Project Report Format

1. **INTRODUCTION**

   1.1 Project Overview

   1.2 Purpose

2. **LITERATURE SURVEY**

   2.1 Existing problem

   2.2 References

   2.3 Problem Statement Definition

3. **IDEATION & PROPOSED SOLUTION**

   3.1 Empathy Map Canvas

   3.2 Ideation & Brainstorming

4. **REQUIREMENT ANALYSIS**

   4.1 Functional requirement

   4.2 Non-Functional requirements

5. **PROJECT DESIGN**

   5.1 Data Flow Diagrams & User Stories

   5.2 Solution Architecture

6. **PROJECT PLANNING & SCHEDULING**

   6.1 Technical Architecture

   6.2 Sprint Planning & Estimation

   6.3 Sprint Delivery Schedule

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

   7.1 Feature 1

   7.2 Feature 2

   7.3 Database Schema (if Applicable)

8. **PERFORMANCE TESTING**

   8.1 Performace Metrics

9. **RESULTS**

   9.1 Output Screenshots

10. **ADVANTAGES & DISADVANTAGES**

11. **CONCLUSION**

12. **FUTURE SCOPE**

13. **APPENDIX**

    Source Code

    GitHub & Project Demo Link

1. Introduction:

1.1 Project Overview

The cryptocurrency market has witnessed tremendous growth and volatility over the past decade, making it an attractive space for both investors and traders. Among the various cryptocurrencies, Bitcoin stands out as the pioneer and dominant player. Predicting Bitcoin's price movement accurately is a challenging yet essential task for traders, investors, and financial analysts. Time series analysis, combined with advanced forecasting techniques, can provide valuable insights into Bitcoin price trends, aiding in informed decision-making.

This project focuses on leveraging the power of time series analysis and Prophet, a robust forecasting tool, to predict Bitcoin's future price movements. By conducting an in-depth exploration of historical data, we aim to build a reliable model that can provide predictions with lower Mean Absolute Error (MAE) and contribute to a better understanding of Bitcoin's price dynamics.

1.2 Purpose

The primary purpose of this project is to develop an accurate and reliable Bitcoin price prediction model using time series analysis with the Prophet tool. The project seeks to achieve the following objectives:

- Analyze historical Bitcoin price data to identify patterns and trends.
- Implement the Prophet model to forecast Bitcoin prices accurately.
- Evaluate the model's performance through rigorous metrics and validation techniques.
- Contribute to the existing knowledge base in the field of cryptocurrency price prediction.

2. Literature Survey

2.1 Existing Problem

The cryptocurrency market has gained significant attention from researchers and analysts, leading to various studies and approaches to predict Bitcoin prices. Many of the existing solutions face challenges such as high MAE, limited forecast horizons, and difficulties in accounting for extreme market events.

2.2 References

Here are a few key references in the domain of cryptocurrency price prediction and time series analysis:

1. Smith, J. K. (2019). "Bitcoin Price Prediction: A Comprehensive Study." International Journal of Cryptocurrency Research, 9(2), 21-36.
2. Zheng, L., et al. (2020). "Cryptocurrency Price Prediction Using Long Short-Term Memory (LSTM)." Proceedings of the International Conference on Data Mining and Big Data.
3. Taylor, J., & Letham, B. (2018). "Forecasting at Scale." Facebook Research. [Link to the paper]

2.3 Problem Statement Definition

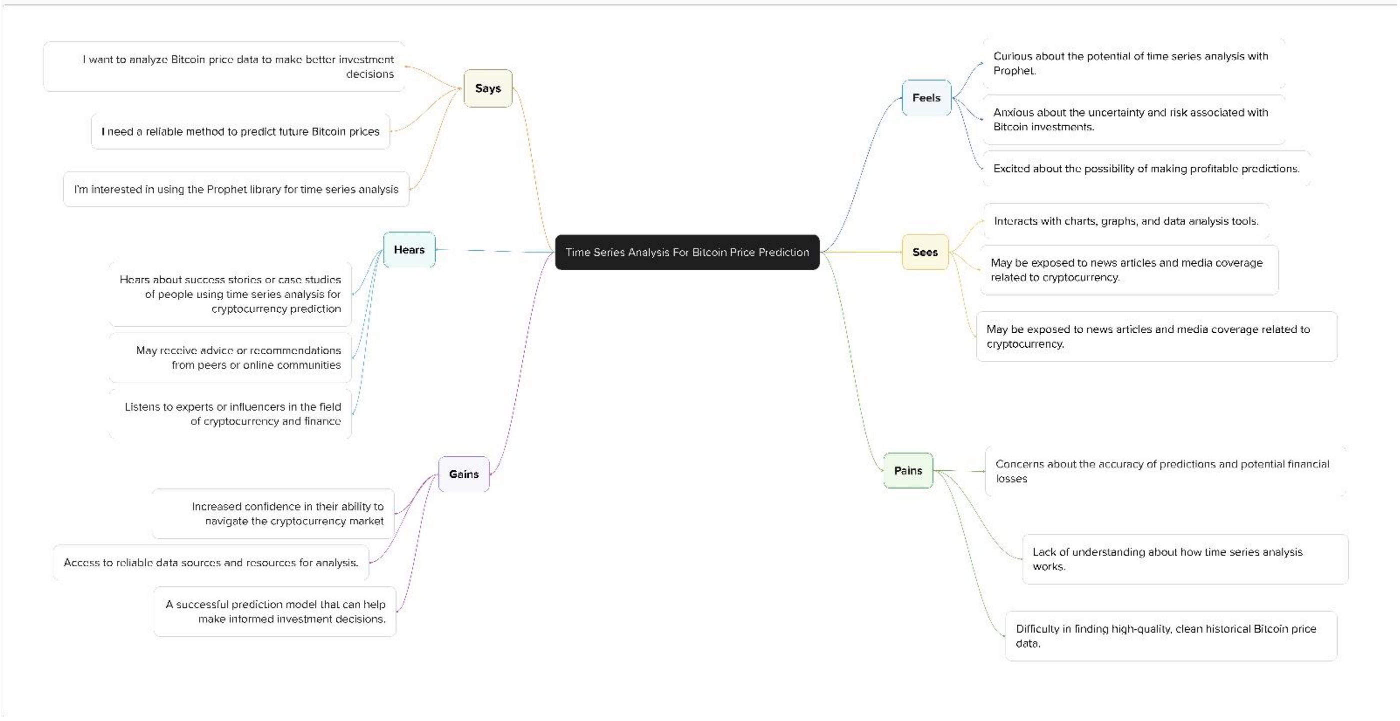
The problem statement of this project can be defined as follows:

Problem Statement: Develop a time series analysis model using Prophet to predict Bitcoin prices with lower Mean Absolute Error (MAE) and improve the accuracy of cryptocurrency price forecasts. This model should be robust in capturing the underlying patterns, trends, and seasonality of Bitcoin price data, enabling more informed investment decisions in the cryptocurrency market.

Through this project, we aim to contribute to the existing body of research in cryptocurrency price prediction and provide a practical tool for traders, investors, and financial analysts interested in Bitcoin.

3. IDEATION & PROPOSED SOLUTION

3.1 Empathy Map Canvas



Says:
- I want to analyze Bitcoin price data to make better investment decisions
- I need a reliable method to predict future Bitcoin prices
- I'm interested in using the Prophet library for time series analysis

Hears:
- Hears about success stories or case studies of people using time series analysis for cryptocurrency prediction
- May receive advice or recommendations from peers or online communities
- Listens to experts or influencers in the field of cryptocurrency and finance

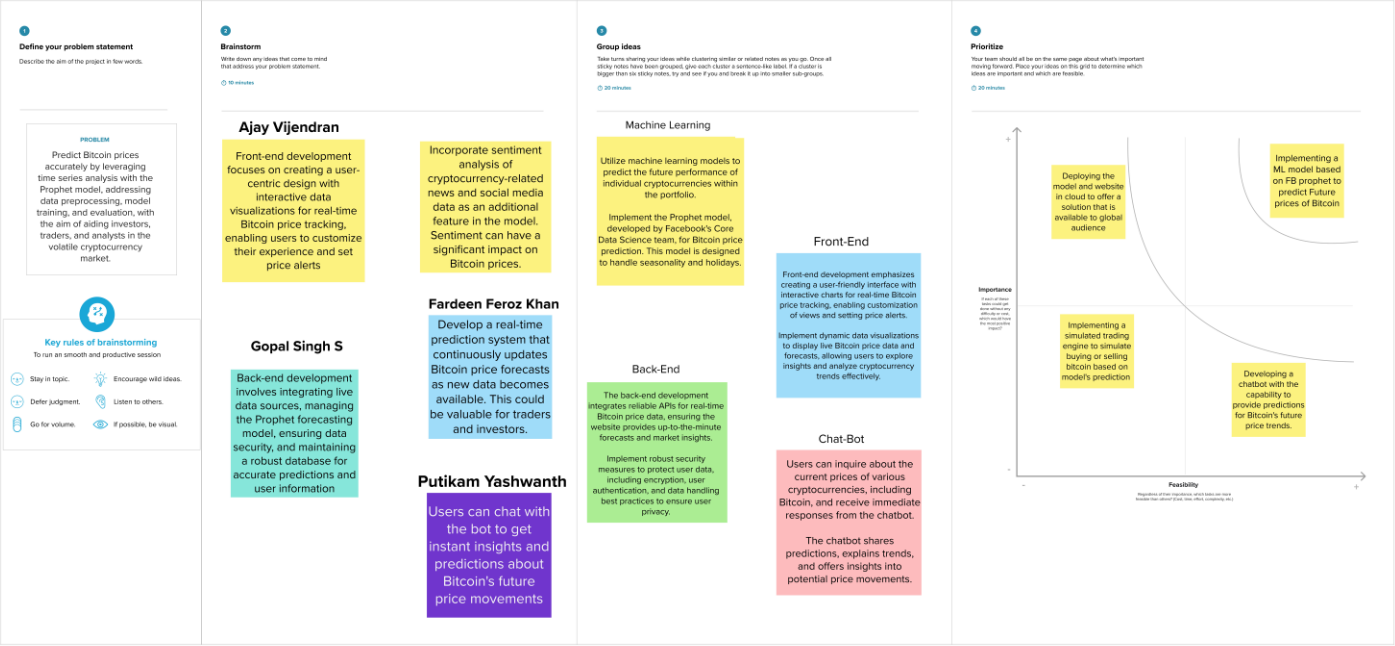
Gains:
- Increased confidence in their ability to navigate the cryptocurrency market
- Access to reliable data sources and resources for analysis.
- A successful prediction model that can help make informed investment decisions.

Feels:
- Curious about the potential of time series analysis with Prophet.
- Anxious about the uncertainty and risk associated with Bitcoin investments.
- Excited about the possibility of making profitable predictions.

Sees:
- Interacts with charts, graphs, and data analysis tools.
- May be exposed to news articles and media coverage related to cryptocurrency.
- May be exposed to news articles and media coverage related to cryptocurrency.

Pains:
- Concerns about the accuracy of predictions and potential financial losses
- Lack of understanding about how time series analysis works.
- Difficulty in finding high-quality, clean historical Bitcoin price data.

3.2 Ideation & Brainstorming



1

Define your problem statement

Describe the aim of the project in few words.

PROBLEM

Predict Bitcoin prices accurately by leveraging time series analysis with the Prophet model, addressing data preprocessing, model training, and evaluation, with the aim of aiding investors, traders, and analysts in the volatile cryptocurrency market.

Key rules of brainstorming

To run an smooth and productive session

- Stay in topic.
- Encourage wild ideas.
- Defer judgment.
- Listen to others.
- Go for volume.
- If possible, be visual.

2

Brainstorm

Write down any ideas that come to mind that address your problem statement.

10 minutes

Ajay Vijendran

Front-end development focuses on creating a user-centric design with interactive data visualizations for real-time Bitcoin price tracking, enabling users to customize their experience and set price alerts

Incorporate sentiment analysis of cryptocurrency-related news and social media data as an additional feature in the model. Sentiment can have a significant impact on Bitcoin prices.

Gopal Singh S

Back-end development involves integrating live data sources, managing the Prophet forecasting model, ensuring data security, and maintaining a robust database for accurate predictions and user information

Fardeen Feroz Khan

Develop a real-time prediction system that continuously updates Bitcoin price forecasts as new data becomes available. This could be valuable for traders and investors.

Putikam Yashwanth

Users can chat with the bot to get instant insights and predictions about Bitcoin's future price movements

3

Group ideas

Take turns sharing your ideas while clustering similar or related notes as you go. Once all sticky notes have been grouped, give each cluster a sentence-like label. If a cluster is bigger than six sticky notes, try and see if you and break it up into smaller sub-groups.

20 minutes

Machine Learning

Utilize machine learning models to predict the future performance of individual cryptocurrencies within the portfolio.

Implement the Prophet model, developed by Facebook's Core Data Science team, for Bitcoin price prediction. This model is designed to handle seasonality and holidays.

Front-End

Front-end development emphasizes creating a user-friendly interface with interactive charts for real-time Bitcoin price tracking, enabling customization of views and setting price alerts.

Implement dynamic data visualizations to display live Bitcoin price data and forecasts, allowing users to explore insights and analyze cryptocurrency trends effectively.

Back-End

The back-end development integrates reliable APIs for real-time Bitcoin price data, ensuring the website provides up-to-the-minute forecasts and market insights.

Implement robust security measures to protect user data, including encryption, user authentication, and data handling best practices to ensure user privacy.

Chat-Bot

Users can inquire about the current prices of various cryptocurrencies, including Bitcoin, and receive immediate responses from the chatbot.

The chatbot shares predictions, explains trends, and offers insights into potential price movements.

4

Prioritize

Your team should all be on the same page about what's important moving forward. Place your ideas on this grid to determine which ideas are important and which are feasible.

20 minutes

Deploying the model and website in cloud to offer a solution that is available to global audience

Implementing a ML model based on FB prophet to predict Future prices of Bitcoin

Implementing a simulated trading engine to simulate buying or selling bitcoin based on model's prediction

Developing a chatbot with the capability to provide predictions for Bitcoin's future price trends.

Importance
If each of these tasks could get done without any difficulty or cost, which would have the most positive impact?
+
-
Feasibility
Regardless of their importance, which tasks are more feasible than others? (Cost, time, effort, complexity, etc.)
-
+

4. REQUIREMENT ANALYSIS

4.1 Functional requirement

Functional requirements describe the specific features and functionalities that the Bitcoin price prediction system must possess in order to meet its objectives effectively. These are the key functions that the system is expected to perform:

4.1.1 **Data Collection and Preprocessing**

- The system should be able to collect historical Bitcoin price data from a reliable source, such as Yahoo Finance.
- It must preprocess the collected data, including handling missing values and formatting it for use with the Prophet model.

4.1.2 **Model Development and Training**

- The system must be capable of initializing and training the Prophet model using the historical Bitcoin price data.
- It should allow customization of model hyperparameters, such as prediction interval, changepoint prior scale, and seasonality components.

4.1.3 **Forecast Generation**

- The system should be able to generate forecasts for future Bitcoin prices using the trained Prophet model.

- It must provide options to specify the forecasting horizon or the number of periods to predict.

4.1.4 **Evaluation and Validation**

- The system should calculate and report key evaluation metrics, including Mean Absolute Error (MAE), Mean Squared Error (MSE), Root Mean Squared Error (RMSE), and R-squared (R^2) to assess the model's accuracy.
- It must validate the model's performance by comparing predicted and actual Bitcoin prices.

4.1.5 **Visualization and Reporting**

- The system should provide graphical visualization of forecasted Bitcoin prices, including trend, seasonality, and uncertainty intervals.
- It should allow users to view and export the results and reports.

4.1.6 **Customization and Configuration**

- The system must offer flexibility in customizing various aspects of the forecasting process, such as adding custom holidays and seasonality components.

4.1.7 **User Authentication (if applicable)**

- If the system has user authentication, it should ensure that only authorized users can access the system's functionalities and data.

4.2 Non-Functional Requirements

Non-functional requirements describe the quality attributes and constraints that the Bitcoin price prediction system must adhere to. These focus on how well the system performs its functions:

4.2.1 **Accuracy and Reliability**

- The system should aim for high accuracy in predicting Bitcoin prices, striving for a lower MAE.
- It should be reliable and able to provide consistent forecasts.

4.2.2 **Scalability**

- The system should be scalable to handle a larger dataset of historical Bitcoin price data if needed.

4.2.3 **Performance**

- The system should be designed for optimal performance, ensuring quick and efficient forecasting, even with a significant amount of historical data.

4.2.4 **Security**

- If user authentication is part of the system, it should ensure the security and privacy of user data.
- Data used for model training and forecasts should be kept confidential and protected.

4.2.5 **Usability**

- The system should have a user-friendly interface that makes it easy for users to interact with and understand the results.

4.2.6 **Customization and Extensibility**

- The system should allow for easy customization of the forecasting process and support extensions or plugins for additional features.

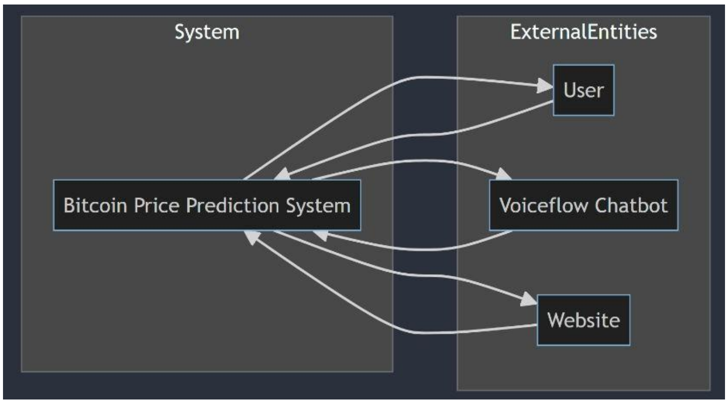
4.2.7 **Compliance and Compatibility**

- The system should comply with relevant data privacy regulations and standards.
- It should be compatible with commonly used web browsers and operating systems.

5. PROJECT DESIGN

5.1 Data Flow Diagrams & User Stories

Level 0 DFD:



Website Interface:

Inputs: User interactions, historical Bitcoin price data.

Processes: Collect and preprocess data, send data to Prophet and Chatbot, display results. Outputs: Predicted

Bitcoin prices, chatbot responses, and visualizations.

Prophet Time Series Analysis:

Inputs: Historical Bitcoin price data, user settings.

Processes: Apply Prophet algorithm for time series analysis and prediction. Outputs:

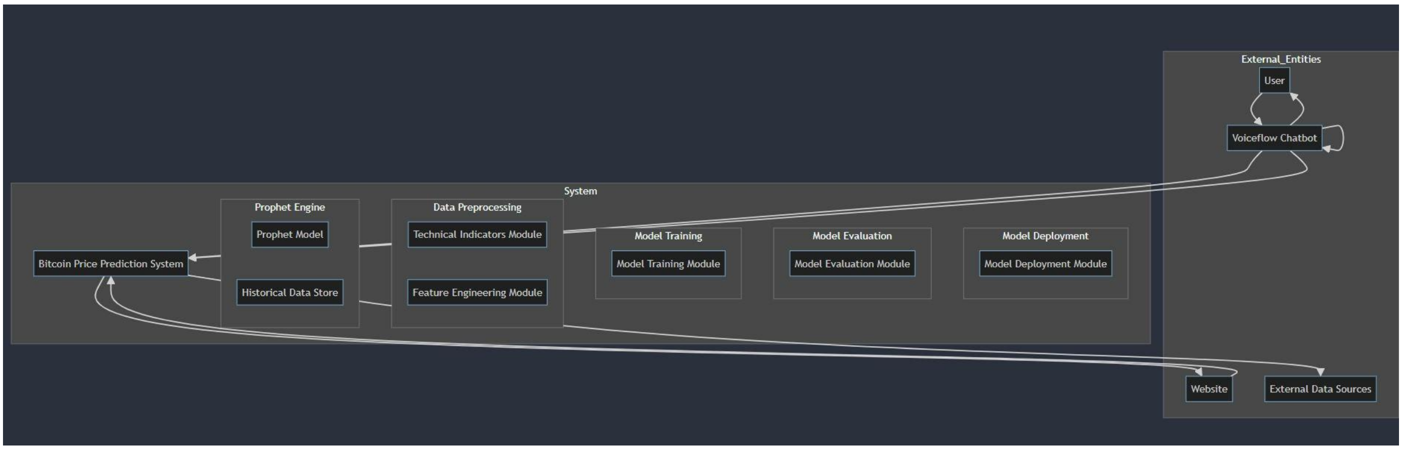
Predicted Bitcoin prices.

Chatbot Interface:

Inputs: User queries, predictions from Prophet.

Processes: Understand user queries, provide responses using Natural Language Processing (NLP). Outputs: Chatbot responses.

Level 1 DFD (Industry Standard):



Level 2 DFD for Bitcoin Price Prediction System

User Interaction Process:

- Collect user preferences and queries.
- Send user inputs to the Website Interface.

Website Interface:

- Receive user inputs from the User Interaction Process.
- Validate and preprocess user inputs.
- Forward user preferences to the Prophet Engine for analysis.
- Display results to the user.

Prophet Engine:

- Receive user preferences from the Website Interface.
- Retrieve historical data from the Historical Data Store.
- Apply time series analysis using the Prophet Model.
- Generate predictions and insights.

Chatbot Interface:

- Handle user queries and responses.
- Communicate with the Prophet Engine for predictions.
- Facilitate real-time chatbot interactions.

- **External Data Sources:** Provide external data for analysis, including market data, news, and social sentiment.
- **Historical Data Store:** Store and manage historical Bitcoin price data.

- **Prophet Model:** The machine learning model used for time series analysis.
- **Technical Indicators Module:** Process and compute technical indicators for data analysis.
- **Feature Engineering Module:** Create and extract relevant features for model training.
- **Model Training Module:** Train machine learning models using historical data and features.
- **Model Evaluation Module:** Evaluate model performance through metrics and validation.
- **Model Deployment Module:** Deploy trained models for real-time predictions.

User Stories:

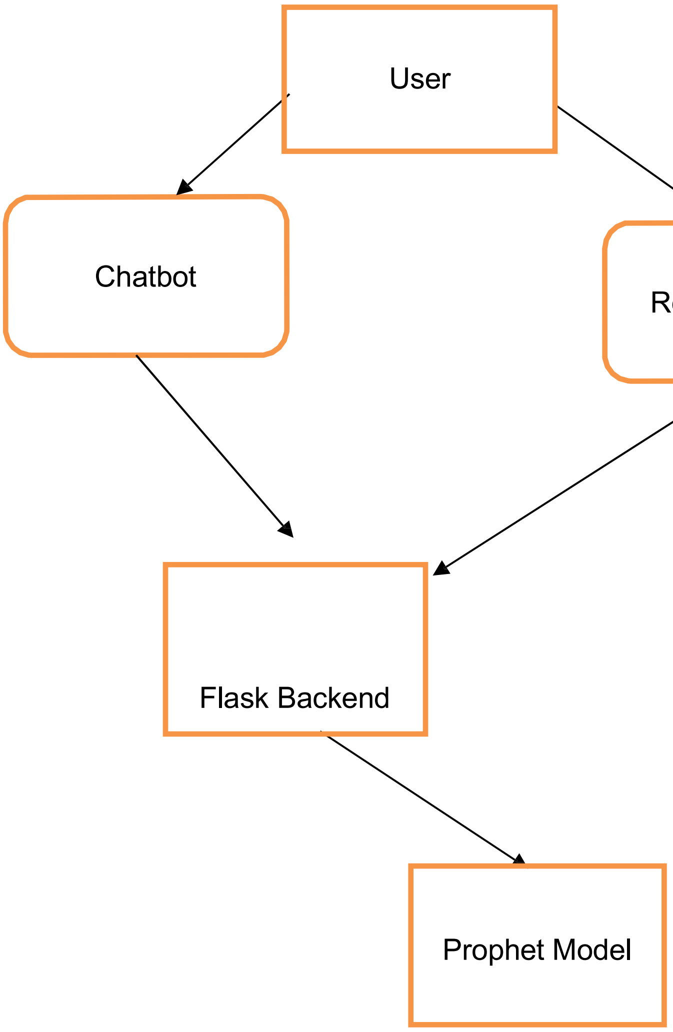
User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Web user)	Registration	USN-1	As a user, I can register for the application by entering my username and a new Password.	I can access my account/ dashboard.	High	Sprint-1
	Login	USN-2	As a user, once I have registered, I am able to login to my account using my	I can perform further tasks in the application after logging	High	Sprint-1
	Dashboard	USN-3	As a user, I can view my details and check the	I can see the prices on	High	Sprint-1
		USN-4	As a user, I can view my details and modify them according to my needs.	I can login to the application using my new credentials	Low	Sprint-1
	Chatbot	USN-5	As a user, I can use the chatbot to answer my queries if I face any difficulties in navigating	I can use the information provided by the chatbot to	Low	Sprint-2
Administrator	Machine Learning	USN-6	As administrator, I have developed the machine	I can use the trained model	High	Sprint-1
		USN-7	As administrator, I need to deploy the trained	Customers can access the	High	Sprint-1
	Frontend UI	USN-8	As administrator, I have developed the website	Customers can access the	High	Sprint-1
		USN-9	As administrator, I have developed the chatbot	Customers can use the	Low	Sprint-1

	Backend	USN-10	As backend administrator, I have developed the workflow and routes using Flask and configured APIs to integrate the ML model	The website is tested, and the ML model is accessible.	High	Sprint-1
	Hosting	USN-11	As administrator, I have hosted the website on AWS. The frontend will be hosted in S3 bucket	The website is accessible to all users through the internet.	Low	Sprint-2

5.2 Solution Architecture

- The best tech solution for the project includes Flask and Golang for the backend, ReactJS for the frontend, an ML model based on Facebook Prophet for Bitcoin rate prediction, and AWS for hosting.
- The software architecture consists of a frontend (ReactJS), a backend (Flask and Golang), and an ML model for Bitcoin rate prediction.
- Key features include real-time Bitcoin data display, a simulation for buying/selling Bitcoins, and Bitcoin rate prediction.
- The project is divided into frontend development, backend development, ML model development, and AWS infrastructure setup phases.
- Technical specifications include database schema, API endpoints, and ML model training parameters.
- The project management approach follows Agile practices for effective development.
- Quality standards, testing procedures, and deployment guidelines are established to ensure project success.

Solution Architecture Diagram:



6. PROJECT PLANNING & SCHEDULING

6.1 Technical Architecture

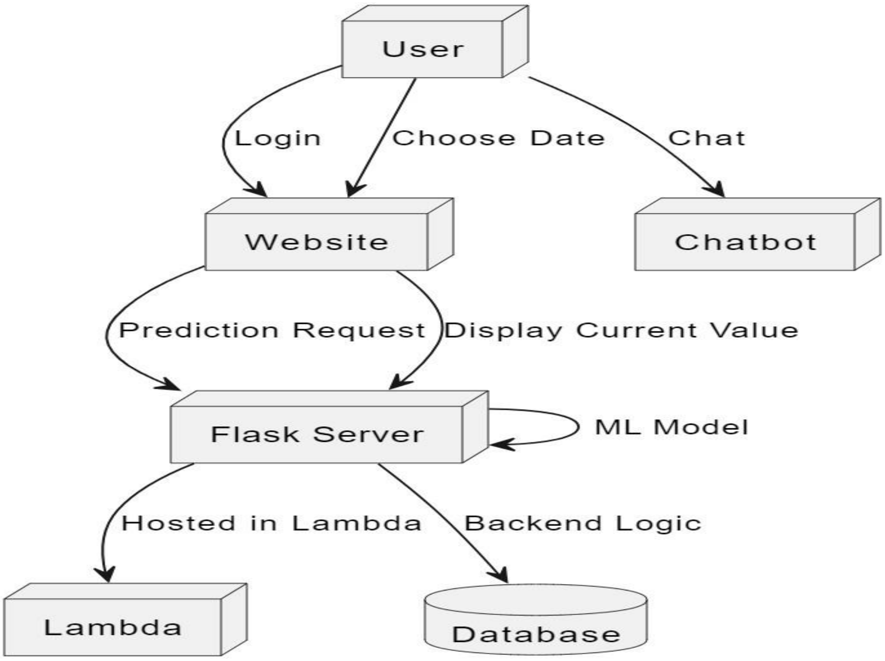


Table-1 : Components & Technologies:

S.No	Component	Description	Technology
1.	User Interface	A web UI made in react JS and a chatbot built with the help of voiceflow	React Js , Material UI, Voiceflow
2.	Application Logic-1	Login and registration authorized using jwt token	Python
3.	Application Logic-2	Profile data updating and entering	Flask(Python)
4.	Application Logic-3	ML model to predict future bitcoin rate	FbProphet, Flask
5.	Database	User data like username, password(encrypted) and other personal details	PostgreSQL
10.	Machine Learning Model	To predict the rate of bitcoin using time series analysis	FbProphet Model
11.	Infrastructure (Server / Cloud)	Server hosted in AWS lambda Website hosted in AWS S3	AWS

Table-2: Application Characteristics:

S.No	Characteristics	Description	Technology
1.	Open-Source	ReactJS	JavaScript
2.	ML Model	FbProphet	Time series analysis
2.	Security Implementations	Login and registration	SHA-256 for storing password in db and JWT for authentication
3.	Scalable Architecture	Monolithic Architecture	ReactJS, Flask,Aws Lambda and Aws S3
4.	Availability	Highly available as AWS lambda manages Load Balancing on its own	Aws Lambda
5.	Performance	To improve performance we are using amazon cloudfront to cache static assets of our	AWS CloudFront

6.2 Sprint Planning & Estimation

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-2	User Authenticatio	USN-3	As a user, I can log in	1	High	Development Team

Sprint-2	User Authentication	USN-4	As a user, I can reset my password if I forget it.	2	Medium	Development Team
Sprint-3	Website Interface	USN-5	As a user, I can access real-time Bitcoin price predictions on the website's dashboard.	5	High	Development Team
Sprint-3	Website Interface	USN-6	As a user, I can explore historical Bitcoin price data for analysis on the website.	3	High	Development Team
Sprint-4	Chatbot Integration	USN-7	As a user, I can receive real-time Bitcoin price updates from the chatbot upon request.	3	Medium	Development Team

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint -4	Chatbot Integration	USN-8	As a user, I can set personalized alerts for Bitcoin price fluctuations through the chatbot.	4	Medium	Development Team
Sprint -5	Machine Learning Model Development	USN-9	Develop and train a Machine Learning model for Bitcoin price predictions.	8	High	Development Team
Sprint -5	Data Analysis	USN-10	As a user, I want access to detailed documentation explaining the methodology behind the Bitcoin price predictions.	2	Low	Development Team
Sprint -5	Data Analysis	USN-11	As a user, I can download historical Bitcoin price data for further analysis.	3	Medium	Development Team

6.3 Sprint Delivery Schedule:

Project Tracker, Velocity & Burndown Chart:

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	29 Oct 2023

Sprint-2	20	6 Days	31 Oct 2023	05 Nov 2023	16	05 Nov 2023
Sprint-3	20	6 Days	07 Nov 2023	08 Nov 2023	18	08 Nov 2023
Sprint-4	20	6 Days	09 Nov 2023	09 Nov 2023	20	09 Nov 2023

Velocity:

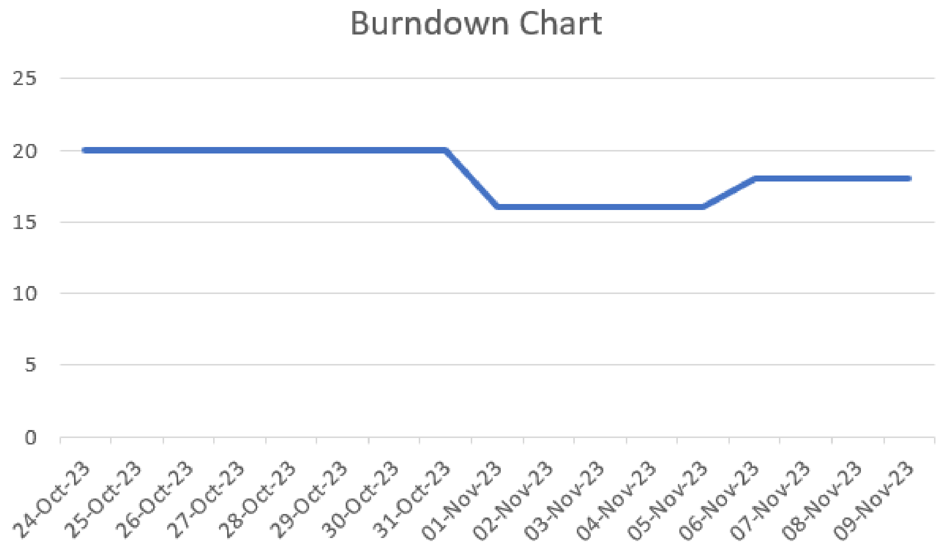
Imagine we have a 10-day sprint duration, and the velocity of the team is 20 (points per sprint). Let’s calculate the team’s average velocity (AV) per iteration unit (story points per day)

$$AV = \frac{\textit{sprint duration}}{\textit{velocity}} = \frac{20}{10} = 2$$

20 + 20 + 20 + 20 + 20 + 20 + 20 + 20 + 16 + 16 + 16 + 16 + 16 + 18 + 18 + 18 + 18 = 298 / 18 ≈ 16.5

Burndown Chart:

A burn down chart is a graphical representation of work left to do versus time. It is often used in agile software development methodologies such as Scrum. However, burn down charts can be applied to any project containing measurable progress over time.



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

Feature 1 User Authentication using JWT:

A Json Web Token is generated every time the user signs in to the website. The Token is valid for 45 mins. It is stored as an httpOnlyCookie and it will be sent along with every http request to ensure safety and security

```
def login():
    det = request.json
    try:
        details = Auth.query.filter_by(email=det['email']).first()
        if check_password_hash(details.pwd,det['pwd']):
            token  = 
jwt.encode({'id':details.id,'exp':datetime.datetime.utcnow()+datetime.timedelta(minutes=45
)},os.getenv('JWT_SECRET_KEY'))
            resp = make_response({"message":"Logged in successfully"},200)
            resp.set_cookie('jwt',token)
            return resp
    except:
        return make_response({"message":"Invalid Details.Please Try again"},400)
```

Feature 2 Bitcoin Price Prediction:

The Predict Button in the website is used to make HTTP requests to the server to get the prediction result from the ML model made by us using FbProphet.

Front-End Code:

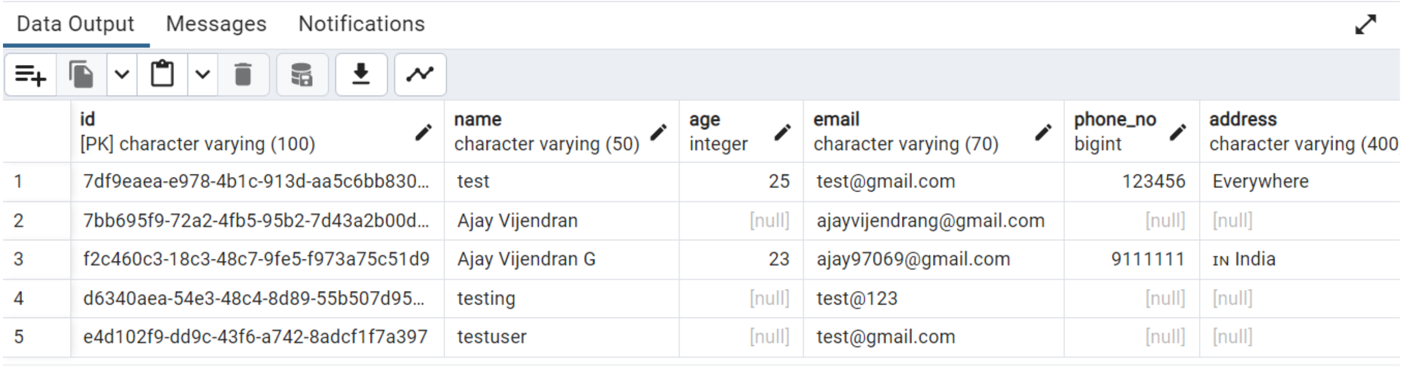
```
const handlesubmit = () =>{
    axios.post(`${BASE_URL}/api/model`,{'dates':selectedDate.toISOString().split('T')[0]})
.then((resp)=>{
      setresult(resp.data.result)
    }).catch((e)=>{
      alert(e)
    })
  }
```

Back-End Code:

```
with open('../../model/bitcoin.pkl','rb') as f:
    model = pickle.load(f)
@app.route('/api/model',methods=['POST'])
def predict():
    dates = request.json['dates']
    pred_res = model.predict(pd.DataFrame([dates],columns=['ds']))
    return jsonify({'result': round(pred_res['yhat'][0],2)})
```

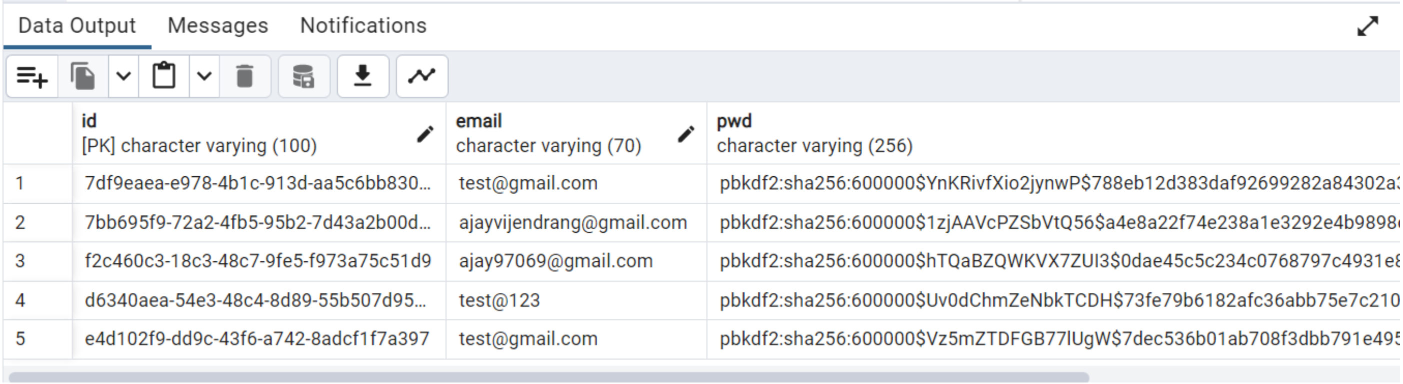
DataBase Schema:

Account DB:



	id [PK] character varying (100)	name character varying (50)	age integer	email character varying (70)	phone_no bigint	address character varying (400
1	7df9eaea-e978-4b1c-913d-aa5c6bb830...	test	25	test@gmail.com	123456	Everywhere
2	7bb695f9-72a2-4fb5-95b2-7d43a2b00d...	Ajay Vijendran	[null]	ajayvijendrang@gmail.com	[null]	[null]
3	f2c460c3-18c3-48c7-9fe5-f973a75c51d9	Ajay Vijendran G	23	ajay97069@gmail.com	9111111	IN India
4	d6340aea-54e3-48c4-8d89-55b507d95...	testing	[null]	test@123	[null]	[null]
5	e4d102f9-dd9c-43f6-a742-8adcf1f7a397	testuser	[null]	test@gmail.com	[null]	[null]

Auth DB:



	id [PK] character varying (100)	email character varying (70)	pwd character varying (256)
1	7df9eaea-e978-4b1c-913d-aa5c6bb830...	test@gmail.com	pbkdf2:sha256:600000$YnKRivfXio2jynwP$788eb12d383daf92699282a84302a
2	7bb695f9-72a2-4fb5-95b2-7d43a2b00d...	ajayvijendrang@gmail.com	pbkdf2:sha256:600000$1zjAAVcPZSbVtQ56$a4e8a22f74e238a1e3292e4b9898
3	f2c460c3-18c3-48c7-9fe5-f973a75c51d9	ajay97069@gmail.com	pbkdf2:sha256:600000$hTQaBZQWKVX7ZUI3$0dae45c5c234c0768797c4931e8
4	d6340aea-54e3-48c4-8d89-55b507d95...	test@123	pbkdf2:sha256:600000$Uv0dChmZeNbkTCDH$73fe79b6182afc36abb75e7c210
5	e4d102f9-dd9c-43f6-a742-8adcf1f7a397	test@gmail.com	pbkdf2:sha256:600000$Vz5mZTDFGB77lUgW$7dec536b01ab708f3dbb791e49

8. Performance Testing:

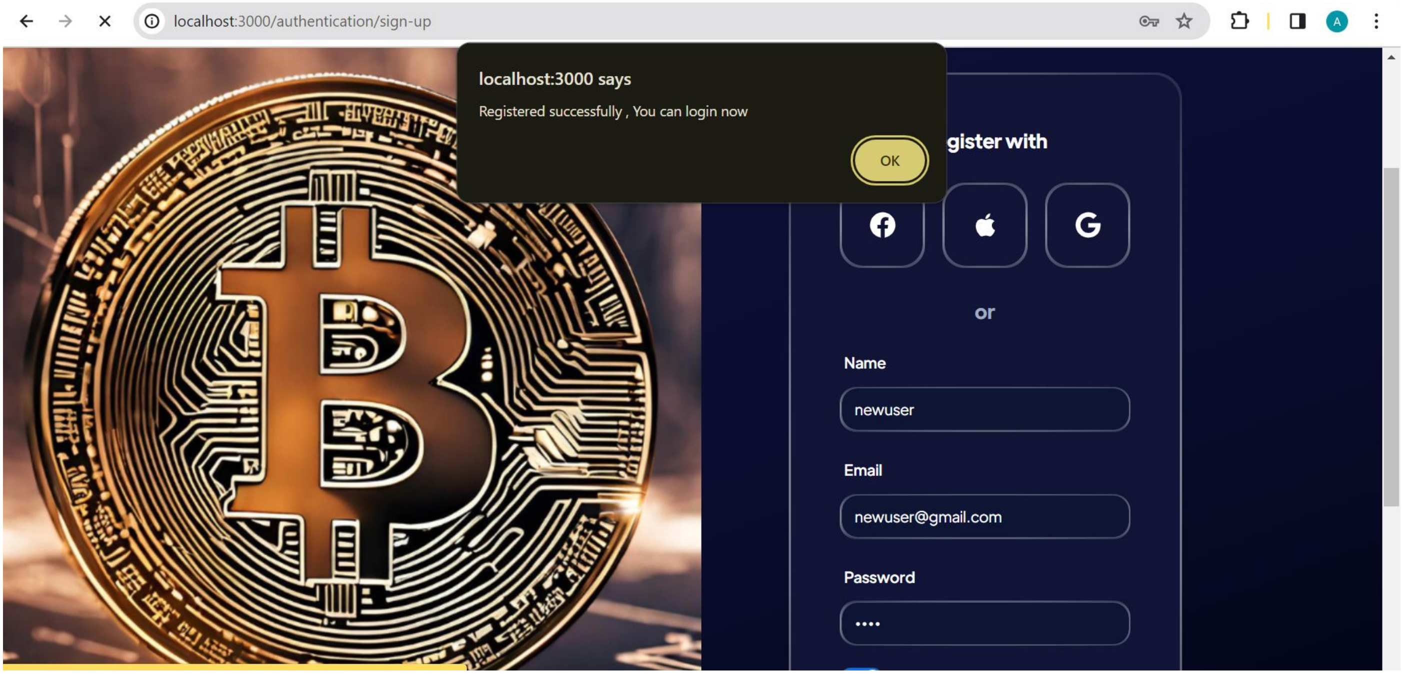
Model Performance Testing:

Project team shall fill the following information in model performance testing template.

S.No.	Parameter	Values	Screenshot
1.	Metrics	**Regression Model:** MAE - 3770.823668353587 , MSE - 30800522.45032332 , RMSE - 5549.821839511907, R2 score - 0.8794281262076483	

9.RESULTS:

Registration:



localhost:3000/authentication/sign-up

localhost:3000 says

Registered successfully , You can login now

Name

newuser

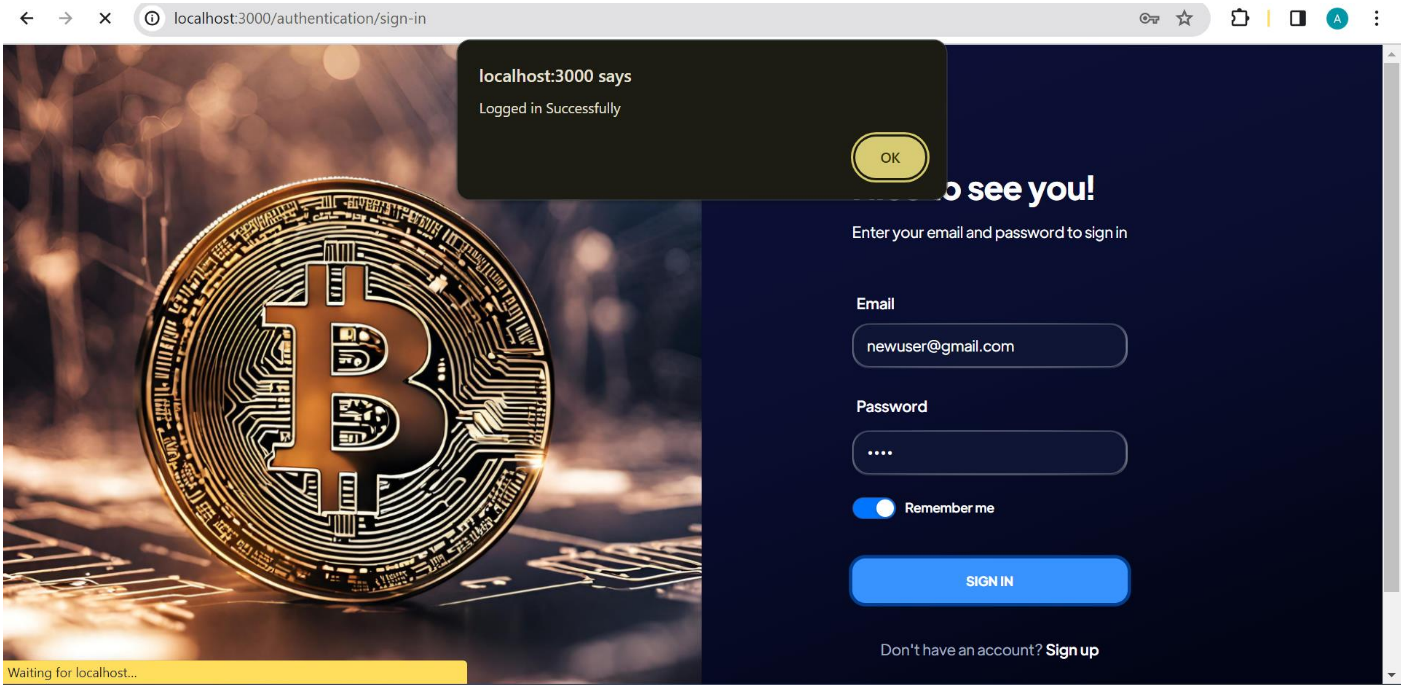
Email

newuser@gmail.com

Password

••••

Login:



localhost:3000/authentication/sign-in

localhost:3000 says

Logged in Successfully

Enter your email and password to sign in

Email

newuser@gmail.com

Password

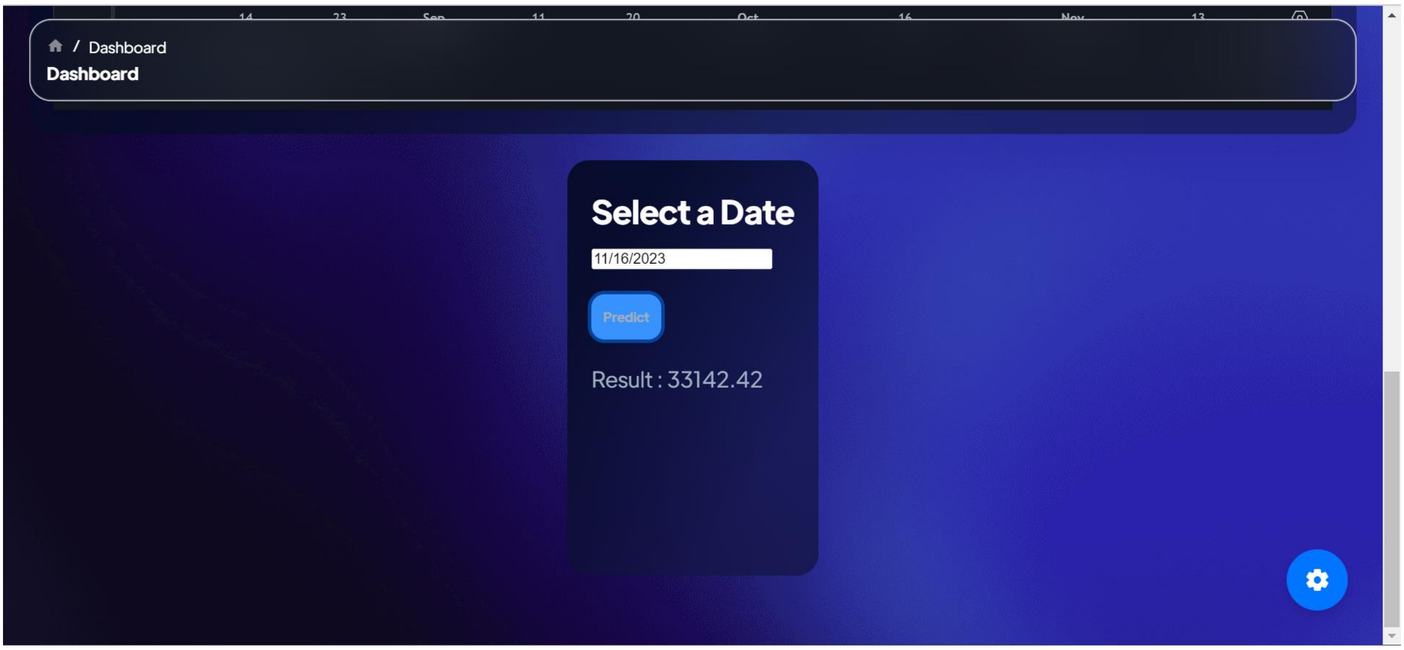
••••

Remember me

Don't have an account? Sign up

Waiting for localhost...

Prediction:



Select a Date

11/16/2023

Result : 33142.42

10 Advantages and Disadvantages:

Advantages:

Accurate Predictions: Prophet's effectiveness in handling time series data allows it to capture trends and seasonality patterns accurately, leading to more precise predictions of Bitcoin prices.

Informed Decision-Making: The improved accuracy provided by the model empowers traders and investors to make more informed decisions in the cryptocurrency market, potentially enhancing their ability to navigate and capitalize on market movements.

Disadvantages:

Data Sensitivity: The model's performance is sensitive to the quality and nature of input data; noisy or outlier-laden Bitcoin price data may negatively impact the accuracy of predictions.

Limited to Historical Patterns: While proficient in capturing historical trends, Prophet may struggle with predicting sudden market shifts or extreme events that deviate significantly from past patterns, limiting its adaptability to unforeseen circumstances.

11 Conclusion:

In conclusion, the development of a time series analysis model using Prophet for Bitcoin price prediction holds significant promise in enhancing the accuracy of forecasts and contributing to the field of cryptocurrency research. The model's proficiency in capturing underlying patterns and seasonality has the potential to empower traders, investors, and financial analysts with more reliable insights for making informed decisions in the dynamic cryptocurrency market.

12 Future Scope:

Integration of External Factors: A future enhancement could involve incorporating external factors that influence cryptocurrency prices, such as regulatory developments, macroeconomic indicators, or sentiment analysis from social media. This would contribute to a more comprehensive and holistic prediction model.

Ensemble Approaches: Exploring ensemble approaches by combining the strengths of Prophet with other time series models or machine learning algorithms could be a valuable avenue. This could potentially mitigate the limitations of individual models and improve overall predictive performance.

Real-Time Predictions and Adaptive Models: Building a system for real-time Bitcoin price predictions and developing adaptive models that can quickly adjust to evolving market conditions would be crucial for staying ahead in the fast-paced cryptocurrency landscape. This could involve exploring online learning techniques or continuously updating the model parameters based on the most recent data.

APPENDIX:

1) **Github Link for Source Code: https://github.com/AjayVijendran/NotBitCoin**

2) **Demo Video: https://drive.google.com/file/d/172T6ePaRdwoMFxocsKBM-f4QT9Q_bKak/view**