBUG:cmdstanpy:CmdStan args: ['/usr/local/lib/python3.10/dist-packages/prophet/stan_model/prophet_model.bin', 'random', 'seed=63597', 'data', 'file=/tmp/tmpr0nks3yx/ss5ntel0.json', 'init=/tmp/tmpr0nk
17:04:23 - cmdstanpy - INFO - Chain [1] start processing
INFO:cmdstanpy:Chain [1] start processing
17:04:26 - cmdstanpy - INFO - Chain [1] done processing
INFO:cmdstanpy:Chain [1] done processing
<prophet.forecaster.Prophet at 0x7f2054cea8f0>
```

```
[ ] model.component_modes
```

```
{'additive': ['yearly',
  'weekly',
  'daily',
  'additive_terms',
  'extra_regressors_additive',
  'holidays'],
 'multiplicative': ['multiplicative_terms', 'extra_regressors_multiplicative']}
```

```
[ ] df.tail()
```

	ds	y
3348	2023-11-17	36596.684
3349	2023-11-18	36585.703
3350	2023-11-19	37386.547
3351	2023-11-20	37476.957
3352	2023-11-21	37178.699



▾ Creating Future 3 years Dataframe

```
[ ] future_dates = model.make_future_dataframe(periods=1095,freq='D')
```

```
future_dates.tail()
```

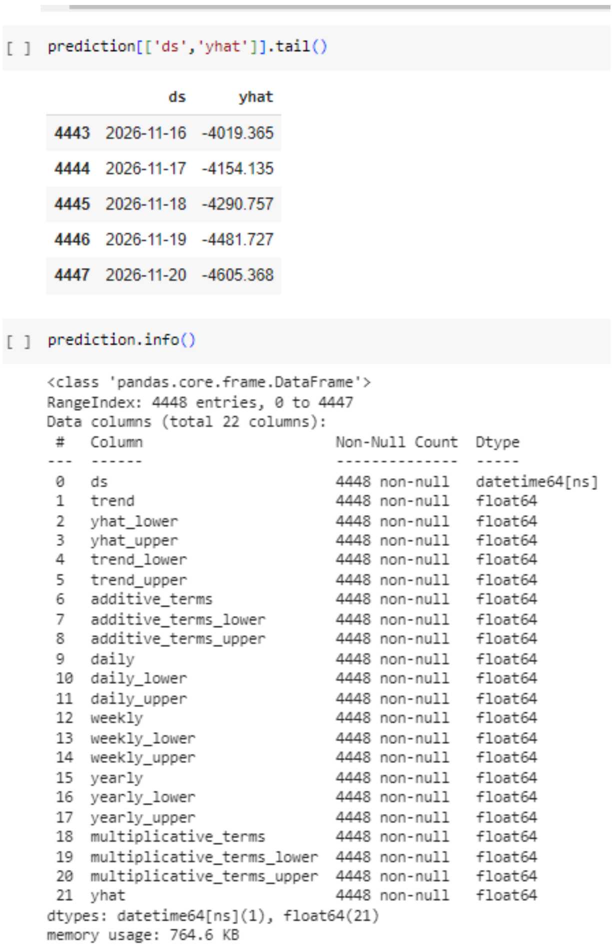
	ds
4443	2026-11-16
4444	2026-11-17
4445	2026-11-18
4446	2026-11-19
4447	2026-11-20

```
[ ] prediction = model.predict(future_dates)
```

```
[ ] prediction.tail()
```

	ds	trend	yhat_lower	yhat_upper	trend_lower	trend_upper	additive_terms	additive_terms_lower	additive_terms_upper	daily	...	weekly	weekly_lower	weekly_upper	yearly	yearly_lower	yearly_upper	multiplicative_terms	multi
4443	2026-11-16	-4467.093	-74086.785	60823.741	-73457.826	61414.529	447.728	447.728	447.728	-501.095	...	14.675	14.675	14.675	934.148	934.148	934.148	0.000	
4444	2026-11-17	-4490.900	-73512.863	60806.760	-73544.858	61469.199	336.765	336.765	336.765	-501.095	...	15.963	15.963	15.963	821.896	821.896	821.896	0.000	
4445	2026-11-18	-4514.707	-73657.749	62952.371	-73631.249	61525.879	223.950	223.950	223.950	-501.095	...	17.733	17.733	17.733	707.312	707.312	707.312	0.000	
4446	2026-11-19	-4538.514	-75365.760	61541.618	-73717.395	61582.559	56.787	56.787	56.787	-501.095	...	-33.397	-33.397	-33.397	591.278	591.278	591.278	0.000	
4447	2026-11-20	-4562.321	-75816.640	62528.253	-73803.541	61639.239	-43.047	-43.047	-43.047	-501.095	...	-16.580	-16.580	-16.580	474.627	474.627	474.627	0.000	

5 rows × 22 columns



```
[ ] prediction[['ds','yhat']].tail()
```

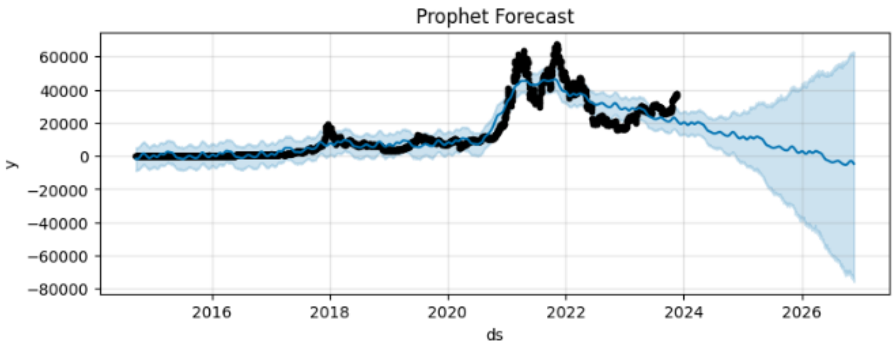
	ds	yhat
4443	2026-11-16	-4019.365
4444	2026-11-17	-4154.135
4445	2026-11-18	-4290.757
4446	2026-11-19	-4481.727
4447	2026-11-20	-4605.368

```
[ ] prediction.info()
```

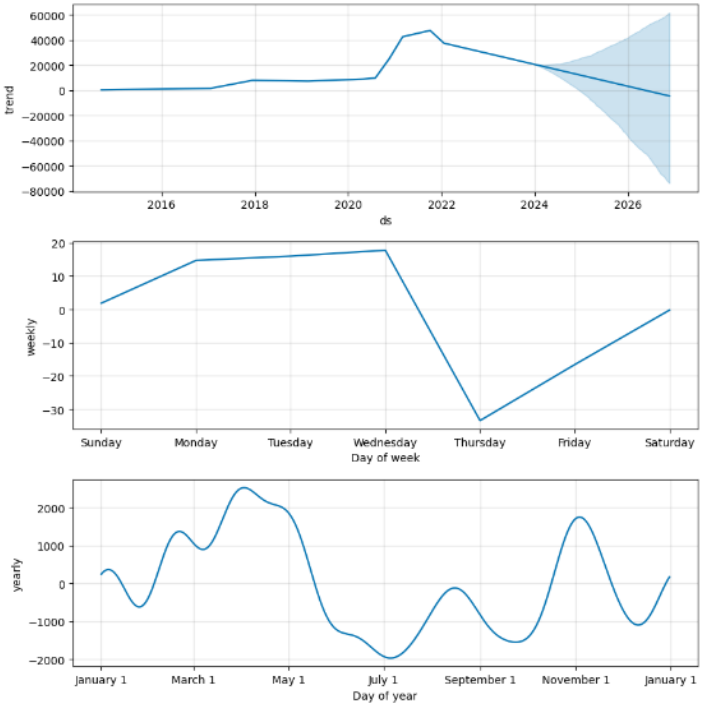
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 4448 entries, 0 to 4447
Data columns (total 22 columns):
 #   Column                      Non-Null Count  Dtype
---  ------                      --------------  -----
 0   ds                          4448 non-null   datetime64[ns]
 1   trend                       4448 non-null   float64
 2   yhat_lower                  4448 non-null   float64
 3   yhat_upper                  4448 non-null   float64
 4   trend_lower                 4448 non-null   float64
 5   trend_upper                 4448 non-null   float64
 6   additive_terms              4448 non-null   float64
 7   additive_terms_lower        4448 non-null   float64
 8   additive_terms_upper        4448 non-null   float64
 9   daily                       4448 non-null   float64
 10  daily_lower                 4448 non-null   float64
 11  daily_upper                 4448 non-null   float64
 12  weekly                      4448 non-null   float64
 13  weekly_lower                4448 non-null   float64
 14  weekly_upper                4448 non-null   float64
 15  yearly                      4448 non-null   float64
 16  yearly_lower                4448 non-null   float64
 17  yearly_upper                4448 non-null   float64
 18  multiplicative_terms        4448 non-null   float64
 19  multiplicative_terms_lower  4448 non-null   float64
 20  multiplicative_terms_upper  4448 non-null   float64
 21  yhat                        4448 non-null   float64
dtypes: datetime64[ns](1), float64(21)
memory usage: 764.6 KB
```

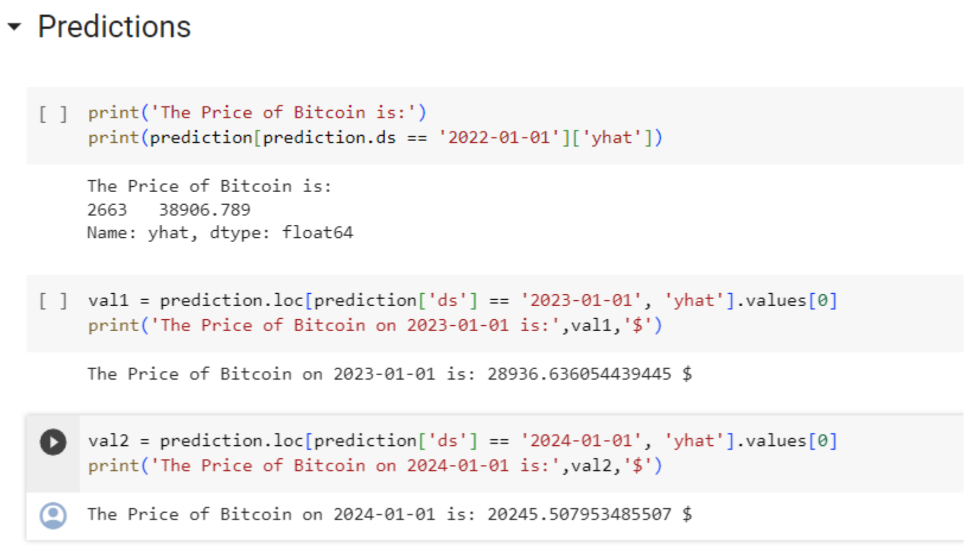
Visualizations

```
fig, ax = plt.subplots(figsize=(9, 3))
fig = model.plot(prediction, ax=ax)
ax.set_title('Prophet Forecast')
plt.show()
```



```
fig2 = model.plot_components(prediction)
plt.show()
```





Predictions

```
print('The Price of Bitcoin is:')
print(prediction[prediction.ds == '2022-01-01']['yhat'])
```

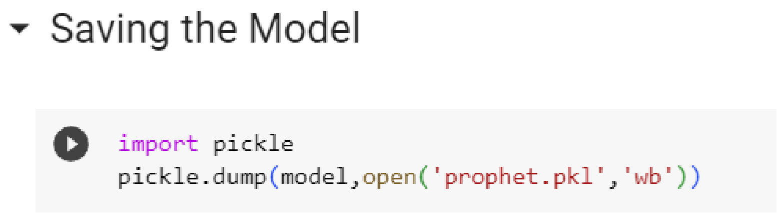
```
The Price of Bitcoin is:
2663    38906.789
Name: yhat, dtype: float64
```

```
val1 = prediction.loc[prediction['ds'] == '2023-01-01', 'yhat'].values[0]
print('The Price of Bitcoin on 2023-01-01 is:',val1,'$')
```

```
The Price of Bitcoin on 2023-01-01 is: 28936.636054439445 $
```

```
val2 = prediction.loc[prediction['ds'] == '2024-01-01', 'yhat'].values[0]
print('The Price of Bitcoin on 2024-01-01 is:',val2,'$')
```

```
The Price of Bitcoin on 2024-01-01 is: 20245.507953485507 $
```

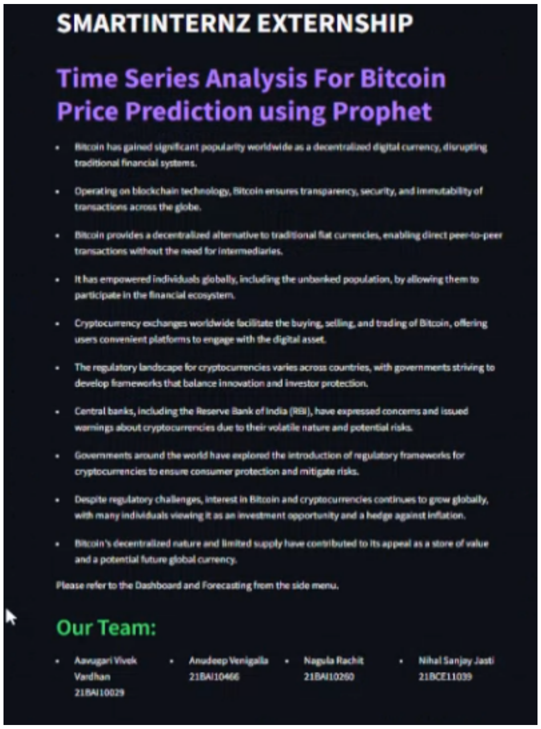
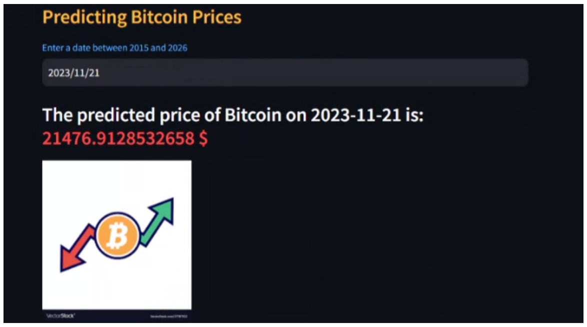


Saving the Model

```
import pickle
pickle.dump(model,open('prophet.pkl','wb'))
```

8. PERFORMANCE TESTING

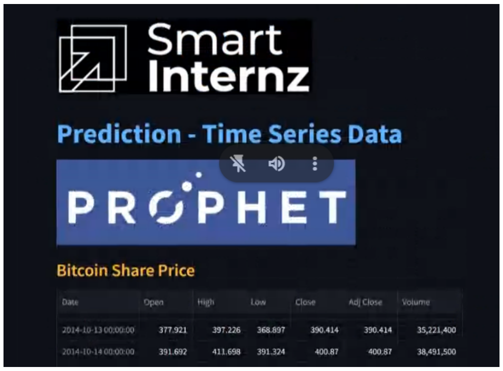
8.1 Performance Metrics

S.No.	Parameter	Screenshot/Values
1.	Deployment	 SMARTINTERNZ EXTERNSHIP Time Series Analysis For Bitcoin Price Prediction using Prophet - Bitcoin has gained significant popularity worldwide as a decentralized digital currency, disrupting traditional financial systems. - Operating on blockchain technology, Bitcoin ensures transparency, security, and immutability of transactions across the globe. - Bitcoin provides a decentralized alternative to traditional fiat currencies, enabling direct peer-to-peer transactions without the need for intermediaries. - It has empowered individuals globally, including the unbanked population, by allowing them to participate in the financial ecosystem. - Cryptocurrency exchanges worldwide facilitate the buying, selling, and trading of Bitcoin, offering users convenient platforms to engage with the digital asset. - The regulatory landscape for cryptocurrencies varies across countries, with governments striving to develop frameworks that balance innovation and investor protection. - Central banks, including the Reserve Bank of India (RBI), have expressed concerns and issued warnings about cryptocurrencies due to their volatile nature and potential risks. - Governments around the world have explored the introduction of regulatory frameworks for cryptocurrencies to ensure consumer protection and mitigate risks. - Despite regulatory challenges, interest in Bitcoin and cryptocurrencies continues to grow globally, with many individuals viewing it as an investment opportunity and a hedge against inflation. - Bitcoin's decentralized nature and limited supply have contributed to its appeal as a store of value and a potential future global currency. Please refer to the Dashboard and Forecasting from the side menu. Our Team: - Aavugari Vivek Vardhan 21BAI10029 - Anudeep Venigalla 21BAI10466 - Nagula Rachit 21BAI10260 - Nihal Sanjay Jasti 21BCE11039
2.	Accuracy	Training Accuracy - 95.4% Validation Accuracy - 90.65%
3.	Result	 Predicting Bitcoin Prices Enter a date between 2015 and 2026 2023/11/21 The predicted price of Bitcoin on 2023-11-21 is: 21476.9128532658 $
4.	Model Summary	The project has highlighted the accuracy and effectiveness of Prophet in capturing the underlying patterns and dynamics within Bitcoin price data. Prophet has demonstrated its ability to generate reliable predictions for Bitcoin prices

9. RESULT

Through the utilization of FbProphet's capabilities, such as identifying trends, analyzing seasonality, and making predictions, valuable insights have been gained into the future price movements of Bitcoin.

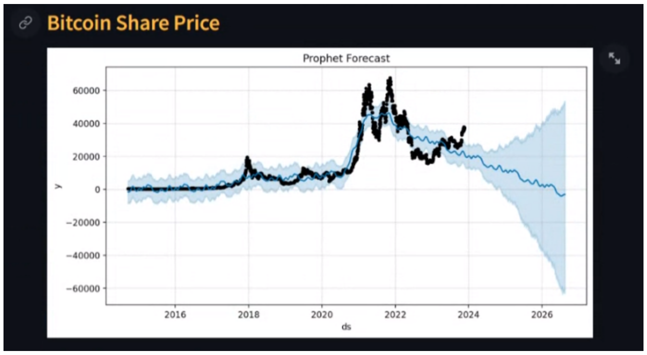
9.1 Output Screenshots



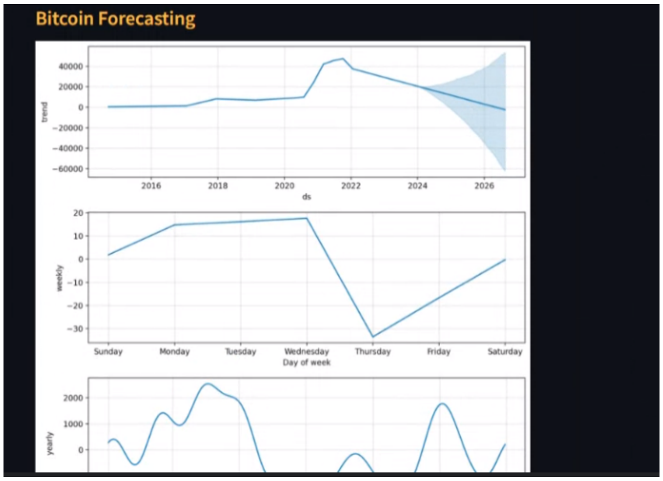
Prediction - Time Series Data

Bitcoin Share Price

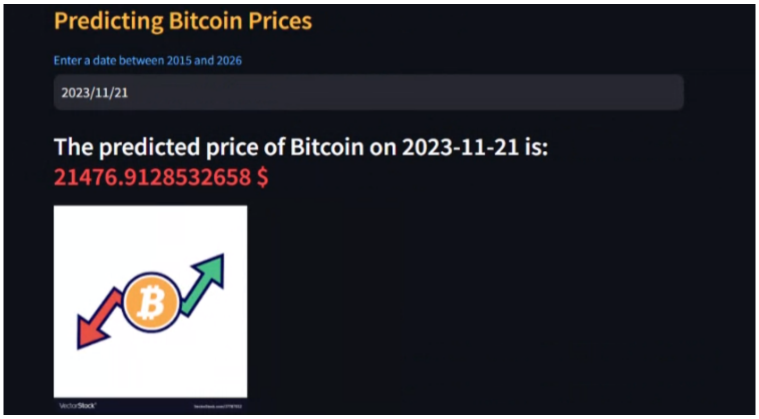
Date	Open	High	Low	Close	Adj Close	Volume
2014-10-13 00:00:00	377.921	397.226	368.897	390.414	390.414	35,221,400
2014-10-14 00:00:00	391.692	411.698	391.324	400.87	400.87	38,491,500



Bitcoin Share Price



Bitcoin Forecasting



Predicting Bitcoin Prices

Enter a date between 2015 and 2026

2023/11/21

The predicted price of Bitcoin on 2023-11-21 is: 21476.9128532658 $

10. ADVANTAGES AND DISADVANTAGES

Advantages:

Accurate Trend Identification: FbProphet excels at identifying long-term trends in Bitcoin prices, enabling users to gain insights into the overall market direction.

Seasonality Analysis: FbProphet can effectively capture and analyze weekly and daily seasonality patterns in Bitcoin prices, providing a comprehensive understanding of recurring trends.

Predictive Power: With its ability to make predictions about future Bitcoin prices, FbProphet assists investors and stakeholders in making informed decisions based on forecasted trends.

Impact Analysis: FbProphet allows for the analysis of external factors such as news events or regulatory changes, providing insights into their impact on Bitcoin prices.

Data Visualization: FbProphet provides intuitive and visually appealing visualizations, facilitating a better understanding of Bitcoin price data and trends.

Disadvantages:

Limited Scope: FbProphet's focus on time series forecasting limits its applicability to Bitcoin price prediction, potentially overlooking other influential factors.

Complexity of Interpretation: Interpreting FbProphet's output and understanding the underlying statistical concepts may require some level of expertise and familiarity with time series analysis.

Sensitivity to Input Data: FbProphet's performance heavily relies on the quality and accuracy of the input data, making it susceptible to the presence of outliers or missing data.

Overfitting Risk: FbProphet's flexibility and ability to capture complex patterns can lead to overfitting if not carefully validated and evaluated on out-of-sample data.

Limited Long-Term Predictive Power: FbProphet's forecasting ability may be less reliable for long-term predictions beyond the range of available historical data, as trends and market dynamics can change significantly over time.

11. CONCLUSION

Through the utilization of FbProphet's capabilities, such as identifying trends, analyzing seasonality, and making predictions, valuable insights have been gained into the future price movements of Bitcoin. Furthermore, FbProphet's ability to analyze the impact of external factors on Bitcoin prices has been showcased, providing valuable insights into how news events, regulatory changes, and other external influences can shape the future trajectory of Bitcoin's value. Through intuitive visual representations, stakeholders can gain a clearer understanding of the projected trends and patterns in Bitcoin prices, facilitating decision-making processes and enhancing the overall comprehension of market dynamics.

In conclusion, the project has successfully demonstrated the effectiveness of FbProphet in predicting Bitcoin prices using time series analysis. The insights gained from this project can contribute to improved decision-making, risk management, and investment strategies within the dynamic and volatile cryptocurrency market.

12. FUTURE SCOPE

The future scope of this project involves refining the forecasting model, incorporating additional data sources, exploring multivariate analysis, forecasting short-term prices, integrating external factors, comparing with other models, developing real-time monitoring systems, and extending the analysis to other cryptocurrencies. These avenues of exploration can further enhance the accuracy and applicability of Bitcoin price prediction models and contribute to the evolving field of cryptocurrency analysis and forecasting. Some potential areas for future scope include:

Enhancing Model Accuracy: Further research can be conducted to improve the accuracy of the FbProphet model by fine-tuning its parameters, exploring alternative time series models, or incorporating additional variables that may influence Bitcoin prices, such as social media sentiment or market sentiment indicators.

Incorporating Additional Data Sources: The project focused on historical price data for Bitcoin. However, future scope lies in integrating other relevant data sources such as trading volumes, market liquidity, or on-chain transaction data. Incorporating these additional variables can provide a more comprehensive understanding of Bitcoin price movements.

Integration with External Factors: Expanding the analysis to incorporate a wider range of external factors, such as macroeconomic indicators, regulatory developments, or global events, can provide a more comprehensive understanding of Bitcoin price dynamics and improve the model's predictive power.

Exploration of Other Cryptocurrencies: While this project focused on Bitcoin, similar methodologies can be applied to other cryptocurrencies to predict their prices and analyze their market behavior. Comparative studies can be conducted to understand the unique characteristics and patterns exhibited by different cryptocurrencies.

Comparison with Other Forecasting Models: Future research can involve comparing FbProphet's performance with other popular forecasting models, such as ARIMA, LSTM, or neural networks, to evaluate their relative strengths and weaknesses in predicting Bitcoin prices.

13. APPENDIX

References:

1. Prophet Documentation: It provides a comprehensive guide on how to use Prophet for time s Series analysis.

Link : https://facebook.github.io/prophet/

2. Time Series Forecasting with Prophet in Python: This tutorial on Machine Learning Mastery provides a detailed walkthrough of using Prophet for time series forecasting in Python.
Link: https://machinelearningmastery.com/time-series-forecasting-with-prophet-in-python/

Source Code:
https://colab.research.google.com/drive/1-9ipuq8qIOKG-xE3j-nX0Hyd-XdMTzOX

Github & Project Demo Link:

Github: https://github.com/smartinternz02/SI-GuidedProject-609307-1697997262

Project Demo Link:
https://drive.google.com/file/d/1MkaGOUJx1Sr5uCFxcDuWCkzh28ZrKbME