

# **CRUDE OIL PRICE PREDICTION**

**TEAM ID - PNT2022TMID32702**

## **A PROJECT REPORT**

*Submitted by*

**L.RAMABATHREN**

**(813819104304)**

**K.RAGUL**

**(813819104071)**

**S.VIJAY PAUL RAJ**

**(813819104115)**

**P.HENRY JONES**

**(813819104303)**

# CONTENTS

## **1. INTRODUCTION**

1.1 Project Overview

1.2 Purpose

## **2. LITERATURE SURVEY**

2.1 Existing problem

2.2 Problem Statement Definition

2.3 References

## **3. IDEATION & PROPOSED SOLUTION**

3.1 Empathy Map Canvas

3.2 Ideation & Brainstorming

3.3 Proposed Solution

3.4 Problem Solution fit

## **4. REQUIREMENTS ANALYSIS**

4.1 Functional requirement

4.2 Non-Functional requirements

## **5. PROJECT DESIGN**

5.1 Data Flow Diagram

5.2 Solution & Technical Architecture

5.2.1 Solution Architecture

5.2.2 Technical Architecture

5.3 User Stories

## **6. PROJECT PLANNING & SCHEDULING**

6.1 Sprint Planning & Estimation

6.2 Sprint Delivery Schedule

6.3 Reports from JIRA

## **7. CODING & SOLUTIONING**

7.1 Feature 1

7.2 Feature 2

## **8. TESTING**

8.1 Test Cases

8.2 User Acceptance Testing

## **9. RESULTS**

9.1 Performance Metrics

## **10. ADVANTAGES & DISADVANTAGES**

## **11. CONCLUSION**

## **12.FUTURE SCOPE**

## **13.APPENDIX**

Source Code

GitHub & Project Demo Link

## **LIST OF FIGURES**

1. Empathy map canvas
2. Ideation
3. Brainstorming
4. Problem Solution Fit
5. Data Flow Diagram
6. Solution Architecture
7. Technical Architecture
8. JIRA Roadmap
9. Sprint 1
10. Sprint 2
11. Sprint 3
12. Sprint 4

## **LIST OF TABLES**

1. Proposed Solution
2. Functional Requirement
3. Problem statement Definition
4. Non-Functional Requirement
5. Components and Technologies
6. Application Characteristics
7. User Stories
8. Sprint Planning and Estimation
9. Sprint Delivery Schedule

# **1. INTRODUCTION**

## **1.1 PROJECT OVERVIEW**

This Project mainly focuses on applying Neural Networks to predict the Crude Oil Price. This decision helps us to buy crude oil at the proper time. Time series analysis is the best option for this kind of prediction because we are using the Previous history of crude oil prices to predict future crude oil. So we would be implementing RNN(Recurrent Neural Network) with LSTM(Long Short Term Memory) to achieve the task.

## **1.2 PURPOSE**

Oil demand is inelastic, therefore the rise in price is good news for producers because they will see an increase in their revenue. Oil importers, however, will experience increased costs of purchasing oil. Because oil is the largest traded commodity, the effects are quite significant. A rising oil price can even shift economic/political power from oil importers to oil exporters. The crude oil price movements are subject to diverse influencing factors.

## 2.LITERATURE SURVEY

Sl.no	Paper Title	Year of publication	Journal or Conference name	Authors	Theme of the paper	Inference
1.	Forecasting Crude Oil Price Using Event Extraction	2021	Shanghai, China	Jiangwei Liu, Xiaohong Huang	Crude oil Price Prediction.	Understanding the concepts of crude oil price using event extraction.
2.	Predictive Analytics for Crude Oil Price Using RNN-LSTM Neural Network	2020	Bandar Seri Iskandar, Malaysia	Norshakirah Aziz, Mohd Hafizul Afifi Abdullah, Ahmad Naqib Zaidi	Prediction of future crude oil price	Understanding the prediction model built based on the RNN-LSTM network
3.	Oil Price Forecast Using Deep Learning and ARIMA	2019	Taiyuan, China	Junhui Guo	Neural network model to make the prediction	Understanding Oil price forecast using deep learning and ARIMA.
4.	Crude Oil Price Forecasting: A Transfer Learning Based Analog Complexing Model	2012	Lanzhou, China	Jin Xiao, Changzheng He, Shouyang Wang	Predicting Crude oil Price using a Transfer based analog complexing model	Understanding the transfer learning based analog complexing model (TLAC)

## 2.1 EXISTING PROBLEM

Crude oil price fluctuations have a far reaching impact on global economies and thus price forecasting can assist in minimising the risks associated with volatility in oil prices. Price forecasts are very important to various stakeholders: governments, public and private enterprises, policymakers, and investors.

## 2.2 PROBLEM STATEMENT DEFINITION

<b>Problem Statement (PS)</b>	<b>I am (Customer)</b>	<b>I'm trying to</b>	<b>But</b>	<b>Because</b>	<b>Which makes me feel</b>
PS-1	Government Organisation	Predict accurate results of crude oil	It takes long duration	There are more number of relevant data	Tired
PS-2	Private Corporation	Crude oil prediction for company profit	It takes more amount of data	Prediction will be accurate when data is more	Disappointed



## 2.3 REFERENCES

- [1]Shuang Gao, Yalin Lei. "A new approach for crude oil price prediction based on stream learning", Geoscience Frontiers, 2017
- [2]Haykin S.  
Neural Networks: A Comprehensive Foundation (2nd edition), Prentice Hall (1999), p. 842
- [3]Lean Yu. "An EMD-Based Neural Network Ensemble Learning Model for World Crude Oil Spot Price Forecasting", Studies in Fuzziness and Soft Computing, 2008
- [4]Kulkarni S., Haidar I. Forecasting model for crude oil price using artificial neural networks and commodity future prices  
International Journal of Computer Science and Information Security, 2 (1) (2009)
- [5]Kaufmann R.K., Ullman B. Oil prices, speculation, and fundamentals: Interpreting causal relations among spot and futures prices Energy Economics, 31 (4) (2009), pp. 550-558

## 3. IDEATION & PROPOSED SOLUTION

### 3.1 EMPATHY MAP CANVAS

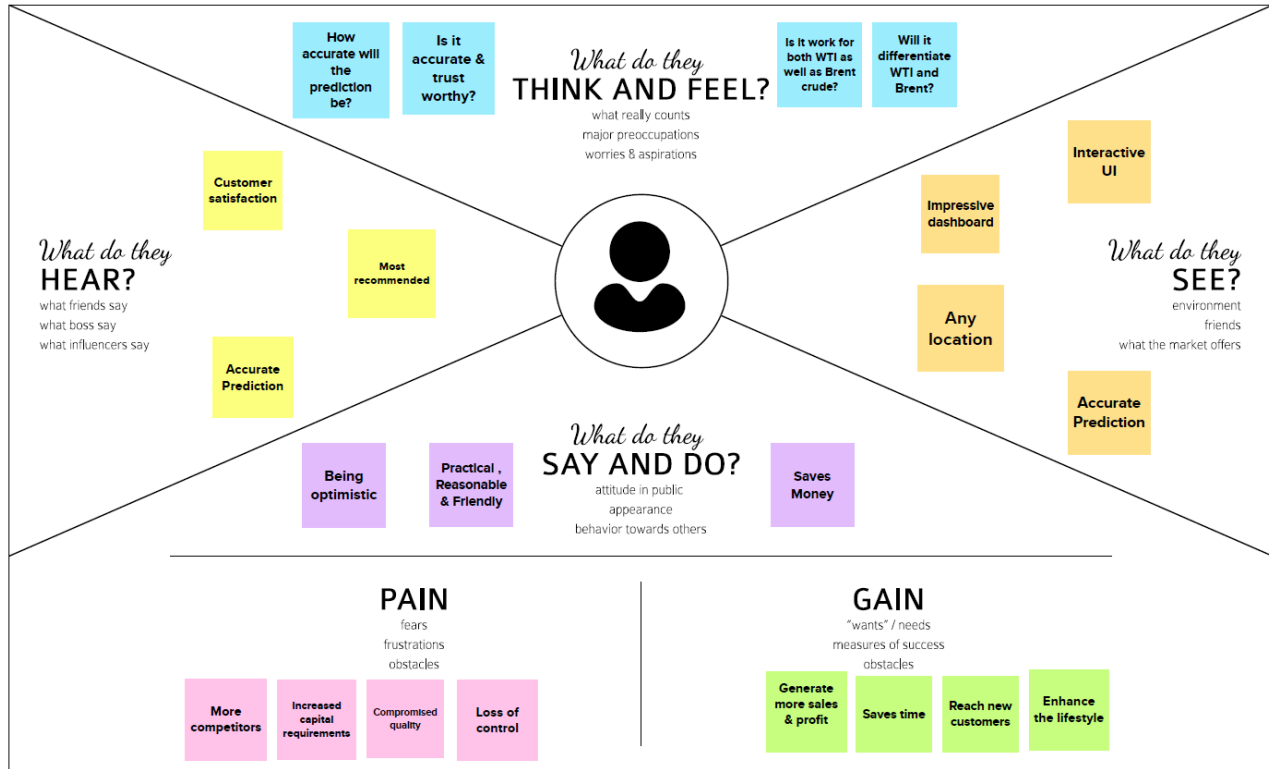


figure -1

### 3.2 IDEATION & BRAINSTORMING

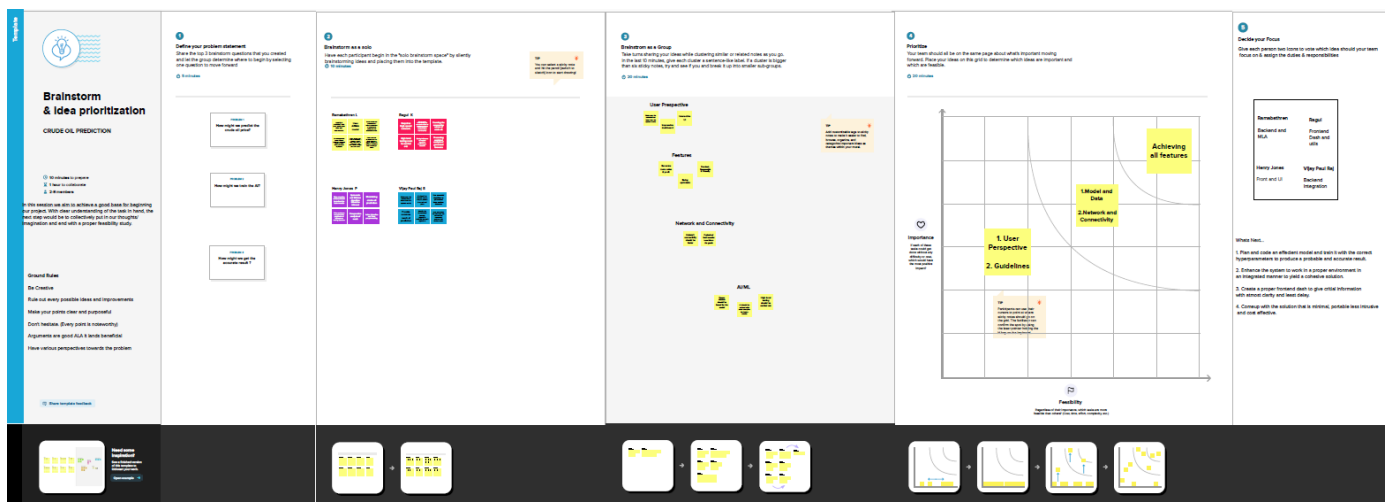


figure-2



figure -3

### 3.3 PROPOSED SOLUTION

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	To predict the number of orders according to the user's requirements.
2.	Idea / Solution description	To create an appropriate machine learning model to forecast the number of orders to gather raw materials for next ten weeks.
3.	Novelty / Uniqueness	Creative and animatory user interface, by which the user will be attracted and easy to use.
4.	Social Impact / Customer Satisfaction	By accurate predictions, wastage of food is reduced.
5.	Business Model (Revenue Model)	Advertisements, premium for large no. of users (organization)
6.	Scalability of the Solution	Large no. of users can access since the website is hosted in cloud .

## 3.4 PROBLEM SOLUTION FIT

Define CS, fit into CC	<div>1. CUSTOMER SEGMENT(S)<div>CS</div><div><ul style="list-style-type: none"><li>Crude Oil Investors</li><li>Petrol Bunk owners</li></ul></div></div>	<div>6. CUSTOMER CONSTRAINTS<div>CC</div><div>Any changes in the Crude oil market will have major impact on price.</div></div>	<div>5. AVAILABLE SOLUTIONS<div>AS</div><div>Useful for preventing risks and other issues.</div></div>	Explore AS, differentia
	<div>2. JOBS-TO-BE-DONE / PROBLEMS<div>—</div><div>The major issue in price prediction is the accurate results by which the crude oil traders and the market is not satisfied.</div></div>	<div>9. PROBLEM ROOT CAUSE<div>RC</div><div><ul style="list-style-type: none"><li>On Demand</li><li>Scarce Resource</li><li>Global exchange rate</li></ul></div></div>	<div>7. BEHAVIOUR<div>BE</div><div><ul style="list-style-type: none"><li>Price prediction helps in reducing the risks.</li><li>Necessary for Traders, investors and government.</li></ul></div></div>	
Identify strong TR & EM	<div>3. TRIGGERS<div>TR</div><div>Finding the impact of the problem and take necessary steps to solve the problem .</div></div>	<div>10. YOUR SOLUTION<div>SL</div><div>Using the Artificial Intelligence, Machine Learning algorithms we can predict the prices of the crude oil. It can be done from the equilibrium between demand and supply, wherein demand forecasts are usually made from GDP ,exchange rates and domestic prices, and supply is predicted from past production data and reserve data.</div></div>	<div>8. CHANNELS of BEHAVIOUR<div>CH</div><div>8.1 ONLINE Investors are happy by gaining huge profits.</div><div>8.2 OFFLINE Traders analyze demand and supply factors and take calculates positions. If their prediction comes true, traders close their position to book profits way before expiry.</div></div>	Identify strong TR & EM
	<div>4. EMOTIONS: BEFORE / AFTER<div>EM</div><div>Challenging to the extreme complexity and generate the solution to the problem.</div></div>			

figure-4

## 4. REQUIREMENT ANALYSIS

### 4.1 FUNCTIONAL REQUIREMENTS

Following are the functional requirements of the proposed solution.

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	Registration through Form Registration through Gmail
FR-2	User Confirmation	Confirmation via Email Confirmation via OTP
FR-3	Graph	Graphs are shown from obtaining the data from the datasets
FR-4	Notification	Notifications will be sent to the users about the crude oil prices
FR-5	Database	Storing the information of the users
FR-6	Support	Answers for the questions of the users will be shown through the live chats

## 4.2 NON-FUNCTIONAL REQUIREMENTS

Following are the non-functional requirements of the proposed solution.

FR No.	Non-Functional Requirement	Description
NFR-1	<b>Usability</b>	It can be used by many clients and it had a feature to change the theme of the dashboard
NFR-2	<b>Security</b>	As we are using user credentials for login and register which is stored in hash function gives the maximum security to the users
NFR-3	<b>Reliability</b>	It is reliable and gives the prefect prediction for the crude oil price
NFR-4	<b>Performance</b>	The performance of the application is fast and works in both mobile and web
NFR-5	<b>Availability</b>	The prediction will be available for every users and alert messages will be shown
NFR-6	<b>Scalability</b>	It is Scalable, Because we identified that the maximum load at which the system operates well

## 5. PROJECT DESIGN

### 5.1 DATA FLOW DIAGRAM

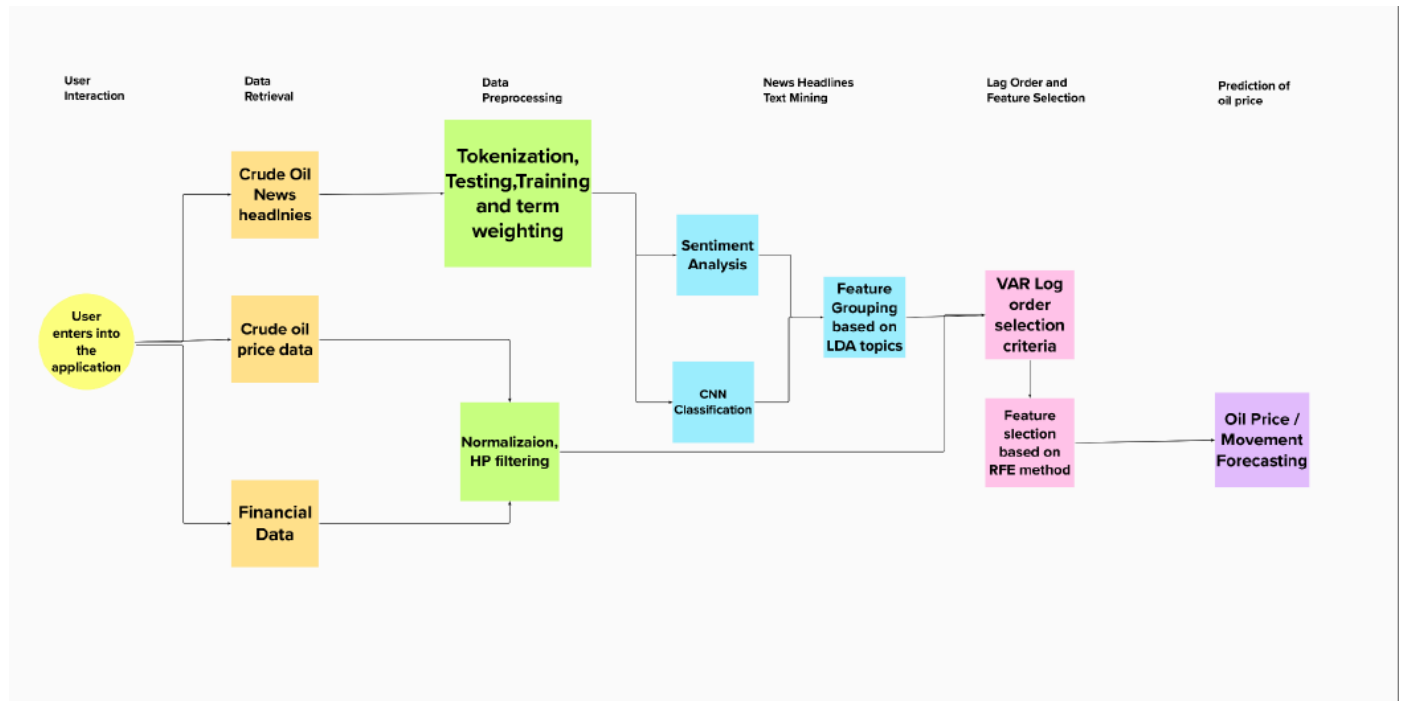


figure-5



## 5.2 SOLUTION & TECHNICAL ARCHITECTURE

### 5.2.1 SOLUTION ARCHITECTURE

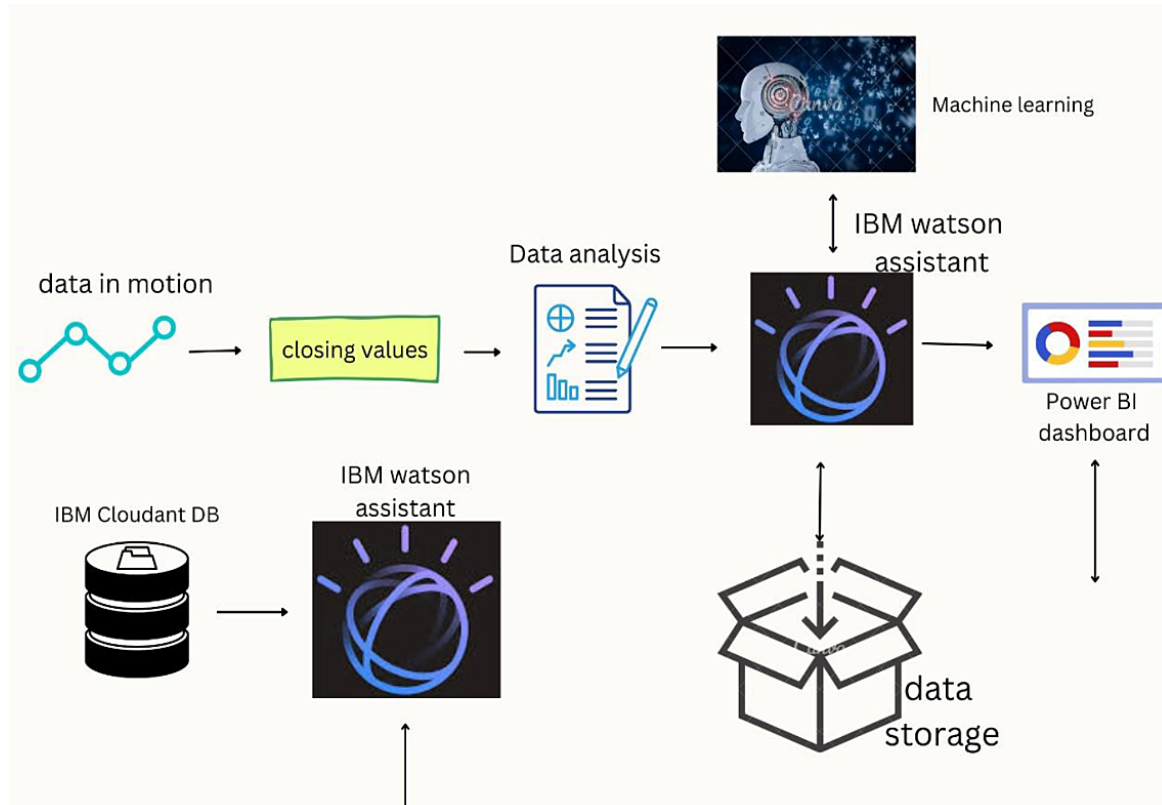


figure-6

## 5.2.2 TECHNICAL ARCHITECTURE

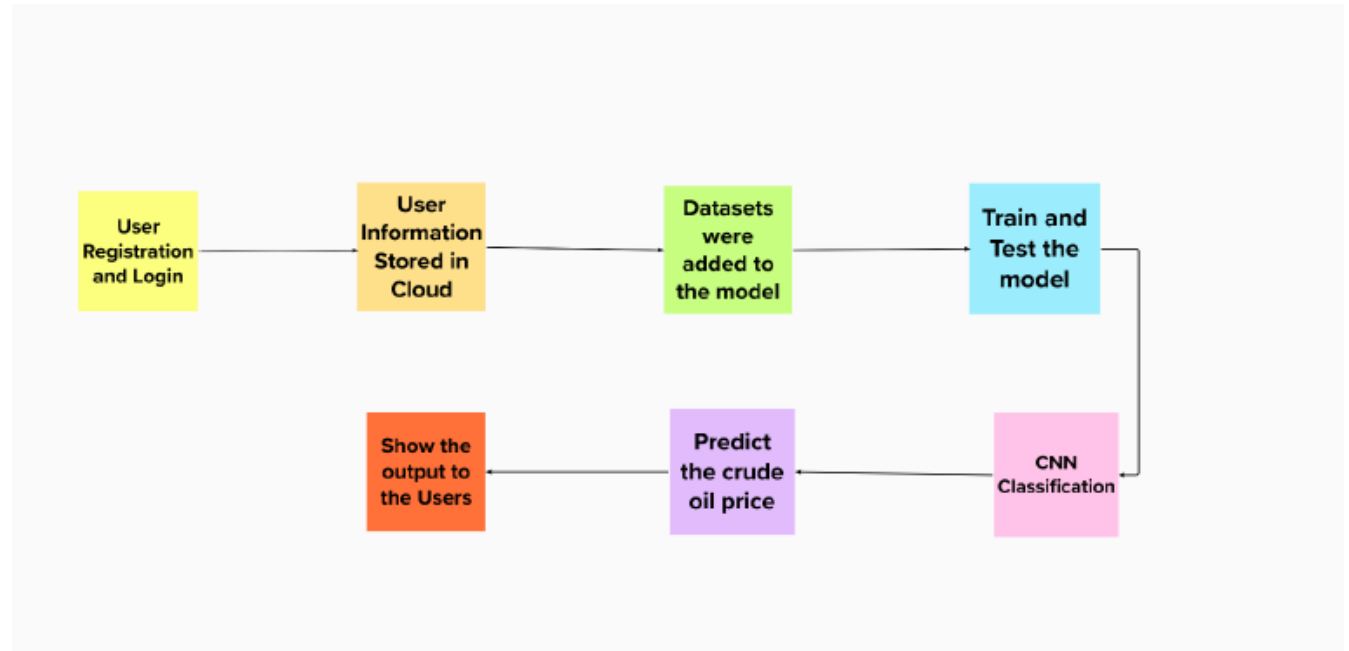


figure-7

**Table-1 : Components & Technologies:**

S.No	Component	Description	Technology
1.	User Interface	Web Application	HTML, CSS, JavaScript
2.	Application Logic-1	Logic for a process is Providing the datasets and gets the output for the price of crude oil	Python
3.	Database	Storing the Credentials of Users	MySQL, NoSQL, etc.
4.	Cloud Database	Use cloud storage for reducing the physical storage and access anywhere	IBM DB2, IBM Cloud etc.

5.	File Storage	Storing the files	IBM Block Storage or Other Storage Service or Local Filesystem
6.	External API-1	Using external API for connecting with external sources	IBM Weather API, etc.
7.	Machine Learning Model	Creating the Machine Learning Model for predicting the crude oil price	Object Recognition Model, etc.
8.	Infrastructure (Server / Cloud)	Application Deployment on Local System / Cloud	Local, Cloud Foundry etc.

**Table-2: Application Characteristics:**

S.No	Characteristics	Description	Technology
1.	Open-Source Frameworks	Google Chrome , Online Websites	Pandas, flask, numpy, tensorflow
2.	Security Implementations	User data will be stored according to CIA model	End to end encryption (SHA- 256)
3.	Scalable Architecture	IBM cloud and firebase both used for better performance in storage and authentication	IBM watson , Firebase, My sql
4.	Availability	Handle huge requests,avoid DDOS and XSS attack.	Coding is effecting and restrictive user access based on need
5.	Performance	Handle more than 1000 users to use server at a time.	Flask

## 5.3 USER STORIES

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Mobile user)	Application	USN-1	As a User Direct Open with Google Play Store App User Can Download the Crude Oil Price	I can access own decisions	High	Sprint-1
	Available Products	USN-2	As a User I can update the energy and the oil price instant	I can view the available products and I can use it,	High	Sprint-1
	Additional Features	USN-3	As a User access the additional features like viewing the charts, news, update the multiple colour themes	I can view then read the price prediction	Low	Sprint-2
	Expectations	USN-4	As a User I needs the perfect prediction of the crude oil price	I can expect accurate prediction	Medium	Sprint-1
	Login	USN-5	As a User I can log into the application by entering email & password	I can login with any social media account	High	Sprint-1
Customer (Web user)		USN-6	As a User I can view the crude oil price	I can view the price directly	High	Sprint-2
Customer Care Executive		USN-7	As a User I can view the history of price prediction of the crude oil	I can accept the terms	Medium	Sprint-1
Administrator		USN-8	As a Administrator I can predict the entire price of crude oil and add some extra features in the app	Show the result to the customer	High	Sprint-1

## 6. PROJECT PLANNING & SCHEDULING

### 6.1 SPRINT PLANNING & ESTIMATION

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Registration	USN-1	As a User Direct Open with Google Play Store App User Can Download the Crude Oil Price	13	High	Ramabathren L
Sprint-1	Login	USN-2	As a User I can log into the application by entering email & password	5	High	Ragul K
Sprint-2	Charts	USN-3	As a User access the additional features like viewing the charts, news, update the multiple colour themes	5	Low	Henry Jones P
Sprint-2	Prediction	USN-4	As a User I needs the perfect prediction of the crude oil price	13	Medium	Vijay Paul Raj S
Sprint-3	User Manual	USN-5	As a User I can see the way to predict the crude oil price	13	High	Ragul K

Sprint-3	Dashboard	USN-6	As a User I can view the crude oil price	5	High	Ramabathren L
Sprint-4	History	USN-7	As a User I can view the history of price prediction of the crude oil	5	Medium	Vijay Paul Raj S
Sprint-4	Admin Access	USN-8	As a Administrator I can predict the entire price of crude oil and add some extra features in the app	13	High	Henry Jones P

## 6.2 SPRINT DELIVERY SCHEDULE

<b>Sprint</b>	<b>Total Story Points</b>	<b>Duration</b>	<b>Sprint Start Date</b>	<b>Sprint End Date (Planned)</b>	<b>Story Points Completed (as on Planned End Date)</b>	<b>Sprint Release Date (Actual)</b>
Sprint-1	20	6 Days	24 Oct 2022	29 Oct 2022	18	29 Oct 2022
Sprint-2	20	6 Days	31 Oct 2022	05 Nov 2022	18	05 Nov 2022
Sprint-3	20	6 Days	07 Nov 2022	12 Nov 2022	18	12 Nov 2022
Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022	18	19 Nov 2022

6.3 REPORTS FROM JIRA

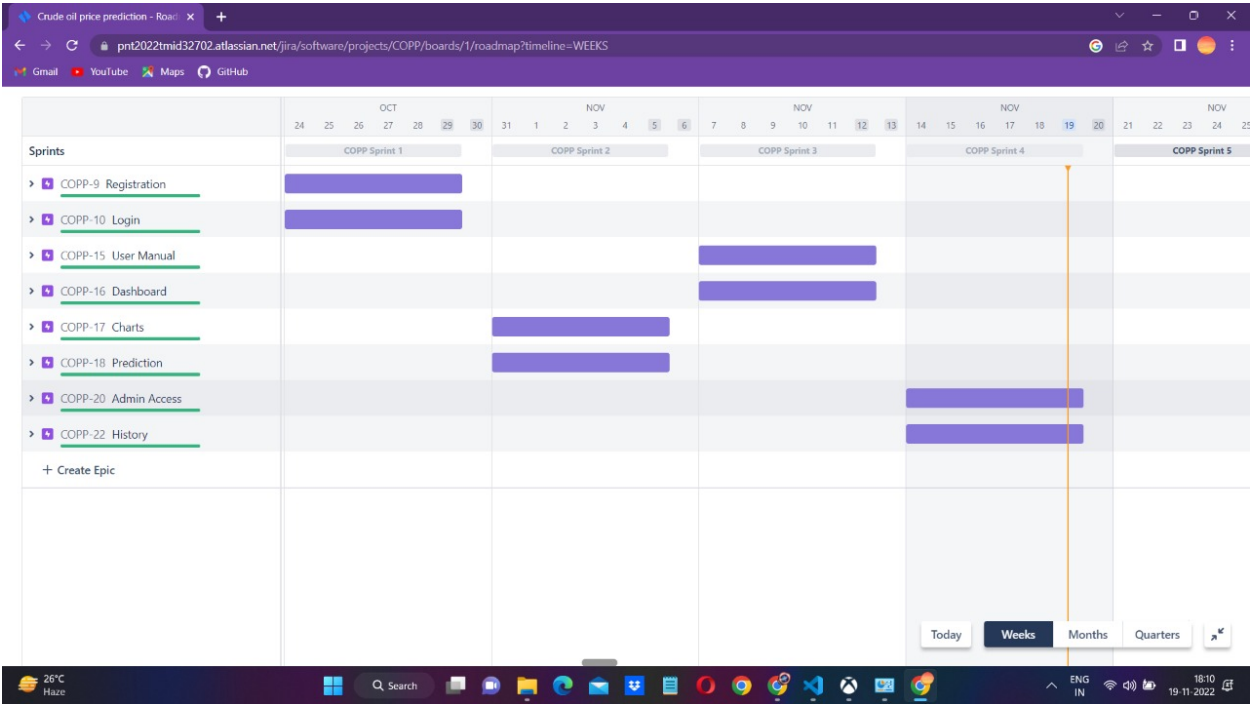


figure-8

SPRINT - 1

Date - October 24th, 2022 - October 29th, 2022

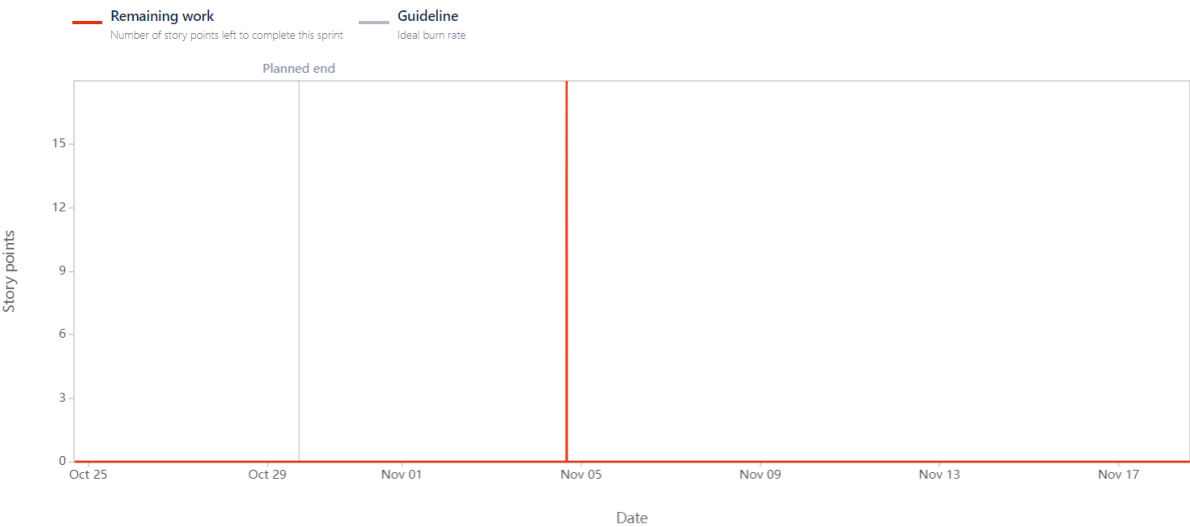


figure- 9

# SPRINT - 2

Date - October 31st, 2022 - November 5th, 2022

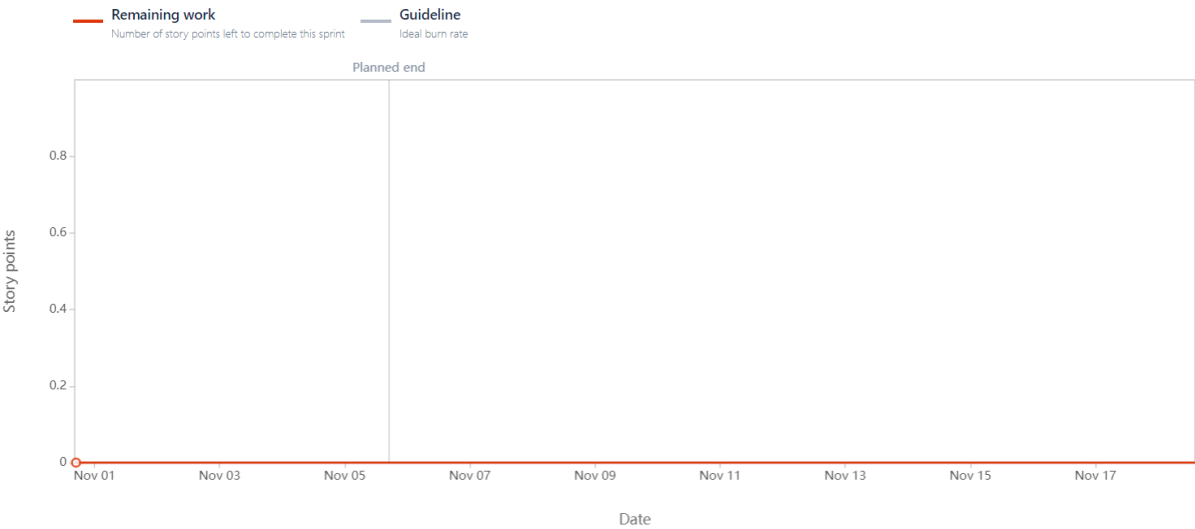


figure- 10

# SPRINT - 3

Date - November 7th, 2022 - November 12th, 2022

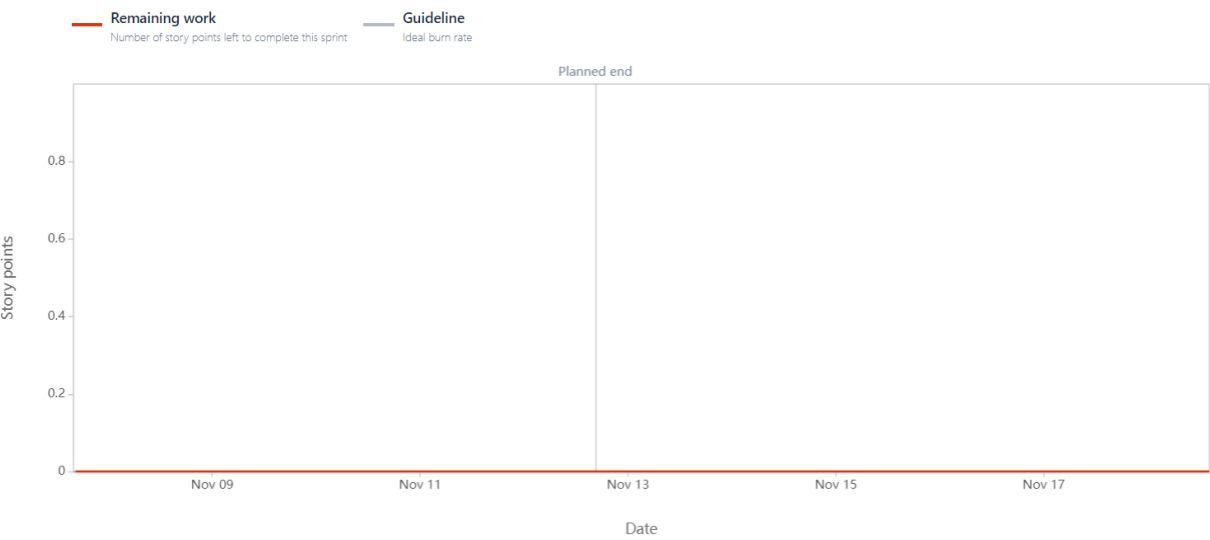


figure- 11



# SPRINT - 4

Date - November 14th, 2022 - November 19th, 2022

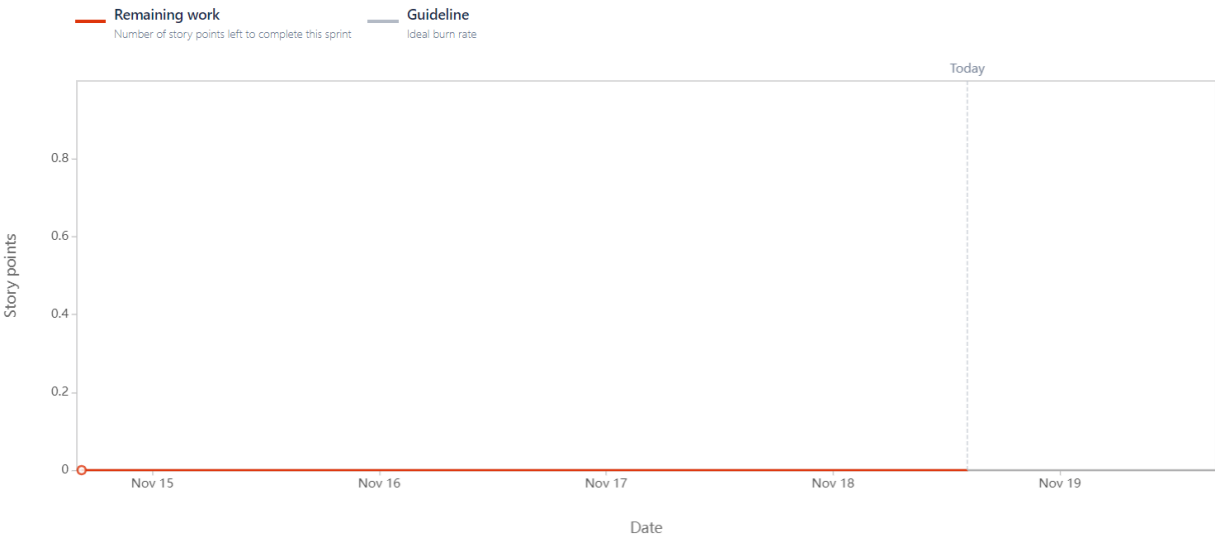


figure- 12

## 7. CODING & SOLUTIONING

### 7.1 FEATURE 1:

#### ADMIN ACCESS:

This page specifically for the admin who has the access to the list of logins and track of their histories.

#### CODE:

##### ADMIN.HTML:

```
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Admin Access</title>
</head>
<style>
  body{
    font-family: sans-serif;
    width: 100%;
    height: 100vh;
    background-image:
    linear-gradient(rgba(0,0,0,0.75),rgba(0,0,0,0.25)),url({{url_for('static',filename=
'crude_oil.jpg')}});
    background-size:cover;
    background-position: center;
    box-shadow: 0px 15px 20px rgba(0,0,0,0.75);
```

```
}  
h2{  
    color: aliceblue;  
    margin-left: 40%;  
  
}  
    nav {  
width: 100px;  
height:46px ;  
background:rgba(0,0,0,0.80);  
border-radius: 8px;  
    box-shadow: 0px 15px 20px rgba(0,0,0,0.75);  
left:35%;  
}  
nav a{  
    font-size: 13px;  
    color: #B9B7BD;  
    font-family: sans-serif;  
    border-radius: 50%;  
    text-decoration: none;  
    line-height: 45px;  
    position: relative;  
    z-index: 1;  
    display: inline-block;  
    text-align: center;  
    text-transform: uppercase;  
}  
nav .animation{  
    position: absolute;
```

```
height: 2px;
bottom: 0;
z-index: 0;
background: white;
border-radius: 8px;
transition: all .5s ease 0s;
}
a:nth-child(1){
width: 100px;
}
</style>
<body>
  <nav><a href=back>Back</a></nav>
  {% for details in admin %}
  <h2>{{ details }}</h2>
  {% endfor %}

</body>
</html>
```

## 7.2 FEATURE 2:

### USERMANUAL:

This page is to help the user who are new to this web application, This page gives you the complete study of how to work in this web application.

### CODE:

#### ABOUT.HTML:

```
<!DOCTYPE html>
```

```
<html lang="en">
```

```
  <head>
```

```
    <meta charset="UTF-8" />
```

```
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
```

```
    <title>about</title>
```

```
  </head>
```

```
  <style>
```

```
    *{
```

```
margin: 0;
```

```
padding: 0;
```

```
box-sizing: border-box;
```

```
}
```

```
body{
```

```
font-family: sans-serif;
```

```
width: 100%;
```

```
height: 100vh;
```

```
background-image: linear-
```

```
gradient(rgba(0,0,0,0.75),rgba(0,0,0,0.25)),url({{url_for('static',filename= 'crude_oil.jpg')}});
```

```
background-size:cover;
```

```
background-position: center;
}
.box{
width: 600px;
padding: 30px;
position: absolute;
top: 50%;
left: 50%;
border-radius: 20px;
transform: translate(-50%,-50%);
background: rgba(0,0,0,0.80);
text-align: center;
box-shadow: 0px 15px 20px rgba(0,0,0,0.75);

}
.box h1
{
color: white;
text-transform: uppercase;
font-weight: 700;

}
.usr input[type="text"]
{
border: 0;
background: none;
display: block;
margin: 20px auto;
text-align: center;
```

```
border: 2px solid #0367fd;
border-color: #B9B7BD;
padding: 14px 10px;
width: 250px;
outline: none;
color: white;
border-radius: 18px;
transition: 0.25s;
```

```
}
```

```
.box input[type="text"]:focus{
width: 270px;
border-color:#EEEDE7;
```

```
}
```

```
.pwd input[type="password"]{
border: 0;
background: none;
display: block;
margin: 20px auto;
text-align: center;
border: 2px solid #0367fd;
border-color: #B9B7BD ;
padding: 14px 10px;
width: 250px;
outline: none;
color: white;
border-radius: 18px;
```

```
transition: 0.25s;
}
.box input[type="password"]:focus{
width: 270px;
border-color:#EEEEDE7;

}
.email input[type="email"]{
border: 0;
background: none;
display: block;
margin: 20px auto;
text-align: center;
border: 2px solid #0367fd;
border-color: #B9B7BD ;
padding: 14px 10px;
width: 250px;
outline: none;
color: white;
border-radius: 18px;
transition: 0.25s;
}
.box input[type="password"]:focus{
width: 270px;
border-color:#EEEEDE7;

}
.box input[type="submit"]{
```



```
border: 0;
background: none;
display: block;
margin: 10px auto;
text-align: center;
border: 2px solid #B9B7BD;
padding: 10px 10px;
width: 100px;
outline:none;
color: #B9B7BD;
border-radius:18px;
transition:0.25s;
cursor: pointer;
}

.box input[type="submit"]:hover{
background: #EEEEDE7;
color: black;
font-weight: 700;
box-shadow:0px 15px 20px rgba(0,0,0,1.50);
}

.link a{
border: 0;
background: none;
display: block;
margin: 10px auto;
text-align: center;

padding: 14px 10px;
width: 100px;
```

outline:none;

color: white;

transition:0.25s;

cursor: pointer;

text-decoration: none;

}

.link a:hover{

color:#B9B7BD;

font-weight: 700;

text-decoration: underline;

}

nav {

width: 100px;

height:46px ;

background:rgba(0,0,0,0.80);

border-radius: 8px;

box-shadow: 0px 15px 20px rgba(0,0,0,0.75);

left:34%;

}

nav a{

font-size: 13px;

color: #B9B7BD;

font-family: sans-serif;

border-radius: 50%;

text-decoration: none;

line-height: 45px;

```
position: relative;
z-index: 1;
display: inline-block;
text-align: center;
text-transform: uppercase;
}
nav .animation{
position: absolute;
height: 2px;
bottom: 0;
z-index: 0;
background: white;
border-radius: 8px;
transition: all .5s ease 0s;
}
a:nth-child(1){
width: 100px;
}
nav .start-register,a:nth-child(1):hover~.animation{
width: 90px;
left: 7px;

}
a:nth-child(2){
width: 100px;
}
nav .start-game,a:nth-child(2):hover~.animation{
width: 60px;
left: 125px;
```

```
}
a:nth-child(3){
width: 100px;}
nav .start-blog,a:nth-child(3):hover~.animation{
width: 60px;
left: 230px;

}
a:nth-child(4){
width: 100px;
}
nav .start-help,a:nth-child(4):hover~.animation{
width: 60px;
left: 335px;

}
p{
color: #B9B7BD;
}
<body>

</style>
<nav><a href=back>Back</a></nav>

<div class="box">
<h1>CRUDE OIL PRICE PREDICTION</h1>

<br>
```

<p>Crude oil is one of the most important commodities in the world, accounting for one-third of global energy consumption.</P><br>

<P>It is a starting material for most of the products that we use in everyday life, ranging from transportation fuels to plastics.</P><br>

<p>Crude oil price fluctuations have a far reaching impact on global economies and thus price forecasting can assist in minimising the risks associated with volatility in oil prices. Price forecasts are very important to various stakeholders: governments, public and private enterprises, policymakers, and investors.</P><br>

<p>According to economic theory, the price of crude oil should be easily predictable from the equilibrium between demand and supply, wherein demand forecasts are usually made from GDP, exchange rates and domestic prices, and supply is predicted from past production data and reserve data.</P><br>

<p>Predicting demand for oil is usually straightforward, however supply is heavily affected by political activity such as cartelisation by OPEC to regulate prices, technological advances leading to the extraction of higher amounts of oil, and wars and other conflicts which can affect supply unpredictably. </P><br>

<p>The prices of crude oil for the last 10 days is obtained and is used to predict the crude oil price for today.</p>

<div class="link">

<a href="/">Home</a>

</div>

</div>

</body>

</html>

## **8. TESTING**

### **8.1 TEST CASES**

## **8.2 USER ACCEPTING TESTING**

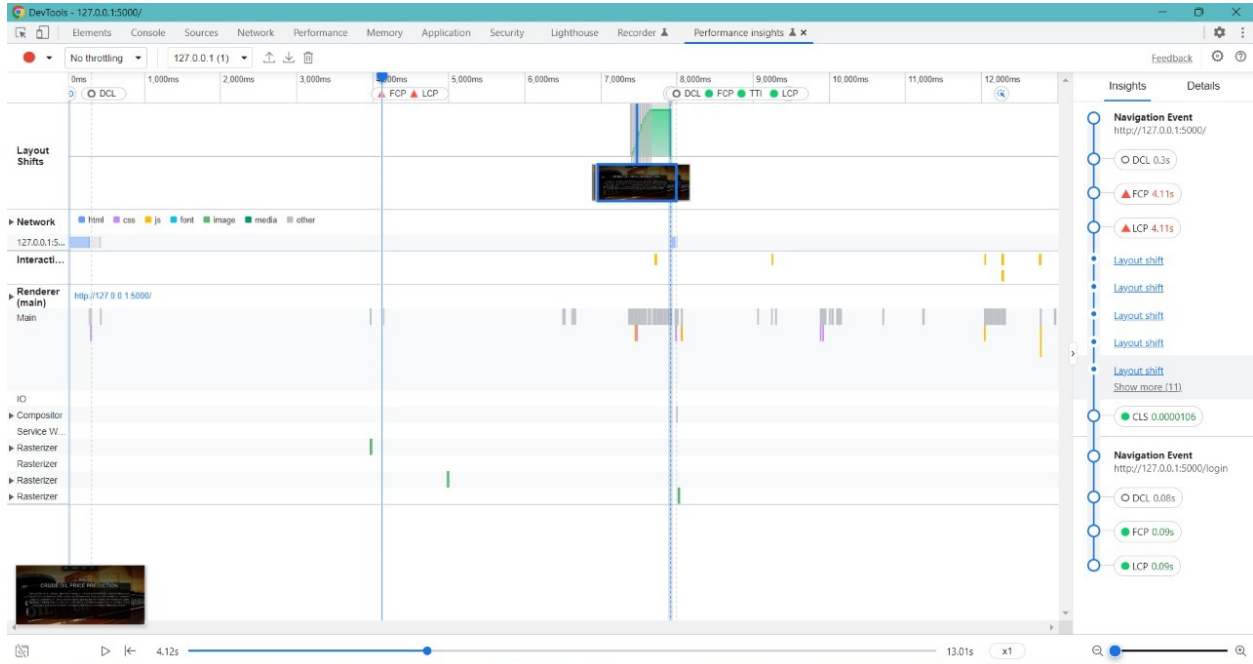




## 9.RESULTS

Thus, We have trained our machine learning model in IBM watson studio have successfully built an web application for predicting the crude oil price for next day.

### 9.1 PERFORMANCE METRICES



## **10. ADVANTAGES & DISADVANTAGES**

### **ADVANTAGES:**

Crude oil price fluctuations have a far reaching impact on global economies and thus price forecasting can assist in minimising the risks associated with volatility in oil prices.

### **DISADVANTAGES:**

Several oil companies have invested crores of dollars in oil exploration by taking bank loans, chances of default, if these companies are not able to get fair price for their commodity

## **11. CONCLUSION**

According to economic theory, the price of crude oil should be easily predictable from the equilibrium between demand and supply, wherein demand forecasts are usually made from GDP, exchange rates and domestic prices, and supply is predicted from past production data and reserve data.

## **12.FUTURE SCOPE**

In Future we are planning to add features like premium membership users by paying the money and a chatbot in the website ,were the users can chat with the admin for any queries or any improvements to the website.

## 13.APPENDIX

### SOURCE CODE:

#### APP.PY:

```
import numpy as np
import os
os.environ['TF_CPP_MIN_LOG_LEVEL'] = '2'
from flask import Flask,render_template,request,redirect,session,url_for,flash
from tensorflow.keras.models import load_model
# import ibm_db
# from .connect import get_db_connection
import ibm_db
def get_db_connection():
    try:
        conn = ibm_db.connect("DATABASE=BLUDB;\ HOSTNAME=ba99a9e6-d59e-4883-8fc0-d6a8c9f7a08f.c1ogj3sd0tgu0lqde00.databases.appdomain.cloud;\
        PORT=31321;\
        Security=SSL;\
        SSLServerCertificate=DigiCertGlobalRootCA.crt;\
        UID=wvm68663;\
        PWD=o8ZTWcmctkTiXpeR;","","")
        print("Connected to DB")
        return conn
    except:
        print("error r while connecting ",ibm_db.conn_errormsg())
        return 0
con=get_db_connection()
```

```
app=Flask(__name__)
app.secret_key="123"
model=load_model('crude_oil.h5',)

name1=input();

@app.route('/')
def home():
    return render_template("index.html")

@app.route('/about')
def about():
    return render_template("about.html")

@app.route('/predict')
def home2():
    return render_template("web.html")

@app.route('/back',methods=['GET','POST'])
def back():
    return render_template("web.html")

@app.route('/admin',methods=['GET','POST'])
def admin():
    # if request.method=='POST':
    email=session["email"]
    password=session['password']
```

```

# qry="select * from USER where email=? AND password=?"
# stmt=ibm_db.prepare(con,qry)
# ibm_db.bind_param(stmt,1,email)
# ibm_db.bind_param(stmt,2,password)
# ibm_db.execute(stmt)
# resp=ibm_db.fetch_assoc(stmt)
# print("resp - ",resp)
print(email)
print(password)
if (email=="admin@gmail.com"):
    if(password=="admin"):
        qry="select NAME from USER"
        stmt=ibm_db.prepare(con,qry)
        resp=ibm_db.execute(stmt)
        var=ibm_db.fetch_assoc(stmt)
        list1=[]
        while var!=False:
            var=ibm_db.fetch_assoc(stmt)
            list1.append(var)
            var=ibm_db.fetch_assoc(stmt)
        print("resp - ",resp)
        print(list1)
        list1.remove(False)
        print(list1)
        return render_template("admin.html",admin=list1)
return render_template("web.html")

```

```
@app.route('/history')
```

```

def history():
    name=session["name"]
    qry="select PRICE from HISTORY where NAME=?"
    stmt=ibm_db.prepare(con,qry)
    ibm_db.bind_param(stmt,1,name)
    resp3=ibm_db.execute(stmt)
    var=ibm_db.fetch_assoc(stmt)
    price1=[]
    while var!=False:
        var=ibm_db.fetch_assoc(stmt)
        price1.append(var['PRICE'])
        var=ibm_db.fetch_assoc(stmt)
    # prices.remove(False)
    print(resp3)
    print(price1)
    res1 = [eval(i) for i in price1]
    return render_template("history.html",history=res1)

```

```

@app.route('/graph')

```

```

def graph():
    name=session["name"]
    session["name"]=name
    qry="select PRICE from HISTORY where NAME=?"
    stmt=ibm_db.prepare(con,qry)
    ibm_db.bind_param(stmt,1,name)
    resp3=ibm_db.execute(stmt)
    var=ibm_db.fetch_assoc(stmt)
    price=[]
    while var!=False:

```



```

        var=ibm_db.fetch_assoc(stmt)
        price.append(var['PRICE'])
        var=ibm_db.fetch_assoc(stmt)
# prices.remove(False)
print(resp3)
print(price)
res = [eval(i) for i in price]
res2=len(res)
return render_template("graph.html",history=res,history1=res2)

@app.route('/register',methods=['GET','POST'])
def register():
    if request.method=='POST':
        try:
            name=request.form['name']
            email=request.form['email']
            password=request.form['password']
            qry="insert into USER(name,email,password)values(?,?,?)"
            stmt=ibm_db.prepare(con,qry)
            ibm_db.bind_param(stmt,1,name)
            ibm_db.bind_param(stmt,2,email)
            ibm_db.bind_param(stmt,3,password)
            resp2=ibm_db.execute(stmt)
            print(resp2)
            flash("Record Added Successfully","success")
            return render_template("login.html")
        except:
            flash("Error in Insert Operations","danger")
    # finally:

```

```
        # return render_template("index.html")

        # con.close()

    return render_template("register.html")
```

```
@app.route('/login',methods=['GET','POST'])
```

```
def login():
```

```
    if request.method=='POST':
        email=request.form['email']
        password=request.form['password']
        qry="select * from USER where email=? AND password=?"
        stmt=ibm_db.prepare(con,qry)
        ibm_db.bind_param(stmt,1,email)
        ibm_db.bind_param(stmt,2,password)
        ibm_db.execute(stmt)
        resp=ibm_db.fetch_assoc(stmt)
        print("resp - ",resp)
        if resp:
            session["name"]=resp['NAME']
            session["u_id"]=resp['USER_ID']
            session["email"]=resp['EMAIL']
            session["password"]=resp["PASSWORD"]
            return render_template("web.html")
        else:
            flash("Usernaem and Password Mismatch","danger")
            return redirect("login")
```

```
return render_template("login.html")
```

```

@app.route('/prediction',methods=['POST'])
def prediction():
    x_input=str(request.form['year'])
    x_input=x_input.split(',')
    print(x_input)
    for i in range(0,len(x_input)):
        x_input[i]=float(x_input[i])
    print(x_input)
    x_input=np.array(x_input).reshape(1,-1)
    temp_input=list(x_input)
    temp_input=temp_input[0].tolist()
    lst_output=[]
    n_steps=10
    i=0
    while(i<1):
        if(len(temp_input)>10):
            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))

            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
```

name=session["name"]

u\_id=session["u\_id"]

qry="insert into HISTORY(USER\_ID,NAME,PRICE) values(?,?,?)"

stmt=ibm\_db.prepare(con,qry)

ibm\_db.bind\_param(stmt,1,u\_id)

ibm\_db.bind\_param(stmt,2,name)

ibm\_db.bind\_param(stmt,3,str(lst\_output[0][0]))

resp2=ibm\_db.execute(stmt)

print(resp2)

```
        return render_template("web.html",showcase="The next day predicted value
is:'+str(lst_output[0][0]))
```

if \_\_name\_\_=='\_\_main\_\_':

app.run(debug=True,port=5000)

## **NEW.PY:**

```
import requests
```

```
# NOTE: you must manually set API_KEY below using information retrieved from your IBM Cloud account.
```

```
API_KEY = "BS2SMFqDeLGEOOnI70kz_nIZbW3T3Zpz3ZxLJUdmoQU1"
```

```
token_response = requests.post('https://iam.cloud.ibm.com/identity/token', data={"apikey":
```

```
API_KEY, "grant_type": 'urn:ibm:params:oauth:grant-type:apikey'})
```

```
mltoken = token_response.json()["access_token"]
```

```
header = {'Content-Type': 'application/json', 'Authorization': 'Bearer ' + mltoken}
```

```
# NOTE: manually define and pass the array(s) of values to be scored in the next line
```

```
payload_scoring = {"input_data": [{"field": ["Date","Closing Value"], "values": [[0,0]]}]}
```

```
response_scoring = requests.post('https://us-south.ml.cloud.ibm.com/ml/v4/deployments/a993a66b-e772-44e9-bee8-0f2c7c0134f1/predictions?version=2022-11-17', json=payload_scoring,
```

```
headers={'Authorization': 'Bearer ' + mltoken})
```

```
print("Scoring response")
```

```
print(response_scoring.json())
```

## **INDEX.HTML:**

```
<!DOCTYPE html>
```

```
<!--
```

```
Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license
```

```
Click nbfs://nbhost/SystemFileSystem/Templates/ClientSide/html.html to edit this template
```

```
-->
```

```
<html>
```

```
<head>
```

```
<title>Home</title>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

</head>

<body>

  <style>

body{

  font-family: sans-serif;

  width: 100%;

  height: 100vh;

backgroundimage:lineargradient(rgba(0,0,0,0.75),rgba(0,0,0,0.25)),url({ {url_for('static',filenam

e= 'crude_oil.jpg')}});

  background-size:cover;

  background-position: center;

  box-shadow: 0px 15px 20px rgba(0,0,0,0.75);

}

nav {

  position: relative;

  width: 430px;

  height:46px ;

  background:rgba(0,0,0,0.80);

  border-radius: 8px;

  box-shadow: 0px 15px 20px rgba(0,0,0,0.75);

left:35%;

}

nav a{

  font-size: 13px;

  color: #B9B7BD;

  font-family: sans-serif;
```

```
border-radius: 50%;
text-decoration: none;
line-height: 45px;
position: relative;
z-index: 1;
display: inline-block;
text-align: center;
text-transform: uppercase;
}
nav .animation{
  position: absolute;
  height: 2px;
  bottom: 0;
  z-index: 0;
  background: white;
  border-radius: 8px;
transition: all .5s ease 0s;
}
a:nth-child(1){
  width: 100px;
}
nav .start-register,a:nth-child(1):hover~.animation{
  width: 60px;
  left: 20px;
}
a:nth-child(2){
  width: 100px;
}
```

```
nav .start-game,a:nth-child(2):hover~.animation{
    width: 75px;
    left: 117px;

}
a:nth-child(3){
    width: 100px;}
nav .start-blog,a:nth-child(3):hover~.animation{
    width: 60px;
    left: 230px;

}
a:nth-child(4){
    width: 100px;

}
nav .start-help,a:nth-child(4):hover~.animation{
    width: 65px;
    left: 333px;
}
a:nth-child(5){
    width: 100px;

}
nav .start-help,a:nth-child(5):hover~.animation{
    width: 63px;
    left: 437px;
```



```
}  
.box{  
  width: 300px;  
  padding: 30px;  
  position: absolute;  
  top: 50%;  
  left: 50%;  
  border-radius: 20px;  
  transform: translate(-50%,-50%);  
  background: rgba(0,0,0,0.80);  
  text-align: center;  
  box-shadow: 0px 15px 20px rgba(0,0,0,0.75);  
}  
.box h1  
{  
  color: white;  
  text-transform: uppercase;  
  font-weight: 700;  
}  
.text{  
  margin-left: 130px;  
  margin-top:150px;  
  bottom: 50%;  
  width: 80%;  
  height: 60%;  
  background: rgba(0,0,0,0.80);  
  text-align: center;  
  color: gray;  
  border-radius: 2%;
```

```
font-size:30px;
font-family: sans-serif;
}
input[type="submit"]{

border: 0;
background: none;
display: block;
margin: 10px auto;
text-align: center;
border: 2px solid #B9B7BD;
padding: 10px 10px;
width: 100px;
outline:none;
color: #B9B7BD;
border-radius:18px;
transition:0.25s;
cursor: pointer;
margin-left: 45%;
margin-top: 3%;
}
input[type="submit"]:hover{
background: #EEEEDE7;
color: black;
font-weight: 700;
box-shadow:0px 15px 20px rgba(0,0,0,1.50);
}
</style>
<nav>
```

```

    <a href="/">Home</a>
    <a href="register">Register</a>
    <a href="login">Login</a>
    <a href="about">about</a>
    <div class="animation start-home"></div>
</nav>
<div class="text">
    <h1>CRUDE OIL PRICE PREDICTION</h1>
    Demand for oil is inelastic, therefore the rise in price is good news for producers because
    they will see an increase in their revenue. Oil importers, however, will experience increased costs
    of purchasing oil. Because oil is the largest traded commodity, the effects are quite significant. A
    rising oil price can even shift economic/political power from oil importers to oil exporters. The
    crude oil price movements are subject to diverse influencing factors
</div>
</body>
</html>

```

## WEB.HTML:

```

<html>
<meta charset="utf-8">
<meta name="viewport" content="width=device-width, initial-scale=1">
<title>Prediction</title>
<style>
    *{
margin: 0;
padding: 0;
box-sizing: border-box;
}
body{

```

```
    font-family: sans-serif;
width: 100%;
height: 100vh;
background-image: linear-gradient(rgba(0,0,0,0.75),rgba(0,0,0,0.25)),url({ {url_for('static',filename= 'crude_oil.jpg')}});
background-size: cover;
background-position: center;

}

.box{
width: 400px;
padding: 30px;
position: absolute;
top: 50%;
left: 50%;
border-radius: 20px;
transform: translate(-50%,-50%);
background: rgba(0,0,0,0.80);
text-align: center;
box-shadow: 0px 15px 20px rgba(0,0,0,0.75);

}

.box h1
{
color: white;
text-transform: uppercase;
font-weight: 700;
```

```
}  
.usr input[type="text"]  
{  
border: 0;  
background: none;  
display: block;  
margin: 20px auto;  
text-align: center;  
border: 2px solid #0367fd;  
border-color: #B9B7BD;  
padding: 14px 10px;  
width: 250px;  
outline: none;  
color: white;  
border-radius: 18px;  
transition: 0.25s;  
  
}  
.box input[type="text"]:focus{  
width: 270px;  
border-color:#EEEDE7;  
  
}  
  
.days input[type="text"]{  
border: 0;  
background: none;  
display: block;  
margin: 20px auto;
```

```
text-align: center;
border: 2px solid #0367fd;
border-color: #B9B7BD ;
padding: 14px 10px;
width: 250px;
outline: none;
color: white;
border-radius: 18px;
transition: 0.25s;
}
.box input[type="password"]:focus{
width: 270px;
border-color:#EEEDE7;

}
.mobile input[type="number"]{
border: 0;
background: none;
display: block;
margin: 20px auto;
text-align: center;
border: 2px solid #0367fd;
border-color: #B9B7BD ;
padding: 14px 10px;
width: 250px;
outline: none;
color:#B9B7BD;
border-radius: 18px;
transition: 0.25s;
```

```
}  
.box input[type="password"]:focus{  
width: 270px;  
border-color:#EEEEDE7;
```

```
}  
.box input[type="submit"]{
```

```
border: 0;  
background: none;  
display: block;  
margin: 10px auto;  
text-align: center;  
border: 2px solid #B9B7BD;  
padding: 10px 10px;  
width: 100px;  
outline:none;  
color: #B9B7BD;  
border-radius:18px;  
transition:0.25s;  
cursor: pointer;
```

```
}  
.box input[type="submit"]:hover{  
background:#B9B7BD;  
color: black;
```

```
font-weight: 700;
box-shadow:0px 15px 20px rgba(0,0,0,1.50);
}
nav {
    position: relative;
    top: 2%;
    width: 700px;
    height:46px ;
    background:rgba(0,0,0,0.80);
    border-radius: 8px;
    box-shadow: 0px 15px 20px rgba(0,0,0,0.75);
    left:28%;
}
nav a{
    font-size: 13px;
    color: #B9B7BD;
    font-family: sans-serif;
    border-radius: 50%;
    text-decoration: none;
    line-height: 45px;
    position: relative;
    z-index: 1;
    display: inline-block;
    text-align: center;
    text-transform: uppercase;
}
nav .animation{
    position: absolute;
    height: 2px;
```



```
    bottom: 0;
    z-index: 0;
    background:#B9B7BD;
    border-radius: 8px;
    transition: all .5s ease 0s;
  }
  a:nth-child(1){
    width: 100px;
  }
  nav .start-register,a:nth-child(1):hover~.animation{
    width: 60px;
    left: 20px;

  }
  a:nth-child(2){
    width: 100px;
  }
  nav .start-game,a:nth-child(2):hover~.animation{
    width: 60px;
    left: 125px;

  }
  a:nth-child(3){
    width: 100px;}
  nav .start-blog,a:nth-child(3):hover~.animation{
    width: 60px;
    left: 230px;

  }
```

```
a:nth-child(4){
  width: 120px;

}
nav .start-help,a:nth-child(4):hover~.animation{
  width: 60px;
  left: 343px;

}
a:nth-child(5){
  width: 120px;

}
nav .start-help,a:nth-child(5):hover~.animation{
  width: 60px;
  left: 465px;

}
a:nth-child(6){
  width: 120px;

}
nav .start-help,a:nth-child(6):hover~.animation{
  width: 60px;
  left: 600px;

}
</style>
```

```

<body><nav>
    <a href="login"> </a>
    <a href="about">about</a>
    <a href="/">Logout</a>
    <a href="history">history</a>
    <a href="graph">Graph</a>
    <a href="admin">Admin</a>
    <div class="animation start-home"></div>
</nav>
<br>
    <center><div><font color="Powderblue" font-family="sans-serif" size=8 ><b><br>Crude
Oil Price Prediction</b></font></div></center>

<br><br><br><br>
<div class="box">
    <form class="main" name="year" action="/prediction" method="post">
        <br>
        <h1>WELCOME {{session["name"]}}</h1>
        <fontsize=20><divclass=days><inputtype="text" name="year" placeholder="Enter
previous 10days price"/></font></div>
        <center><input type="submit" class="logbtn" value="Predict"></center>
        <h1></h1>
        <divclass="bor"><b><fontcolor="white"size=5><br>{{ showcase }}</font></b></div>
    </form>
</div>
<div>
</div>
</body>
</html>

```

## **GITHUB & PROJECT DEMO LINK:**

<https://github.com/IBM-EPBL/IBM-Project-16964-1659626033.git>

<https://youtu.be/nm80Yop6gfc>