มหาวิทยาลัยอัสสัมชัญ
ASSUMPTION UNIVERSITY
of THAILAND

VINCENT MARY
SCHOOL of SCIENCE and
TECHNOLOGY

Senior Project Report

Fuel Stock Recommender System

Paranan Vitpornnitipacha,
Jirapat Suwanjunee,
Krittamet Chuwongworaphinit

Chayapol Moemeng (Advisor)

CS 4200/ITX 3010 Senior Project 2 (1/2022)

Senior Project Approval

Project title:	Fuel Stock Recommender System
Academic Year:	1/2022
Authors:	Paranan Vitpornnitipacha (6135118)
	Jirapat Suwanjunee (6217410)
	Krittamet Chuwongworaphinit (6111252)
Project Advisor:	Chayapol Moemeng

The Senior Project committee's cooperation between the Department of Computer Science and Information Technology, Vincent Mary School of Science and Technology, Assumption University had approved this Senior Project. The Senior Project in partial fulfillment of the requirement for the degree of Bachelor of Science in Computer Science and Information technology.

Approval Committee:

……..

(Chayapol Moemeng)

Project Advisor

……..

(Asst. Prof. Dr. Benjawan Srisura)

Committee Member

……..

(Asst. Prof. Dr. Paitoon Porntrakoon)

Committee Member

Abstract

The fuel and gasoline industry plays an important role in the world's Gross Domestic Product by providing energy resources fuel and gas companies require high operational efficiency in order to maintain profits. Because of the oil global economic framework, impacting everything from transportation to heating & electricity to industrial production & manufacturing. Service station business at this time, not only managing or thinking of a business strategy to be profitable, but it also includes the management of fuel storage and there are a lot of problems occurring during this business.

FSRS is a recommender system that focuses on recommending the gasoline and fuel stock as a solution to predict future purchases to appropriate the fuel stock situation for service station operators to be easier to manage business.

Acknowledgements

We wish to start by expressing our thanks to our adviser, Ajarn Chayapol Moemeng, for his guidance and advice, which helped us finish our project promptly and effectively. He provided us with excellent advice and helped us through difficult circumstances. A. Chayapol has continually pushed us to tackle new challenges from the start to finish of the project, which has helped us learn more about our project. His cooperation and passion were a big part of why the project was successful.

Table Of Contents

Table Of Figures

Table Of Tables

Chapter 1: Introduction

1.1 Problem Statement

Fuel and natural gas are major industries in the energy market and play an influential role in the world's Gross Domestic Product by providing energy resources as the primary fuel sources. The processes and systems involved in producing and distributing fuel and gas are highly complex during manage services station there are many issues consider for the fuel ordering of the service station which are listed as following:

1.1.1 Wet Stock Management

One of the most important problems of wet management is about the requirement and condition when deciding to create an order. There are several factors that the owner has to consider. First is about transportation, truck compartment container standard 44,000 liter can drive no more than 60 km/hr which consist of 7 compartments each compartment has a different capacity 9000 - 10000, 9000 - 10000, 5000 - 6000, 5000 - 6000, 5000 - 6000, 4000 - 5000, and 4000.

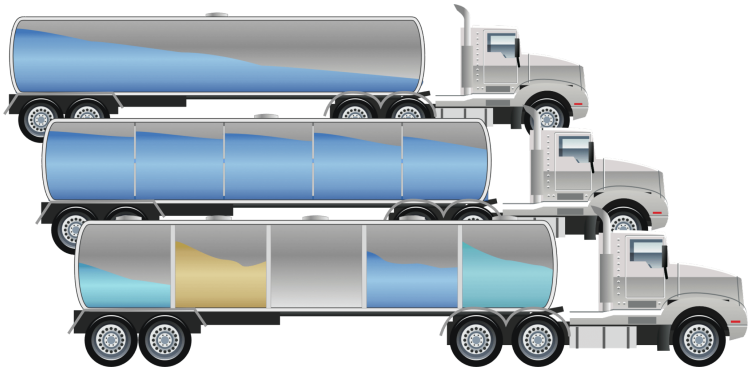


Figure 1. Truck Tank Compartment

Second, the owner of the service station has to consider the minimum requirement in each tank. For example, if you order 10000 liters of fuel which will be stored in a 9000-10000 liters compartment so it can contain a maximum of 10000 and a minimum of 9000 can not lower than 9000 in that tank. Moreover, there also a minimum total must be equal or more than 22000 liters when creating an order. Each time the owner must order one large tank at 9000 -10000 liters because if it contains a small quantity, it is not worth the cost of transportation so the order won't be accepted.

1.1.2 Stock Checking

The owner of the service station has to check fuel and gasoline stock every morning. There is no automated reporting, and notifications so they do not know how much to order the right amount of minimum fuel for the truck transportation. Moreover, the owner have to spend a lot of time calculating fuel manually and there is a mistake in planning in the matter of the order which causes price damage and minimum ullage.

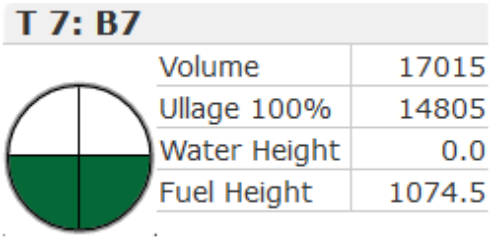
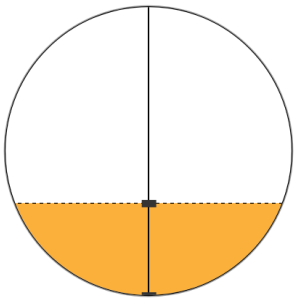


Figure 2. Tank 7 Diesel B7 Stock

The tank capacity is 100% maximum, but it can actually be reduced to only 90% legally based on the law in Thailand. On the other hand the minimum tank bottom that allow to sell fuel to customer is at 800 liters, if lower than 800 liters the fuel can't sell to customer because it won't be able to extract fuel so it can cause of damage to the customer's car if the remaining fuel in the tank is low the bottom of the tank has impurities and fuel stains may cause of damage in customer vehicle.

1.1.3 Human Error

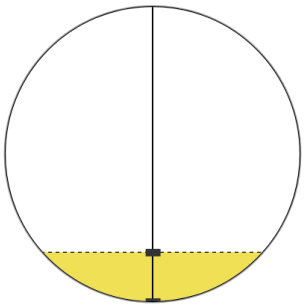
What goes wrong with fuel operators is that they order the wrong product because they misunderstand the name or brand of similar fuel. A simple example is Gasohol 95 and Gasohol 95 Plus as shown below. The owner ordered the wrong product for 9000 liters because he ordered the wrong product. In spite of selling an average of 200 liters per day due to fuel premium grade it takes up to 2 months to drain the stock this causes damage in business management.



Tank 4: GASOHOL 95

Volume	5074 lit
Ullage 100%	10091 lit
Ullage 90%	8574 lit
Height	750.0 mm
Water Volume	0 lit
Water Height	0.0 mm
Temperature	31.2 °C
Alarms / Warnings	
Maximum Product	15665 lit
High Product	95 %
Overfill	90 %
Delivery Needed	0 %
Low Product	1500 lit
High Water Warning	21.0 mm
High Water Alarm	21.0 mm
Tank Setup	
Capacity	0 lit
Diameter	2140.0 mm

Figure 3. Gasohol 95 Tank Detail



Tank 1: GASOHOL 95 PLUS

Volume	2733 lit
Ullage 100%	12643 lit
Ullage 90%	11105 lit
Height	489.0 mm
Water Volume	0 lit
Water Height	0.0 mm
Temperature	32.2 °C
Alarms / Warnings	
Maximum Product	15875 lit
High Product	95 %
Overfill	90 %
Delivery Needed	0 %
Low Product	1500 lit
High Water Warning	21.0 mm
High Water Alarm	21.0 mm
Tank Setup	
Capacity	0 lit
Diameter	2140.0 mm

Figure 4. Gasohol 95 Plus Tank Details

1.1.4 Complexity of Ordering

Relevant people or family members cannot do this due to the complexity of ordering.

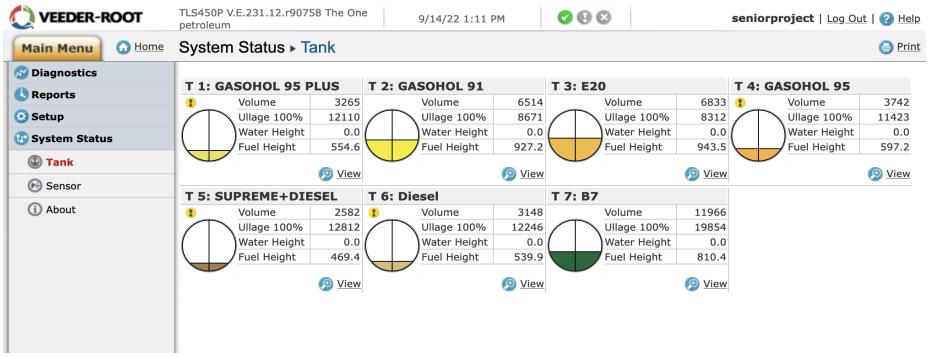


Figure 5. Veeder-Root Web Interface

1.2 Scope of the project

FSRS Stock Recommender system is a project that focuses on the recommendation step of the process on algorithms to predict and analyze for certain products. These predictions are used as recommendations, leading to a better customer experience and an important increase in revenues. This project is related to using algorithms to recommend fuel stock for service stations. There is one stakeholders for this project

1.2.1 Service Station Operator

The following are the requirement of this project

- With Line Chatbot users can ask to view the daily amount of fuel.
- Users can also see the average sales per day of fuel in each tank.
- Users are able to see how many days left of fuel will be out of tanks.
- Recommend the fuel order for the service station which tank and how much amount of fuel should be ordered.

Chapter 2: Related Work

2.1 Recommendation Systems

A recommendation system is a subclass of information filtering systems that attempts to estimate a user's rating or preference for an item. In layman's terms, it is an algorithm that recommends relevant goods to consumers. For example, which movie to watch on Netflix, which product to buy on e-commerce, which book to read on Kindle, and so forth. Recommendation Systems are widely used in the business industries. It is extremely important in various businesses since they may create a considerable amount of revenue when they are efficient or serve as a means to differentiate considerably from rivals. Scalability of algorithms with real-world datasets is another challenge with recommendation systems. In most situations, the old strategy has been overwhelmed by the number of items and clients, resulting in dataset issues and decreased performance.

2.2 Expert Systems

An expert system is a computer-based decision-making system that is interactive and dependable, and it combines both facts and heuristics to handle complicated decision-making issues. It is said to represent the highest level of human knowledge and competence. An expert system's objective is to tackle the most difficult problems in a certain subject. Many industries have used expert systems to solve the problem such as The CaDet expert system is a diagnostic support system that can detect cancer at early stages. These systems are used in the hospital to diagnose cancer. The expert system excels at addressing any sort of complicated problem in a specified domain with great efficiency and accuracy. However, If the knowledge base includes incorrect information, the expert system's response may be incorrect. Moreover, It cannot learn from experience and hence requires manual updating.Its also significantly more difficult to acquire knowledge for design.

Chapter 3: Proposed Methodology

3.1 Methodology

The recommender system deals with a large volume of information present by filtering the most important information based on the data provided by a user and other factors that take care of the user's preference and interest. It finds out the match between user and item and imputes the similarities between users and items for recommendation. Both the users and the services provided have benefited from these kinds of systems. The quality and decision-making process has also improved the business entrepreneur.

There are several different things that can be recommended by the system and this project will use this methodology to recommend gasoline and fuel to predict stock order and remaining amount of fuel and gasoline. We believe that the recommender system will come to reduce expenses and increase income for service stations.

3.2 Algorithms

During the project implementation we have researched various methods to see which one is best fit for our project.

First of all we have used Puppeteer which is a Node JS library where it facilitates automation and simplification of development by giving developers control over their tools. Developers can create and maintain straightforward automated tests thanks to it. We used this for the scraping of the essotheone web site to collect data of all the tanks and store it in MongoDB. MongoDB is a cross-platform document-oriented database application that is open source. MongoDB, a NoSQL database application, employs documents that resemble JSON and may or may not include schemas. Moreover, we also create a new api that gets all the data from the database using Express.js. Express is a Node.js back-end web application framework for creating RESTful APIs.

Furthermore we used a line chat bot to communicate with the users to recommend the fuel ordering using rule-based methods. A rule-based system is a system that applies human-made rules to store, sort and manipulate data. To work, rule-based systems require a set of facts or source of data, and a set of rules for manipulating that data. Rule-based logic is at the heart of most automated processes.

Chapter 4: Design of The System (or Work)

4.1 Functional Requirements Specification

4.1.1 Stakeholders

A stakeholder is the service station operator and other members in business. In the beginning the service station usually faced problems when deciding to create a fuel order to maintain tank capacity to suit the situation. At the service station there are approximately 7 tank available to store each different type of fuel such as Gasohol 95 plus, Gasohol 91, E20, Gasohol 95, Diesel supreme plus, Diesel and Diesel B7 so the service station operator is a person to take responsibility to manage each fuel tank, estimate the amount of fuel stock to create an order manually which has to concern many factor condition. For example, minimum, maximum of fuel and truck tank compartment when delivered to a station which require 22000 liters per time when created order if less than that it would not be accepted. Moreover, these things can be the only service operator that can make a decision to create a fuel stock order in the morning.

Fuel Stock Recommender system will provide the feature and tools to help stakeholder to manage fuel station business easier . These features recommend the fuel that is appropriate to create an order, sort of the best seller of fuel, chatbot and predict preferences.

4.1.2 Use Cases

In the Fuel Stock Recommender system project, it explains about recommending the fuel and gasoline order to reduce the time of business entrepreneurs so it will help service station operators not have to waste time to check every day on a certain website, looking for fuel stock. So this project come to automate all tasks from backend to front end to get response notification order via a line chat bot to recommend the amount of fuel and gasoline on that day not to overestimate or underestimate but it in appropriate way so the duty of service operator is just to follow the bot suggestion to make fuel stock order.

4.1.3 Tank Volume

The station operator can communicate with the line bot and ask the current volume of fuel in the tank everyday. It will respond with the list of all tanks and volume that get data scraped from the website everyday 8:00 am.

4.1.4 Best Seller

The station operator can also ask the line bot to see the best seller fuel from the past 7 days. It will show the average sales of the fuel in this case it will also help in recommending the step how much fuel volume should be ordered.

4.1.5 Most out of Tank

This part shows the users the estimation of how many days left that the fuel in the tanks will run out.

4.1.5 Suggestion

The station operator can type in the line chat bot to ask for the suggestion which type and how much amount of fuel should be ordered.

4.2 System Design (consider items that applied)

This part is about the system design and procedures of the Fuel Stock Recommender System (FSRS). It represents the process, and workflow of the systems.

4.2.1 Process / Workflow

In this part, flowchart shows the steps as boxes of various kinds, and their order by connecting the boxes with arrows analyzing, designing, a process in various fields. Figure 7 showing the action of workflow when the service station operator uses line bot to recommend the order it represents a process when using FSRS to recommend gasoline and fuel.

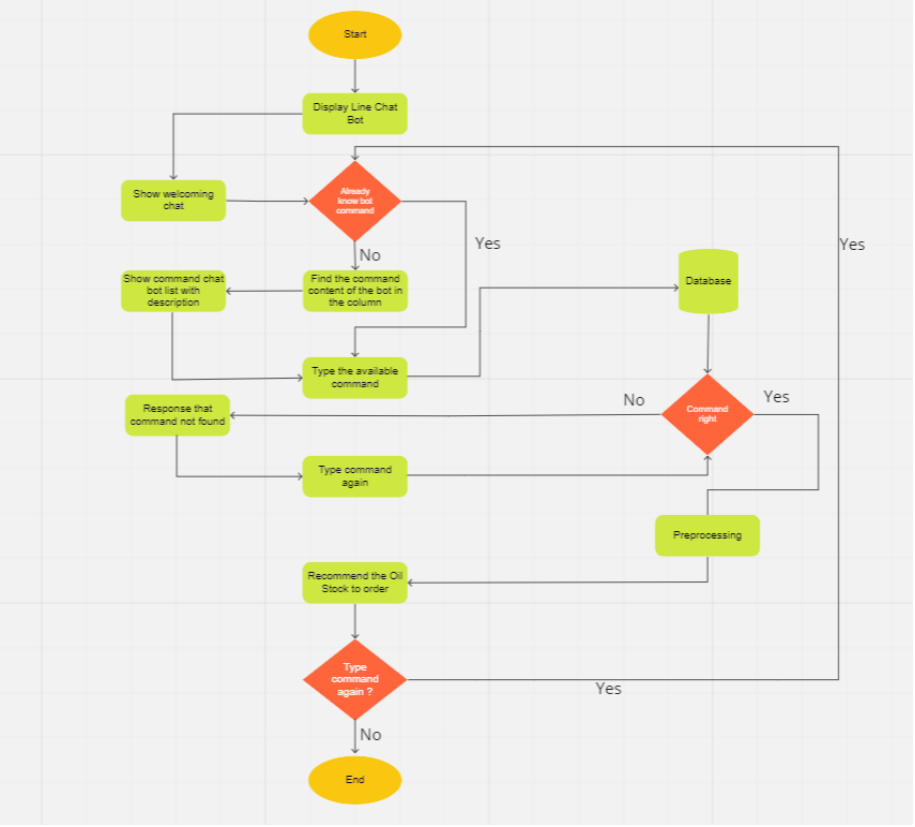


Figure 6. Work Flow of Fuel Recommender System

4.2.2 System Architecture

The backend system represents an overall backend that automates scraping data from the service station web page by using a library well known as puppeteer.

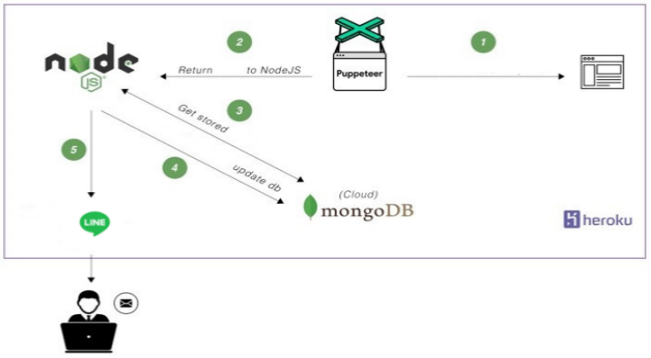


Figure 7. System Architecture of FSRS

Puppeteer is a node library which provides a high-level API to control from Chrome and Firefox over the DevTools Protocol that runs headless by default, and can be configured to run full non-headless both in Chrome and Firefox. Fundamentally, puppeteer is an automation tool and not a test tool.

Figure 8. Puppeteer Logo

For this part we use puppeteer to scrape and generate data from the veeder-root web interface. It will automatically open a Firefox browser instance, open a new page in the browser and navigate to the website. The reason that we have to scrape through the Firefox search engine is because the veeder-root web interface could not be accessed via Chrome browser. The figure shows data that we scrape from the veeder-root web interface.

```
{
  "_id": "62e7255b2a8f70a15ee4b406",
  "date": "1/8/2022",
  "Tank": [
    {
      "TankName": "T 1: GASOHOL 95 PLUS",
      "Volume": "3166",
      "Ullage": "12209",
      "Waterheight": "0.0",
      "Fuelheight": "542.6"
    },
    {
      "TankName": "T 2: GASOHOL 91",
      "Volume": "9520",
      "Ullage": "5665",
      "Waterheight": "0.0",
      "Fuelheight": "1250.6"
    },
    {
      "TankName": "T 3: E20",
      "Volume": "6265",
      "Ullage": "8880",
      "Waterheight": "0.0",
      "Fuelheight": "882.0"
    },
    {
      "TankName": "T 4: GASOHOL 95",
      "Volume": "11228",
      "Ullage": "3937",
      "Waterheight": "0.0",
      "Fuelheight": "1422.0"
    },
    {
      "TankName": "T 5: SUPREME+DIESEL",
      "Volume": "1818",
      "Ullage": "13575",
      "Waterheight": "0.0",
      "Fuelheight": "367.5"
    },
    {
      "TankName": "T 6: Diesel",
      "Volume": "3626",
      "Ullage": "11767",
      "Waterheight": "0.0",
      "Fuelheight": "597.3"
    }
```


Figure 9 . Fuel Data in JSON format


```
[
  'T 1: GASOHOL 95 PLUS',
  'T 2: GASOHOL 91',
  'T 3: E20',
  'T 4: GASOHOL 95',
  'T 5: SUPREME+DIESEL',
  'T 6: Diesel',
  'T 7: B7'
]
value [
  'Volume',       '4705',   'Ullage 100%',  '10670',
  'Water Height', '0.0',    'Fuel Height',  '721.5',
  'Stick Height', '',       'Volume',       '4923',
  'Ullage 100%',  '10263',  'Water Height', '0.0',
  'Fuel Height',  '752.4',  'Stick Height', '',
  'Volume',       '5730',   'Ullage 100%',  '9415',
  'Water Height', '0.0',    'Fuel Height',  '823.5',
  'Stick Height', '',       'Volume',       '8390',
  'Ullage 100%',  '6775',   'Water Height', '0.0',
  'Fuel Height',  '1109.9', 'Stick Height', '',
  'Volume',       '2503',   'Ullage 100%',  '12891',
  'Water Height', '0.0',    'Fuel Height',  '459.3',
  'Stick Height', '',       'Volume',       '7034',
  'Ullage 100%',  '8359',   'Water Height', '0.0',
  'Fuel Height',  '973.4',  'Stick Height', '',
  'Volume',       '18819',  'Ullage 100%',  '13001',
  'Water Height', '0.0',    'Fuel Height',  '1168.7',
  'Stick Height', ''
]
```


Figure 10 .Puppeteer Scraping Result

4.3 Tech Stack

- Node JS
- Puppeteer
- Heroku
- MongoDB
- Express.js
- Line Chat Bot

4.4 UI UX Design

UI/UX design is user oriented so, we make sure that the design meets the needs of the user that will help users achieve the desired results smoothly. The Figure 12 and Figure 13 below shows the design of the line chat bot responding to user. UI design, can take various forms. Basically, list items are grouped into three main categories by order of importance as: supporting visuals, primary text and metadata.

4.4.1 Why is list UI design important ?

Since users can only interact with your app or website through the UI, the design determines their primary display. If you've tried a low-effort app or website You know how frustrating it can be to deal with a poorly designed UI.

4.4.2 UI design that use in project

- **Supporting visuals**

  As the name suggests, supporting visuals draw attention to the list item, helping to convey basic information about the item to your users.

- **Primary text**

  Primary text is the main text element in a list item. It should represent the most important piece of information for your users.

- **Single-line list**

  A single-line list, like the name suggests, has just one line per item. Single-line lists can be very effective in that they provide optimal scannability.

- **Two-line list**

  Two-line lists provide a primary and secondary text. These are helpful when you need to go into a little bit more detail about the line item itself.

- **Consistency**

  When it comes to improving the scannability of your list UI design, consistency is very important. A good list UI design has to have a consistent design all the way through. Otherwise, it doesn't really serve its function as a list and doesn't help the user to scan through all the information which is precisely what it's meant to do.

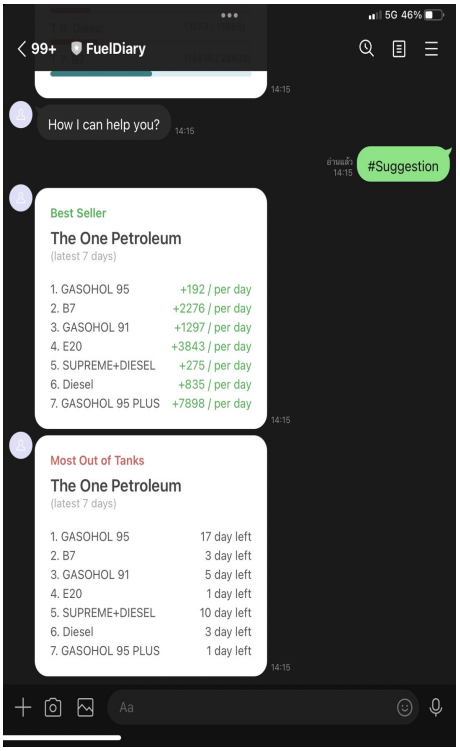


Figure 11 Show Fuel in daily.

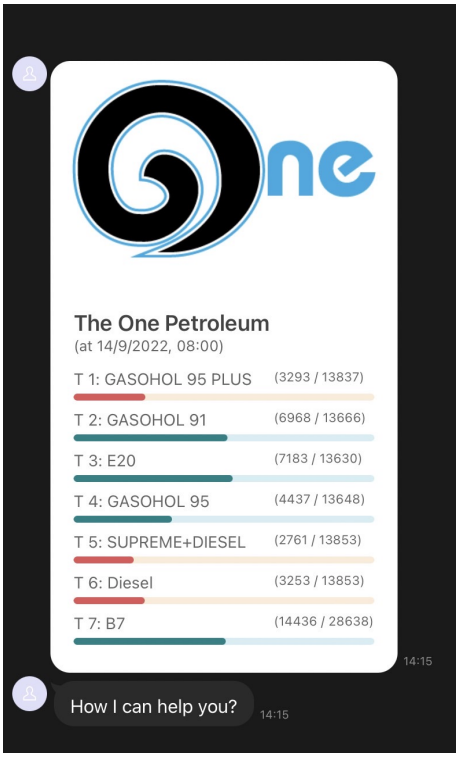


Figure 12 Amount of fuel volume per day.

4.5 Database

For the database system in this project using mongoDB to record data each day because it has an awesome with a complete application data platform. With MongoDB Atlas, the cloud offering by MongoDB, access to a collection of services that all integrate nicely with databases. MongoDB's document model allows virtually any data structure to be modeled and manipulated easily. MongoDB's BSON data format, inspired by JSON, allows you to have objects in one collection with different sets of fields. After we have done with recording data in the database, the next thing that we do is deploy a process, to let it work automated scraping the data from esso website our group select cloud platform Heroku. Heroku is a cloud platform as a service (PaaS) supporting several programming languages.

The Heroku Network runs its customers' apps in virtual containers that run on a trusted runtime environment. Heroku calls these containers "dynos." These Dynos can run code written in Node JS, Ruby, PHP, Go, Scala, Python, Java or Clojure. Moreover, Heroku also offers custom buildpacks where developers can deploy apps in other languages. Heroku allows developers to scale their apps on the fly by increasing the number of dynos or changing the dynos type where the app works.

4.6 Rule Based Case and Calculations

The table 1 below explains how we used a rule based system to implement the fuel recommendation. We have listed all the 8 cases that are considered during the project implementation.

Table 1. Rule Based Case:

No.	Case
1.	Fuel order greater than or equal 22000 liter and have to order at least 1 fuel at the biggest compartment of 9000-10000 L The delivery truck has 7 compartments consist of: 1) 9000-1000, 2) 9000-10000, 3) 5000-6000 ,

No.	Case
	4) 5000-6000, 5) 5000-6000, 6) 4000-5000 7) 4000
2.	Volume left in the tank must not be less than 1900 L (actually 800 ,we write 1900 due to maintain the quality of fuel)
3.	Tank capacity is 100% maximum, but it can actually be filled up to only 90%.
4.	Calculate and find how many days it will take for the fuel in a tank to run out and then order by how many days the tanks will run out. Less number of days will come first.
5.	Pick the tank that will run out in less than or equal to 4 days and that tank will be able to fill at least 4000 L from the maximum tank capacity.
6.	Additional case from no.5 if there is fuel less than the current minimum 1900 liter it will be taken into consideration without a number of days that each fuel is running out of stock. For example, Tank 1 Gasohol 95 Plus there are 1600 liters left in the tank capacity but the day that fuel runs out is in the next 8 days. Although the expiration date is more than 4 days, due to rule No.2 to maintain the fuel quality then we will take it into consideration as well.
7.	After we have the tanks that pass all the conditions above we can confirm that those are the fuel that should be ordered and to know which fuel should be ordered with how much amount we check with the daily average sales of the fuel. In which the best seller tanks will be considered in the biggest amount 9000-1000 L everytime. On the other hand, the fuel that has the least sales such as T1: gasohol 95 plus will be chosen to the compartment of 4000L first. This allows us to reduce the stock of fuel. So that we do not take the money to sink with abortion that sells less for longer than necessary.
8.	Below the following statement will be the case of ordering, what kind of chances can be ordered. By the way it will recommend 28,000 L and 34,000L it will order 28000 L because we do not need to maintain each fuel for too long. So, then the method that we will use to fill the tank is divided into 4 cases which are ordered and sorted by using the if else statement. For example, fuel that is considered to fill 2 tanks 18000 4000 but if the maximum fuel in the tank can fill exceeds we will consider to order the next condition which is B7 16,000, Gasohol 95 6000 L. For example, fuel Tank 7 B7 16000 L the Tank number below is the tank number that has calculated that is B7 16000 is the combination between 10000 L and 6000L 1.1) 2 tank case 1.2) 3 tank case 1.3) 4 tank case 1.4) 5 tank case

No.	Case
	2 tank case -> 18,000, 4,000 = 22,000 EX. T7: B7 18000 (9000, 9000) , T1 Gasohol 95 plus: 4000 16,000, 6,000 = 22,000 Ex T7 B7: 16000 , T4 Gasohol 95: 6000 L 13,000, 9,000 = 22,000 **3 tank case** -> 9,000, 9,000, 4,000 10,000, 6,000, 6000, 12,000, 6,000, 4000 = 22000 13,000, 5,000, 5000 = 23000 **4 tank case** -> 9,000, 5,000, 5,000, 4,000 = 23000 **5 tank case** -> 9,000, 5,000, 5,000, 5,000, 4000 = 28000 9,000, 9,000, 6,000, 6,000, 4,000 = 34000 10,000, 9,000, 6,000, 5,000, 4,000 = 34000

Chapter 5: Result

5.1 Project Result

This part is about the demonstration of FSRS work. There are several scenarios provided. To begin with the scenario on date 07/10/22. As you can see in the Figure 14 There are three card components shown on chat bot best seller and most out of tanks.

1. **The Best Seller**

   The Best Seller predicts the volume per day that the service station will sell fuel and gasoline to customers to calculate the tank balance as you can see in Table 2 in column five **Tank Balance** we will use that to find out which fuel that service station operator should order.

2. **Most Out of Tank**

   Most Out of Tank show the number of days that each fuel will run out of the tank so the less day left, the greater risk that the fuel will run out the tank.

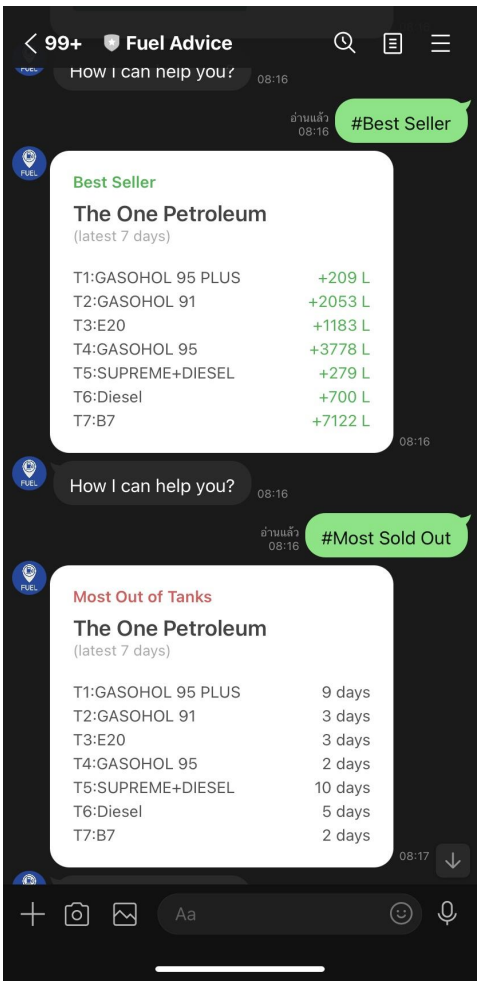


Figure 13 Fuel Best Seller and Most out of Tanks

3. Volume of Fuel

The Figure 15 below shows the volume of the fuel in the tanks that get data from the API that we scrape from the website everyday at 8:00 am.

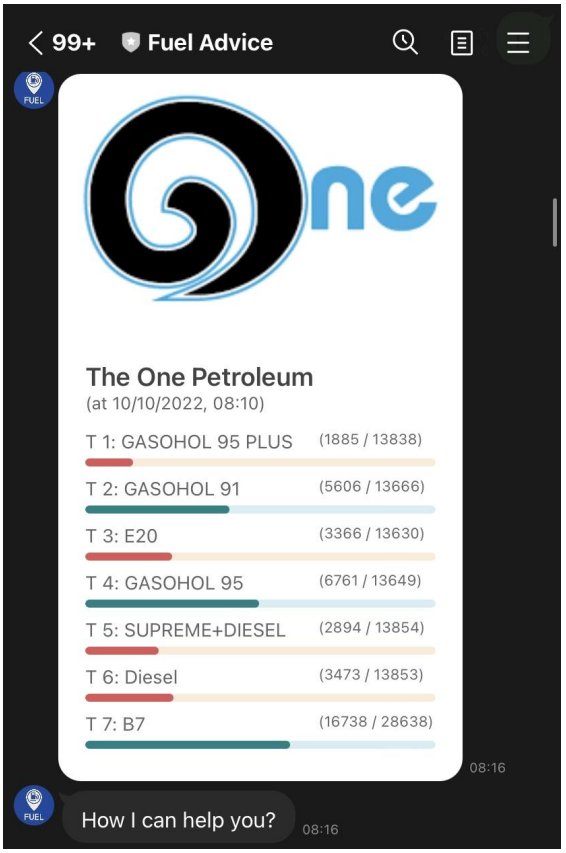


Figure 14. Estimated daily spending

4. Suggestion

The Figure 16 below shows the suggestion of the fuel in each to order everyday at 8:00 am.

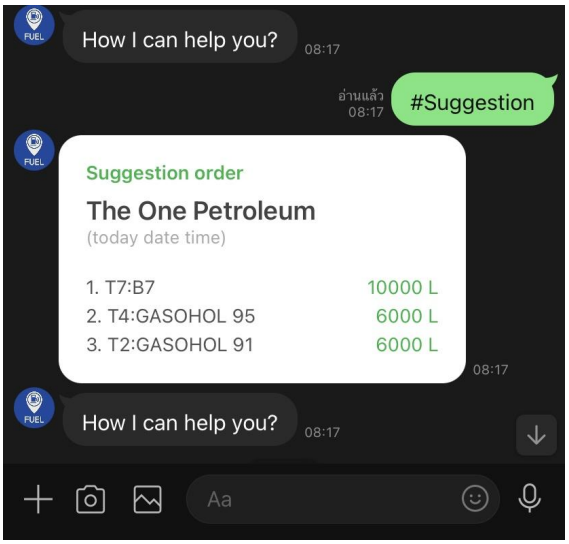


Figure 15. Suggestion order

5.2 Evaluation

Analysis: We evaluate the model by doing the back test of the data 50 days from 20/08/2022 - 11/10/2022 shown in Table 2 below. The performance result of the system is 54% accurate from what the actual order is. We evaluate the test based on which day there is order or not if it matches with the recommended order. We also see if the type of tanks recommended to order match the actual order. Lastly, we also see the amount of recommended order is similar and not much different from the actual order. However, our recommendation is a suggestion that helps to guide the station operator what should be ordered. The station operator can consider to follow our recommendations or order less than or more than what the system has suggested.

Table 2 Back Test Result:

Date	Recommend order	Actual order
11/10/22	**No suggestion**	**T7 B7: 12000 L** **T1 Gasohol 95 Plus: 4000 L** **T5 Supreme+Diesel: 6000 L**
10/10/22	**T7 B7: 9000 L** **T4 Gasohol 95: 5000 L** **T2 Gasohol 91: 5000 L** **T3 E20: 5000 L** **T1 Gasohol 95 Plus: 4000 L**	**T7 B7: 10000 L** **T4 Gasohol 95: 6000 L** **T2 Gasohol 91: 6000 L** **T3 E20: 6000 L** **T6 Diesel: 4000 L**
09/10/22	**No suggestion**	**No order**
08/10/22	**T7 B7: 16000 L** **T4 Gasohol 95: 6000 L**	**T7 B7: 22000 L** **T4 Gasohol 95: 9000 L**
07/10/22	**No suggestion**	**No order**
06/10/22	**No suggestion**	**No order**
05/10/22	**T7 B7: 9000 L** **T4 Gasohol 95: 9000 L** **T2 Gasohol 91: 4000 L**	**T7 B7: 18000 L** **T4 Gasohol 95: 6000 L** **T3 E20 4000 L**
04/10/22	**No suggestion**	**No order**
03/10/22	**T7 B7: 9000 L** **T4 Gasohol 95: 5000 L** **T2 Gasohol 91: 5000 L**	**T7 B7 : 10000 L** **T6 Diesel: 6000 L**

Date	Recommend order	Actual order
	T6 Diesel: 4000 L	
02/10/22	**No suggestion**	**No order**
01/10/22	**T7 B7: 9000 L** **T5 Supreme+Diesel: 4000 L** **T4 Gasohol 95: 5000 L** **T3 E20: 5000 L** **T2 Gasohol 91: 5000 L**	**T7 B7: 16000L** **T5 Supreme+Diesel: 4000 L** **T4 Gasohol 95: 9000 L** **T3 E20: 6000 L** **T2 Gasohol 91: 6000 L**
30/09/22	**No suggestion**	**No order**
29/09/22	**T7 B7: 9000 L** **T4 Gasohol 95: 5000 L** **T3 E20: 4000 L** **T2 Gasohol 91: 5000 L**	**T7 B7: 19000 L** **T4 Gasohol 95: 6000 L** **T3 E20: 4000 L** **T2 Gasohol 91: 6000L**
28/09/22	**No suggestion**	**No order**
27/09/22	**T7 B7: 10000 L** **T4 Gasohol 95: 6000 L** **T2 Gasohol 91: 6000 L**	**T7 B7: 16000 L** **T4 Gasohol 95: 6000 L** **T2 Gasohol 91: 6000 L**
26/09/22	**No suggestion**	**No order**
25/09/22	**T7 B7: 9000 L** **T4 Gasohol 95: 9000 L** **T2 Gasohol 91: 4000 L**	**T7 B7: 16000 L** **T6 Diesel: 6000 L** **T4 Gasohol 95: 9000 L**
24/09/22	**No suggestion**	**No order**
23/09/22	**T7 B7: 9000 L** **T4 Gasohol 95: 5000 L** **T3 E20: 4000 L** **T2 Gasohol 91: 5000 L**	**T7 B7: 10000L** **T4 Gasohol 95: 6000L** **T3 E20: 6000 L** **T2 Gasohol 91: 6000 L**
22/09/22	**No suggestion**	**No order**
21/09/22	**T7:B7 9000 L** **T4 Gasohol 95: 5000 L** **T2 Gasohol 91: 5000 L** **T1 Gasohol 91 Plus: 4000 L**	**T7 B7: 18000L** **T4 Gasohol 95: 6000 L** **T2 Gasohol 91: 6000 L** **T1 Gasohol 91 Plus: 4000 L**
20/09/22	**No suggestion**	**No order**
19/09/22	**No suggestion**	**T7 B7: 9000 L** **T6 Diesel: 4000 L** **T4 Gasohol 95: 9000 L** **T2 Gasohol 91: 6000 L**
18/09/22	**No suggestion**	**No order**

Date	Recommend order	Actual order
17/09/22	T7 B7: 9000 L T4 Gasohol 95: 5000 L T2 Gasohol 91: 5000 L T3 E20: 5000 L T5 Supreme+Diesel: 4000 L	T7 B7: 16000 L T4 Gasohol 95: 6000 L T3 E20: 6000 L T5 Supreme+Diesel: 4000 L
16/09/22	No suggestion	No order
15/09/22	No suggestion	T2 Gasohol 91: 6000 L T4 Gasohol 95: 6000 L T7 B7: 10000 L
14/09/22	T7:B7 9000 L T4:GASOHOL 95 5000 L T2:GASOHOL 91 5000 L T6:Diesel 4000 L	T4 Gasohol 95: 6000 L T6 Diesel: 4000 L T7 B7: 16000 L
13/09/22	No suggestion	No order
11/09/22	No suggestion	No order
10/09/22	T7 B7: 10000 L T6 Diesel: 6000 L T4 Gasohol 95 6000 L	T7 B7: 15000 L T6 Diesel: 4000 L T4 Gasohol 95: 6000 L
08/09/22	T7:B7 9000 L T4 Gasohol 95: 5000 L T2:GASOHOL 91 5000 L T3:E20 4000 L	T7 B7: 16000 L T4 Gasohol 95 : 6000 L T2 Gasohol 91: 6000 L T3 E20: 4000 L
07/09/22	No suggestion	No Order
06/09/22	T7 B7: 9000 L T4 Gasohol 95: 5000 L T3 E20: 4000 L T2 Gasohol 91: 5000 L	T7 B7: 14000 L T4 Gasohol 95: 8900 L T3 E20: 4000 L T2 Gasohol 91: 5900 L
05/09/22	No suggestion	No Order
04/09/22	T7 B7: 9000 L T6 Diesel: 4000 L T4 Gasohol 95: 5000 L T2 Gasohol 91: 5000 L	T7 B7: 15000 L T6 Diesel: 4000 L T4 Gasohol 95: 6000 L T2 Gasohol 91: 6000 L

Date	Recommend order	Actual order
03/09/22	No suggestion	No order
02/09/22	No suggestion	T7 B7: 9000 L T4 Gasohol 95: 6000 L T1 Gasohol 95 Plus: 4000 L
01/09/22	T7 B7: 9000 L T4 Gasohol 95: 5000 L T3 E20: 4000 L T2 Gasohol 91: 5000 L	T7 B7: 9000 L T5 Supreme + Diesel: 4000 L T3 E20: 6000 L T2 Gasohol 91: 6000 L
31/08/22	No suggestion	No Order
30/08/22	T7:B7 9000 L T6:Diesel 4000 L T4 Gasohol 95: 5000 L T2:Gasohol 91 5000 L	T7 B7: 10000 L T6 Diesel: 4000 L T4 Gasohol 95: 6000 L T2 Gasohol 91: 6000 L
29/08/22	No suggestion	T7 B7: 19000 L
28/08/22	T7 B7: 9000 L T4 Gasohol 95: 9000 L T3 E20: 4000 L	T7 B7: 15000 L T4 Gasohol 95: 6000 L T3 E20: 6000 L T2 Gasohol 91: 6000 L
27/08/22	No suggestion	No order
26/08/22	T7:B7 10000 L T4 Gasohol 95: 6000 L T2 Gasohol 91: 6000 L	T7 B7: 15000 L T 6 Diesel: 4000 L T4 Gasohol 95: 6000 L T2 Gasohol 91: 6000 L
24/08/22	No suggestion	No order
23/08/22	T7:B7 9000 L T4:Gasohol 95 5000 L T3:E20 4000 L T2:Gasohol 91 5000 L	T7 B7: 15000 L T4 Gasohol 95: 6000 L T3:E20 4000 L
22/08/22	No suggestion	No order
21/08/22	T7 B7: 9000 L T4 Gasohol 95: 9000 L	T7 B7: 16000 L T4 Gasohol 95: 6000 L

Date	Recommend order	Actual order
	T2 Gasohol 91: 4000 L	**T2 Gasohol 91: 6000 L**
20/08/22	**No Suggestion**	**No order**

Chapter 6:

6.1 Conclusion

Our project will help the users' life more easily when managing wet stock. It doesn’t only recommend which and how much amount of fuel to be ordered but the user can also view the amount of fuel left in each tank, which tank is the best seller in a week, which tank is almost out of tank with calculate amount of data the fuel is going to run out. So there won’t be a problem of over ordering, the amount of fuel in the tank less than 10% or the calculation of fuel ordering should be more than 22000 Liter. Furthermore, we use a line chat bot which is easy to communicate with users and also cost savings.

6.2 Future Work

In the future, we can increase the accuracy recommendation by trying other algorithms or improve and add more cases for the rule based algorithm. Moreover, we can also recommend fuel order based on the fuel market price so that it can help the user to reduce their budget and increase income when ordering fuel. Furthermore, we can also try to get the real time volume of fuel in the tanks. So the user can see the real amount of fuel anytime.

References

[1] Recommendation System

Recommendation System -Understanding The Basic Concepts

[2] Expert System

Expert Systems in Artificial Intelligence - Javatpoint

[3]UI/UX Design Priciple

https://www.justinmind.com/ui-design/list

[4]Heroku Deploy with Git

https://devcenter.heroku.com/articles/git#deploy-your-code

https://devcenter.heroku.com/articles/git#detach-from-the-build-process

[5] Vedeer-Root Web Interface

https://essotheone.thaiddns.com:4433/#TankOverView

[6]Puppeteer

https://pptr.dev/