# Crude Oil Price Prediction

## Team ID : PNT2022TMID14370

## Team Members :

- Niranjan Jp
- Rex Milton
- Mohammed Umar
- Praveen Kumar AM
- Madhavan

# 1.Introduction

## 1.1 Project Overview

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.

This Guided 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

Crude oil is amongst the most important resources in today's world, it is the chief fuel and its cost has a direct effect on the global habitat, our economy and oil exploration, exploitation and other activities. Prediction of oil prices has become the need of the hour, it is a boon to many large and small industries, individuals and the government.

This Guided 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

# 2. Literature Survey

## 2.1 Existing Problem

Although many methods and models have been developed for predicting crude oil prices, it remains one of the most challenging forecasting problems due to the high volatility of oil prices. In this project, we propose a novel approach for crude oil price prediction based on a new machine learning paradigms and Neural Network Concept . The main advantage of our project is that the prediction model can capture the changing pattern of oil prices since the model is continuously updated whenever new oil price data are available, with very small constant overhead.

## 2.2 References


[1] Kaufmann, R. K., & Ullman, B. (2009). Oil prices, speculation, and fundamentals: Interpreting causal relations among spot and futures prices. Energy Economics, 31(4), 550–558.

[2] Shobhit Nigam. "Chapter 84 Single Multiplicative Neuron Model in Reinforcement Learning" , Springer Science and Business Media LLC, 2019

[3] "Harmony Search and Nature Inspired Optimization Algorithms" , Springer Science and Business Media LLC, 2019

[4] Shuang Gao, Yalin Lei. "A new approach for crude oil price prediction based on stream learning" , Geoscience Frontiers, 2017

[5] Ramakanta Mohanty. "Software Reliability Prediction Using Group Method of Data Handling" , Lecture Notes in Computer Science, 2009

[6] Kulkarni, S., Haidar, I., 2009. Forecasting model for crude oil price using artificial neural networks and commodity future prices. International Journal of Computer Science and Information Security 2 (1).

[7] Haykin, S. (1999). Neural Networks: A Comprehensive Foundation, 2nd edition, Prentice Hall, 842 pages

[8] Onur Dursun. "3 Methodology" , Walter de Gruyter GmbH, 2014

[9] Lakshmanan, Indhurani, and Subburaj Ramasamy. "An Artificial Neural-Network Approach to Software Reliability Growth Modeling" , Procedia Computer Science, 2015.

[10] Haykin, S. (2009). Neural Networks and Learning Machines, 3rd edition, Pearson, 938 pages

[11] N. Raj Kiran, V. Ravi. "Software reliability prediction by soft computing techniques" , Journal of Systems and Software, 2008

[12] Lean Yu. "An EMD-Based Neural Network Ensemble Learning Model for World Crude Oil Spot Price Forecasting" , Studies in Fuzziness and Soft Computing, 2008

[13] Xin Wang, Ji Wu, Chao Liu, Senzhang Wang, Wensheng Niu. "A Hybrid Model Based on Singular Spectrum Analysis and Support Vector Machines Regression for Failure Time Series Prediction" , Quality and Reliability Engineering International, 2016

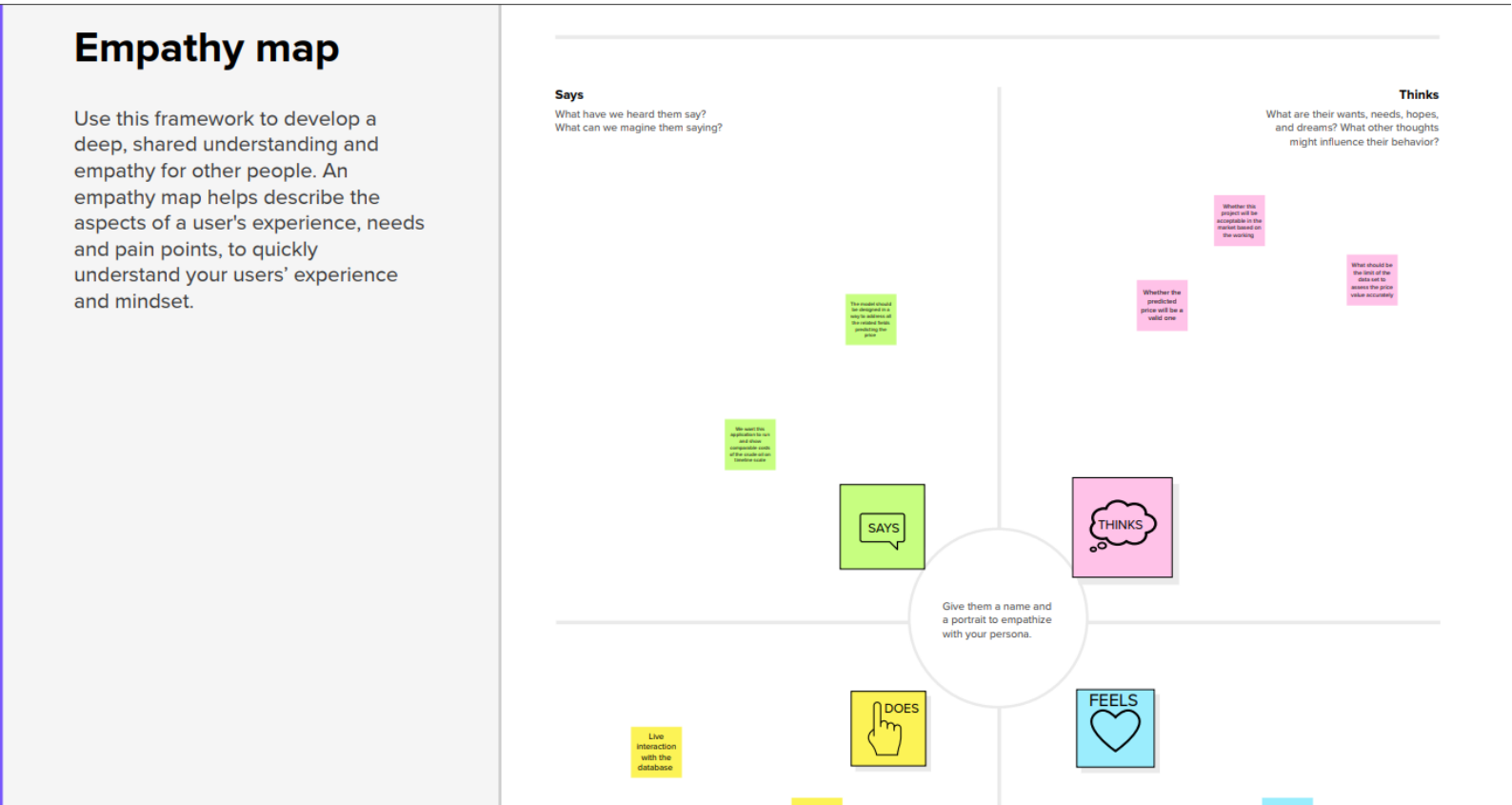
## 2.3 Problem Statement Definition

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.

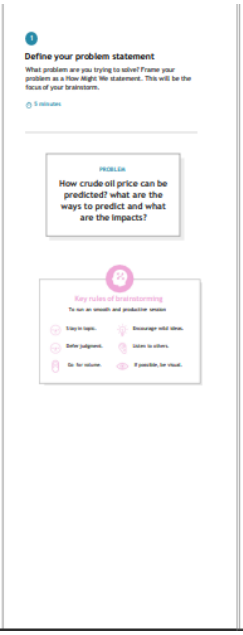
This Guided Project mainly focuses on applying Neural Networks to predict the Crude Oil Price. This decision helps us to buy crude oil at the proper time. Time series analysis is the best option for this kind of prediction because we are using the Previous history of crude oil prices to predict future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

# 3. IDEATION & PROPOSED SOLUTION

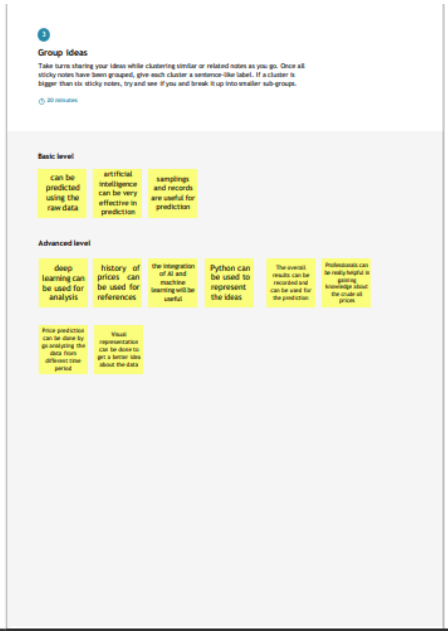
## 3.1 Empathy Map Canvas



## 3.2 Ideation & Brainstorming









## 3.3 Proposed Solution

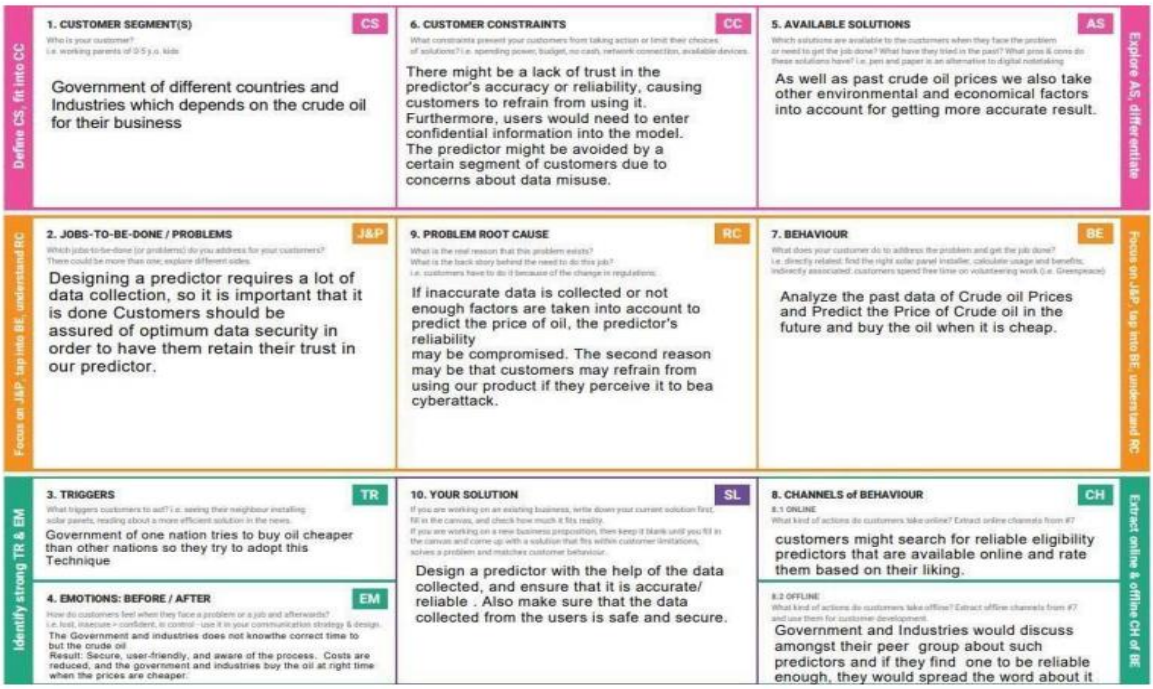
| S.No. | Parameter | Description |
|---|---|---|
| 1. | Problem Statement (Problem to be solved) | Oil demand is inelastic, therefore the rise in price is good news for producers because they will see an increase in their revenue. Oil importers, however, will experience increased costs of purchasing 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. | Idea / Solution description | This Guided 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. |

| | | |
|---|---|---|
| 3 | Novelty / Uniqueness | We use the concept of Artificial Neural Network and Machine Learning To predict the price of Crude Oil More accurately Than other existing Models.<br>The main advantage of artificial neural network is that it continuously captures the unstable pattern and variations of crude oil price. |
| 4. | Social Impact / Customer Satisfaction | Crude oil is amongst the most important resources in today's world, it is the chief fuel and its cost has a direct effect on the global habitat, our economy and oil exploration, exploitation and other activities.<br>Prediction of oil prices has become the need of the hour, it is a boon to many large and small industries, individuals and the government. |
| 5 | Business Model (Revenue Model) | Financially, this project could benefit the small scale and large-scale industries. The receipt and expenditure of oil revenues are matters for fiscal policy, and we consider them in the context Of India's federal system, where fiscal responsibility is divided between the federal government and state governments. But the time profile Of oil revenues is distinctive compared with fiscal revenues more generally: oil revenues are volatile, driven largely by the volatility of oil prices. The required data sets are obtained from Kaggle.com The dataset was used to train various models . |
| 6 | Scalability of the Solution | In this Project , We use Artificial Neural Network and various ML Algorithms To predict the Unstable Variations of Crude Oil Price Over a Given Time Period.<br>To Predict the Price of Crude Oil In Future We Train The Data Model With The Past Oil Prices Data Which we Obtain From Kaggle.com. |

# 3.4 Proposed Solution fit

**Project Title** : Crude Oil Price Prediction

**Project Design Phase-I - Proposed Solution Fit** **Team ID** : PNT2022TMID14370




# 4. REQUIREMENT ANALYSIS

## 4.1 Functional requirement

**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. |
| FR-2 | User Confirmation | Confirmation via SMS. |
| FR-3 | Fetching input data | Give the model the input data. |
| FR-4 | Generating Results | Prediction of Oil Prices. |

## 4.2 Non Functional requirement

**Non-functional Requirements:**

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

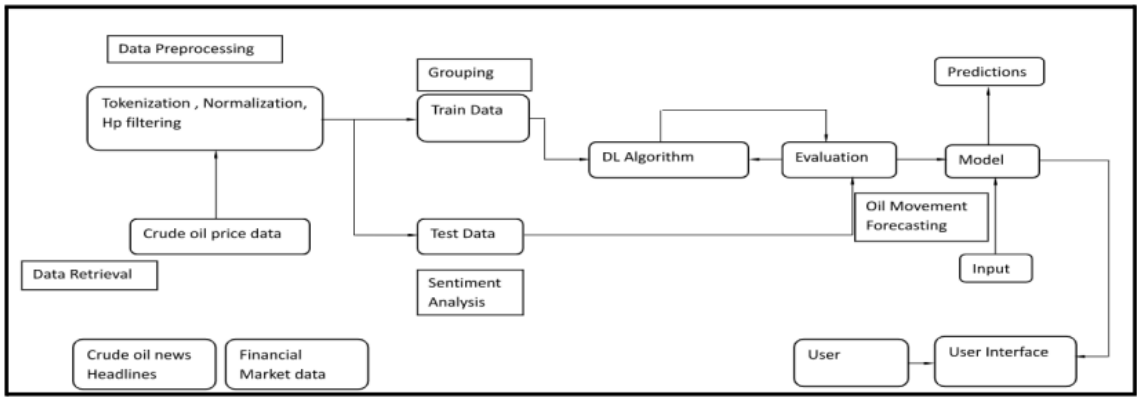
| NFR No. | Non-Functional Requirement | Description |
|---|---|---|
| NFR-1 | **Usability** | user interfaces are easy to use. |
| NFR-2 | **Security** | Sensitive data is protected. |
| NFR-3 | **Reliability** | Because there is very little variance from the prediction, the testing is highly dependable. |
| NFR-4 | **Performance** | Using LSTM networks gives highly performance. |
| NFR-5 | **Availability** | The system tested with 4 datasets and the system operating properly. |
| NFR-6 | **Scalability** | LSTM network model works efficiently for large number of users. |

# 5. PROJECT DESIGN

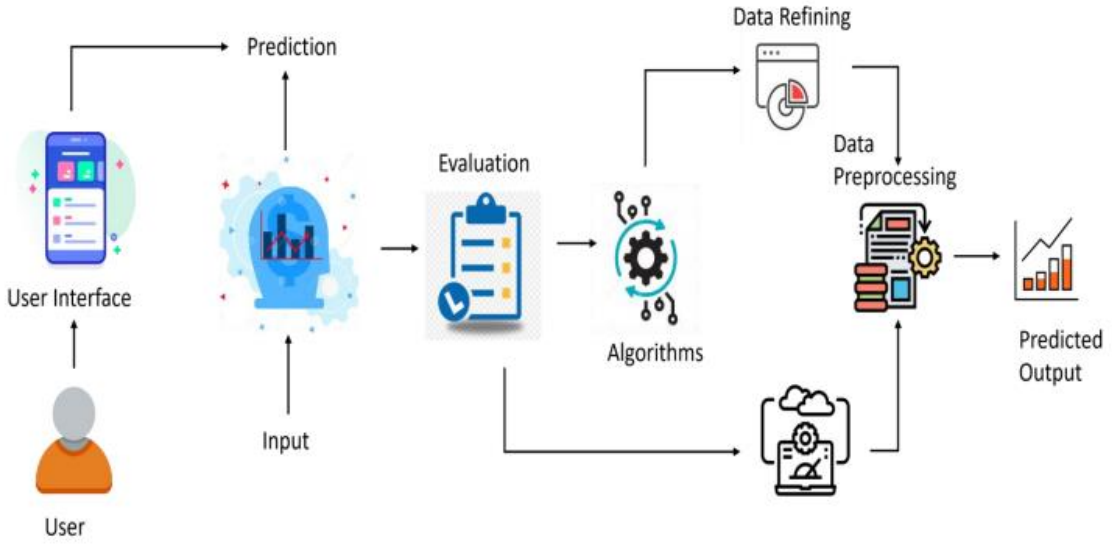
## 5.1 Data Flow Diagrams

**Data Flow Diagrams:**

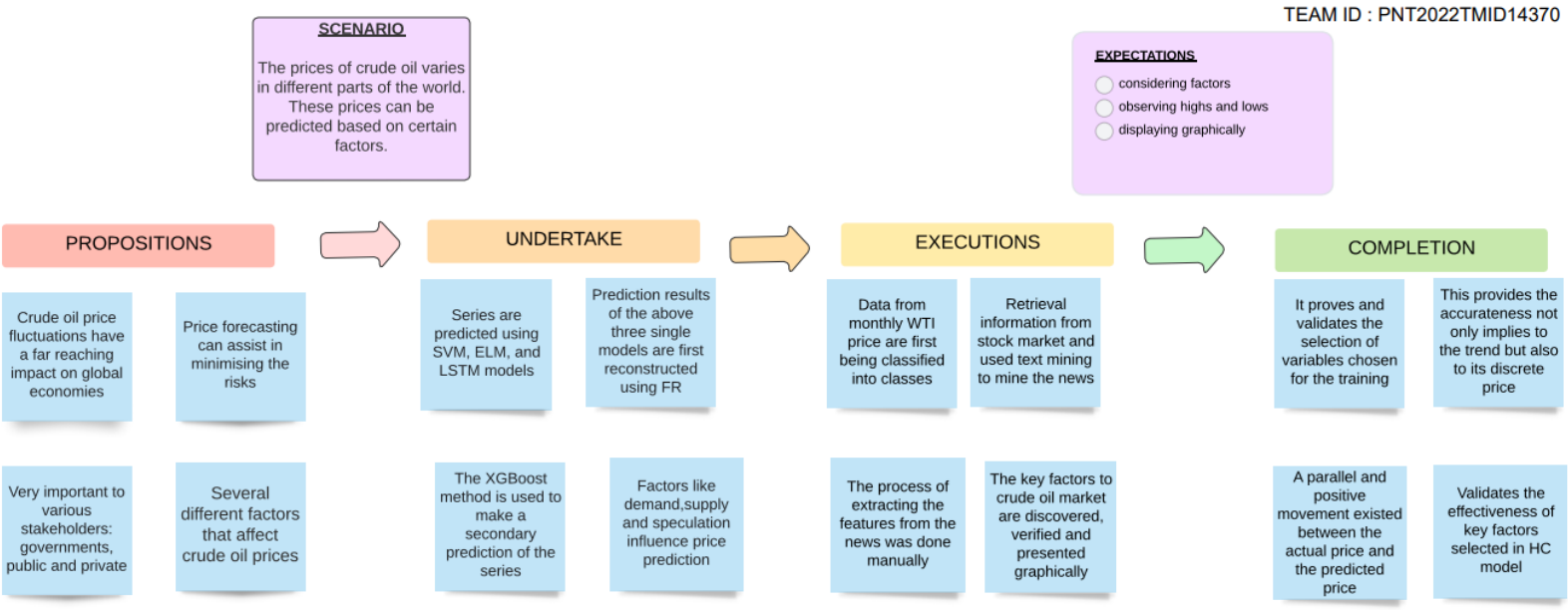
A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It shows how data enters and leaves the system, what changes the information, and where data is stored.



## 5.2 Solution & Technical Architecture




## 5.3 User Stories




# 6. PROJECT PLANNING & SCHEDULING

## 6.1 Sprint Planning & Estimation

| Sprint | Functional Requirement (Epic) | UserStory Number | UserStory/Task | Story Points | Priority | Team Members |
|---|---|---|---|---|---|---|
| Sprint-1 | Data Collection | USN-1 | Download Crude Oil Price Dataset | 2 | Medium | Rex Milton S |
| Sprint-1 | Data Pre processing | USN-2 | Importing The Dataset into Workspace | 1 | Low | Niranjan JP |
| Sprint-1 | | USN-3 | Handling Missing Data | 3 | Medium | Praveen Kumar AM |
| Sprint-1 | | USN-4 | Feature Scaling | 3 | Low | Mohammed Umar |
| Sprint-1 | | USN-5 | Data Visualization | 3 | Medium | Madhavan |
| Sprint-1 | | USN-6 | Splitting Data into Train and Test | 4 | High | Rex Milton S |
| Sprint-1 | | USN-7 | Creating A Dataset with Sliding Windows | 4 | High | Niranjan JP |
| Sprint-2 | Model Building | USN-8 | Importing The Model Building Libraries | 1 | Medium | Praveen Kumar AM |
| Sprint-2 | | USN-9 | Initializing The Model | 1 | Medium | Mohammed Umar |
| Sprint-2 | | USN-10 | Adding LSTM Layers | 2 | High | Madhavan |
| Sprint-2 | | USN-11 | Adding Output Layers | 3 | Medium | Rex Milton S |
| Sprint-2 | | USN-12 | Configure The Learning Process | 4 | High | Niranjan JP |

| Sprint | Functional Requirement (Epic) | UserStory Number | UserStory/Task | StoryPoints | Priority | TeamMembers |
|---|---|---|---|---|---|---|
| Sprint-2 | | USN-13 | Train The Model | 2 | Medium | Praveen Kumar AM |
| Sprint-2 | | USN-14 | Model Evaluation | 1 | Medium | Mohammed Umar |
| Sprint-2 | | USN-15 | Save The Model | 2 | Medium | Madhavan |
| Sprint-2 | | USN-16 | Test The Model | 3 | High | Rex Milton S |
| Sprint-3 | Application Building | USN-17 | Create An HTML File | 4 | Medium | Niranjan JP |
| Sprint-3 | | USN-18 | Build Python Code | 4 | High | Praveen Kumar AM |
| Sprint-3 | | USN-19 | Run The App in Local Browser | 4 | Medium | Mohammed Umar |
| Sprint-3 | | USN-20 | Showcasing Prediction On UI | 4 | High | Madhavan |
| Sprint-4 | Train The Model On IBM | USN-21 | Register For IBM Cloud | 4 | Medium | Rex Milton S |
| Sprint-4 | | USN-22 | Train The ML Model On IBM | 8 | High | Niranjan JP |
| Sprint-4 | | USN-23 | Integrate Flask with Scoring EndPoint | 8 | High | Praveen Kumar AM |

## 6.2 Sprint Delivery Schedule

| Sprint | Total StoryPoints | Duration | Sprint StartDate | SprintEndDate (Planned) | Story Points Completed (as on Planned EndDate) | Sprint Release Date(Actual) |
|---|---|---|---|---|---|---|
| Sprint-1 | 20 | 6Days | 24Oct2022 | 29Oct2022 | 20 | 29Oct2022 |
| Sprint-2 | 20 | 6Days | 31Oct2022 | 05Nov2022 | 20 | 03Nov2022 |
| Sprint-3 | 20 | 6Days | 07Nov2022 | 12Nov2022 | 20 | 10Nov2022 |
| Sprint-4 | 20 | 6Days | 14Nov2022 | 19Nov2022 | 20 | 17Nov2022 |

# 7. CODING & SOLUTIONING

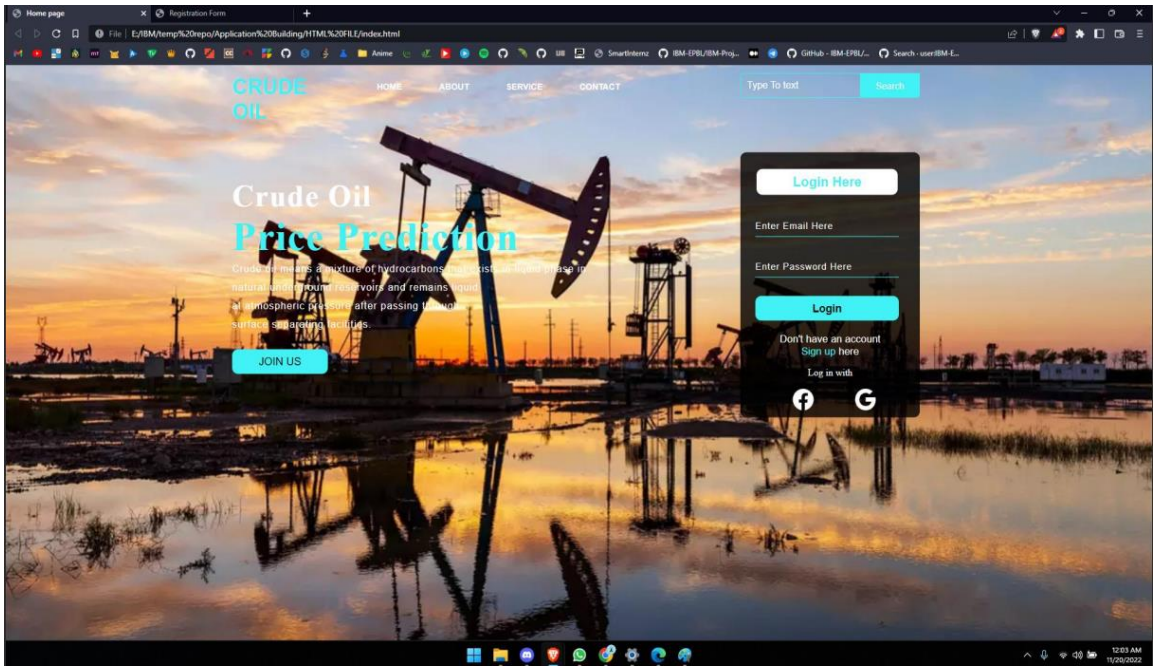
## 7.1 Feature 1

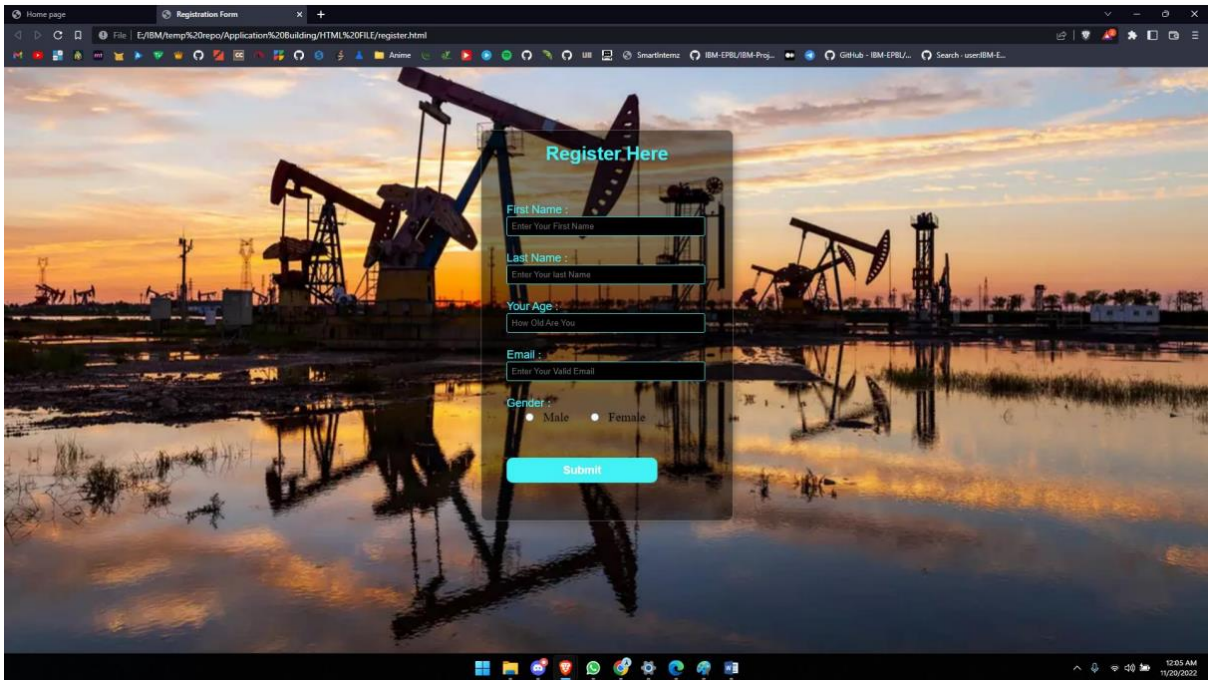
- IBM Watson Platform
- Web UI
- Python Code
- HTML
- CSS
- JS

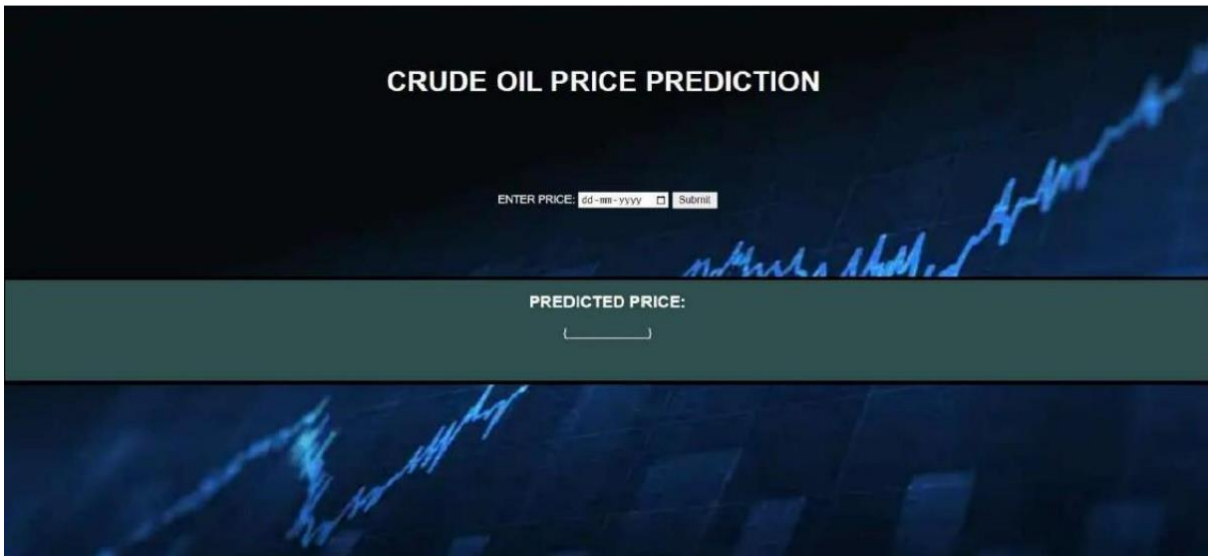
## 7.2 Feature 2

- Cloudant DB
- Neural Network
- NLP
- Artificial Intelligence

# 8. Testing And Results






# 9. Advantages

- User Friendly
- Predicts Crude Oil price precisely and accurately
- Helps Industries and Factories to Buy Crude oil At proper time.

# 10. Disadvantages

- The Predicted price by the system will not be always correct.
- Need Accurate Data of Crude Oil Prices in past to predict the accurate price of crude oil.

## 11. Conclusion

Crude Oil plays a major in the nations economy so that predicting the crude oil prices proves worthy and our project predicts the crude oil prices to a high accuracy.

## 12. Future Scope

Our projects plays a major role in predicting the crude oil prices it is of great importance in the upcoming years.

## 13. Appendix

## 13.1 Source Code

```
import numpy as np
from flask import Flask, render_template, request
from tensorflow.keras.models import load_model
app = Flask(__name__,template_folder='template')
model = load_model("./model/crude_oil.h5")
@app.route('/')
def home():
    return render_template('index.html')
@app.route('/predict')
def home2():
    return render_template('predict.html')
@app.route("/login", methods=['POST','GET'])
def Login():
    if request.method=='POST':
        a = request.form['year1']
        b = request.form['year2']
        c = request.form['year3']
        d = request.form['year4']
        e = request.form['year5']
        f = request.form['year6']
        g = request.form['year7']
```

```
h = request.form['year8']
i = request.form['year9']
j = request.form['year10']
x_input = [a, b, c, d, e, f, g, h, i, j]
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


        print(lst_output)


        return render_template("predict.html", showcase='The next day predicted value is:' +
str(lst_output))
if __name__ == '__main__':
    app.run(debug=True, port=5000)
```

## 13.2 GitHub

**https://github.com/IBM-EPBL/IBM-Project-39676-1660487181**