PetroTrack: Fuel Tracking System (Web-Based Project)

BY

Farhan Tanvir
ID: 211-15-14668

FINAL YEAR DESIGN PROJECT REPORT

This Report Presented in Partial Fulfillment of the Requirements for the
Degree of Bachelor of Science in Computer Science and Engineering

Supervised By

Ms. Sharmin Akter
Assistant Professor
Department of CSE
Daffodil International University

Co-Supervised By

Ms. Nusrat Khan
Lecturer
Department of CSE
Daffodil International University

DAFFODIL INTERNATIONAL UNIVERSITY
DHAKA, BANGLADESH
JANUARY 12, 2025

APPROVAL

This Project titled **"PetroTrack: Fuel Tracking System"**, submitted by **Farhan Tanvir**, ID No **211-15-14668** to the Department of Computer Science and Engineering, Daffodil International University, has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of B.Sc. in Computer Science and Engineering and approved as to its style and contents. The presentation has been held on **12/01/2025**.

BOARD OF EXAMINERS

Dr. Md. Zahid Hasan — **Chairman**
Associate Professor
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Dr. Md Alamgir Kabir — **Internal Examiner**
Assistant Professor
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

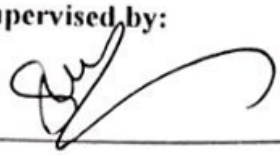
Mr. Tanvirul Islam — **Internal Examiner**
Lecturer
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Dr. Md. Manowarul Islam — **External Examiner**
Associate Professor
Department of Computer Science and Engineering
Jagannath University

DECLARATION

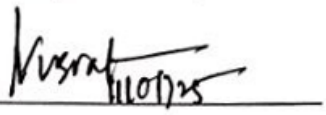
I hereby declare that, this project has been done by me under the supervision of **Ms. Sharmin Akter, Assistant Professor, Department of CSE,** Daffodil International University. I also declare that neither this project nor any part of this project has been submitted elsewhere for award of any degree or diploma.

Supervised by:



Ms. Sharmin Akter
Assistant Professor
Department of CSE
Daffodil International University

Co-Supervised by:



Ms. Nusrat Khan
Lecturer
Department of CSE
Daffodil International University

Submitted by:

Farhan Tanvir
ID: 211-15-14668
Department of CSE
Daffodil International University

ACKNOWLEDGEMENT

This work would not have been possible without the support and contributions of many individuals over the past two semesters. We are deeply grateful to everyone who has assisted us in one way or another.

First, we express our heartfelt thanks and gratefulness to the almighty for His divine blessing making it possible for us to complete the **Final Year Design Project (FYDP)** successfully.

We are grateful and wish our profound indebtedness to **Ms. Sharmin Akter, Assistant Professor,** Department of Computer Science and Engineering, Daffodil International University, Dhaka, Bangladesh. Deep knowledge and keen interest of our supervisor in the field of **"Software Development"** to carry out this project. Her endless patience, scholarly guidance, continual encouragement, constant and energetic supervision, constructive criticism, valuable advice, reading many inferior drafts, and correcting them at all stages have made it possible to complete this project.

We would like to express our heartfelt gratitude to the Head of the Department of Computer Science and Engineering, for his kind help in finishing our project and also to other faculty members and the staff of the Department of Computer Science and Engineering, Daffodil International University.

We would like to thank our entire course-mates at Daffodil International University, who took part in this discussion while completing the coursework.

Finally, we must acknowledge with due respect the constant support and patience of our parents.

ABSTRACT

The "PetroTrack: Fuel Tracking System" is an innovative solution meticulously designed to transform the landscape of fuel management across various industry sectors. It employs cutting-edge technologies such as Node.js, Next.js, React.js, and MongoDB to provide real-time monitoring and precise analysis of fuel consumption patterns. The primary goal of this system is to encourage responsible fuel use, enhance operational fuel efficiency, and significantly reduce costs by offering tailored insights and integrating effortlessly with existing fleet management frameworks. The document provides an in-depth examination of the system's sophisticated technological architecture, detailed hardware configuration, and extensive testing protocols. The system is projected to deliver substantial benefits, including superior decision-making capabilities, significant reductions in operational expenses, and the promotion of environmentally sustainable fuel management practices. These attributes mark it as an essential advancement for industries heavily reliant on fuel resources, aiming to optimize resource utilization and sustainability.

Table of Contents

List of Figures

List of Tables

Chapter 1
Introduction

1.1 Overview

Through the increasing rate of technological advancement, industries across the globe are being impacted and fuel management isn't any different. With rising energy costs and greater consciousness toward the same, effective fuel management is imperative for businesses relying heavily on fuel resources. Introduction of the "Fuel Tracking System" web-based application in order to digitize the way fuel consumption is tracked and managed for fuel stations and suppliers. Using real-time monitoring, sophisticated data analysis, and user-centric UI, this system aims to increase operational efficiency, minimize waste, and supplement responsible consumption patterns. This project serves as an innovative approach to overcoming traditional challenges in fuel management including inefficiency, lack of transparency and time-consuming manual processes.

1.2 Motivation

Fuel stations and suppliers run into a variety of problems every day such as inventory mismanagement, gaps in fuel supply, and wastage that leads to increased operational costs. Such these methods of fuel management are often manual tracking and consistently prone to random errors and inefficiencies. Additionally, the increasing adoption of digitization and real-time solutions across industries has created a demand for a more efficient and automated fuel management system. When designing this system, the main motivation was to develop a solution to these operational issues by introducing a solution that would empower stakeholders with data-driven analytics insights, to facilitate efficient decision-making. The project also aims to encourage the development of better fuel management practices for cost and environmental sustainability.

1.3 Problem Statement

Balancing solution for fuel stations, and suppliers fuel management is key point of operational efficiency. But conventional methods tend to cause inefficiencies like improper

inventory management, excessive fuel consumption, and delays in their operations due to shortage of supplies. Moreover, without real-time monitoring tools and advanced analytics to predict trends and optimize resource allocation, these problems compound further. Without a reliable and friendly system, they find it challenging to maintain their service levels or minimize disruptions. The broader, web-based platform of the "Fuel Tracking System" can then fill in these gaps, allowing for real-time monitoring, data analysis, and integration with existing fleet management systems.

1.4 Objectives

The main goals of the Fuel Tracking System project are:

- **Functions Automated**: Develop a web interface where suppliers and fuel stations get auto alerts regarding the availability of fuel.
- **Minimize disruptions:** Reduce order fulfillment disruptions due to fuel shortages through effective management of inventory and orders.
- **Monthly Sales Analysis:** Deliver comprehensive visualizations and reports for the fuel station owners to analyze monthly sales and performance.
- **Improved Fuel Usage:** Encourage fuel efficiency through monitoring and personalized recommendations.
- **Data Analysis:** Use algorithms to analyze historical consumption data and identify trends for optimization.
- **Cost Efficiency:** The algorithm enables the identification of gaps in these utilizations that could lead to halting the production, as a result minimizes the wastage as well as reduced the operational costs.
- **Seamless Integration:** Ensure compatibility with existing systems ensuring seamless data flow and enhancing its functionality and sharing capabilities.
- **User-Friendly Design**: To facilitate ease of data interpretation, develop an intuitive interface with customizable reports and visualizations along with real-time alerts.
- **Integration**: Integrate well with the current fleet system to synchronize the functions and data sharing.

1.5 Project Outcome

The expected results that have to be delivered by the "Fuel Tracking System" are:

- **Increased Efficiency:** Assist fuel station employees and businesses with optimizing inventory and order management tools for reduced operational delays and disruptions.

- **The second solution will focus on cost savings**: By realizing where fuel is being wasted through advanced analytics and real-time monitoring, station owners and suppliers can reduce costs dramatically.

- **Developed Decision-Making:** Utilize complete data analysis and trend forecasting to provide stakeholders with actionable insights.

- **Customer Value:** Provide a user experience for station owners and staff that simplifies operations with visualizations, customizable reports, and real-time alerts.

- **Environmental Sustainability:** This can minimize carbon emissions and environmental influence, promote responsible fuel consumption habits.

- **Live Tracking:** Allow suppliers and petrol stations to track sales and fuel availability in real-time to prevent any service interruption and improve resource planning.

The next crucial step is convincing the widest possible audience to adopt digital fuel-saving technologies.

1.6 Organization of the Report

This report is organized in six chapters, with the following headings, each covering a different aspect of the Fuel Tracking System project.

- **Chapter 1: Introduction**
  This chapter provides an overview of project motivation, problem statement, goals and anticipated results. This helps us to appreciate what the system is for and why it matters.

- **Chapter 2: Background**

This chapter reviews the literature on fuel management systems and examines related technologies and previous solutions. It likewise outlines the limitations of existing systems, showing why the proposed solution is needed.

- **Chapter 3: Methodology**
  The development of the Fuel Tracking System, including its design, implementation, and evaluation, is presented in this chapter. There are specifics on the tools, techniques and frameworks used throughout the project lifecycle.

- **Chapter 4: Implementation and Testing**
  This chapter presents the technical realization of the system, its architecture, core features and user interface. It also describes the outcomes of conducting tests and evaluations on the system's performance.

- **Chapter 5: Standards in Engineering and Design Obstacles**
  This chapter presents the engineering standards that were followed for the project and will also discuss the challenges that came during the design phase. It also points up how these challenges were overcome.

- **Chapter 6: Conclusion**
  The project is summarized in this chapter, with its contributions, accomplishments and possibilities for enhancement in the future. It ends with considerations on the Fuel Tracking System's system-wide impacts and implications.

1.7 Summary

In summary, PetroTrack is designed to transform fuel management at petrol stations by addressing critical issues such as inaccurate fuel tracking and the lack of real-time data. Utilizing advanced web technologies and data analytics, the system offers real-time monitoring, automated fuel ordering, and detailed consumption analysis. This chapter has introduced the project's background, motivation, problem statement, objectives, scope, limitations, and the structure of this report. The subsequent chapters will explore the development process, system architecture, and implementation details, providing a thorough overview of the project and its impact on operational efficiency and environmental sustainability.

Chapter 2
Background

2.1 Overview

In recent years, there has been a growing demand for real-time fuel management systems, especially in sectors where the efficient and reliable tracking of resources is essential. The evolution of web-based applications has facilitated the addressing of these needs by providing users with instant access to data concerning fuel levels, usage statistics, and overall resource management. This chapter delves into existing literature on real-time fuel management systems, underscores notable technological advancements, and pinpoints deficiencies that the PetroTrack system seeks to address. By analyzing various methods of fuel tracking and management feedback mechanisms, we gain deeper insights into how PetroTrack can enhance operational efficiencies at petrol stations.

2.2 Literature Review

This makes web-based fuel tracking systems highly reliable for many businesses, such as those from the transportation and logistics, construction, and power generation sectors. With the help of advanced hardware and software technologies, these systems can monitor fuel levels, detect fuel theft, and provide real-time fuel data analysis, which improve operations and reduce fuel acquisition losses at scale.

Fuel theft has always been a huge concern for organizations involved in fuel-intensive industries. To combat this issue advanced fuel tracking systems, utilize a multi-faceted approach to theft detection. At the heart of these systems is the integration of high-precision sensors. For instance, the use of ultrasonic sensors employs sound waves to precisely measure fuel levels in the tanks and can also detect unauthorized withdrawals (Tatababu et al, 2024) [1, 2]. In contrast, capacitive sensors work by measuring capacitance in the tank, allowing real-time reporting of levels, and any anomalies. These embedded sensors fitted inside the fuel storing tanks and dispensers constantly record its fuel inventory giving very accurate and real-time readings, making it easier and faster to determine if there has been any attempt at theft.

The systems also include real-time alerts to enable a quick response in case of any suspicious activity. LTFB: GSM modems provide SMS or phone call alerts to authorized personnel as abnormal fuel level changes or unauthorized access is detected (Kadu, 2023) (Tatababu et al., 2024) [1, 2]. Additionally, web-based interfaces offer visual and auditory notifications for immediate intervention and response by authorized personnel. These alerts can be set to notify appropriate personnel when something goes awry, potentially even

specifically when fuel levels drop dramatically, unauthorized access is detected, or activity occurs at strange hours, to name a few examples.

Remote monitoring capabilities are essential for good fuel management. Web-based interfaces offer users a single platform where they can access real-time data regarding fuel levels, consumption trends, and other key parameters. However, these interfaces usually have interactive dashboards illustrating essential performance indicators (KPIs), including fuel consumption rates, stock levels, or historical data. Reports on usage, order history, and cost analysis can be generated for additional decision-making intelligence. In addition, remote access controls allow the authorized personnel to remotely monitor fuel dispensers, control fuel access accordingly, and change system settings whenever required [4].

Today's fuel tracking systems provide a variety of comprehensive fuel management functions. These include:

- **Inventory Management:** It keeps the record of the fuel inventory levels accurately giving a wide range of businesses to control stock levels, reduce wasteful stock, or stockouts. It also helps to dramatically mitigate the chances of having not enough fuel, allowing operations to carry on uninterrupted.
- **Order Management:** Facilitating the process of ordering fuel, from bulk order placement and delivery tracking to invoice management. This can lead to greater efficiency, lower administrative costs, and better communication between consumers and suppliers of fuel.
- **Cost Management:** Generating cost analysis reports, guiding fuel waste detection, and optimizing fuel purchasing strategies. Businesses can use this information to pinpoint and correct discrepancies in fuel consumption, generating considerable cost savings.
- **Compliance Management:** Compliance with applicable rules and industry standards pertaining to fuel management and transport. “If they are compliant, that saves them from penalties and gives them good standing.”

On the other hand, there are important challenges that need to be considered for successful implementation of web-based fuel tracking systems.

- **Protecting Data:** Security and privacy of sensitive data like fuel consumption records and user credentials are critical. Cyber threats: Effective cyber security implementations need to be adopted based on" the assessed threats to information assets and detailed in the applicable security policies.
- **User Training and Adoption:** Making sure users receive sufficient training on how to operate the system and understand the data is vital to getting the most out

of it. Straightforward and concise user manuals together with follow-on support and training sessions can ensure user adoption and help reduce frustration.

- **Maintenance & Support:** Regular maintenance of the system, software updates, and continuous technical support are needed to ensure the system's continued processing capabilities and reliability. It may involve technical issue resolution, system upgrades, and providing user support (Liang, 2024) [6].

Overall, web-based fuel tracking systems are a major step forward in fuel management technology. With the help of advanced sensors, real-time data analysis, and robust security measures, these systems help businesses streamline operational efficiency, reduce losses, and optimize fuel management initiatives. With technology evolving, we will ever more sophisticated and integrated solutions emerge to further optimize the consumption of fuel and help to achieve a more sustainable future.

2.3 Gap Analysis

The intent of this gap analysis is to address gaps in existing fuel management systems and articulate the value add that the Fuel Tracking System provides. The analysis identifies pressing gaps filled by the proposed solution, drawing on an examination of existing practices, tools, and technologies.

Inefficient Manual Processes

1. **Current Situation:** There are manual tracking systems in place for many fuel stations where human errors are likely to occur leading to delays and inaccuracies in inventory management.
2. **Fitness for Purpose:** No automated approach for fuel tracking and reporting.
3. **Proposed Solution:** The Fuel Tracking System simplifies and automates inventory monitoring while using real-time data instead of manual.

Limited Real-Time Monitoring

1. **Current State:** Existing systems have no capability to give real-time monitoring of fuel levels & consumption. That causes delays in responding to shortages or anomalies.
2. **Announced Need:** Lack of real-time alerts and insights for proactive decision-making.
3. **Solution:** Real-time data tracking and alerts allow stakeholders to take necessary action in real time, preventing any interruptions in operations.

Minimal Use of Advanced Analytics

1. **Current Status:** Most of the systems do not utilize data analytics to provide insights on fuel consumption patterns, wastage or trends.
2. **Gap:** Lack of data-driven decision-making tools.
3. **Proposed Solution:** Using machine learning algorithms to analyze large datasets to identify patterns and produce better forecasts, optimizing resource allocation and capacity management based on analysis.

Not Enough Integration Capability

1. **Current State:** The majority of fuel management solutions available today are standalone systems making it difficult to integrate into fleet management or other operational platforms.
2. **Gap Identified:** Integration with existing management systems is not seamless.
3. Proposed Solution: Fuel Tracking System Feature Compatibility with Other Platforms

User Experience Challenges

1. **Current State:** Most existing systems have yet to provide interfaces that are easy to use or features that user can configure according to their personal needs, leading to low adoption.
2. **Identified Gap**: Lack of usability and customization
3. **Proposed Solution:** The system provides a responsive, user-centric design built with customizable dashboards, visualizations, and reports based on user requirements.

Ignoring Environmental Sustainability

1. **Current State:** Environmental benefits, like responsible fuel consumption or carbon output reduction, are rarely highlighted by existing systems.
2. **Gap in the System:** Sustainability practices are not being focused on.
3. **Proposed Solution**: Ensuring a cleaner and greener environment is a vital goal in today' s world.

To better understand the strengths and weaknesses of existing Fuel tracking solutions, a comparison of several international platforms services is presented below in table 2.3.1. This comparison highlights key features, strengths, and downsides, providing valuable insights for enhancing the Petrotrack application.

Table 2.3.1: Comparison between existing works

Features	**Simple fuel calculator**	**Spritmonitor**	**PetroTrack (Proposed)**
Tracking fuel level	Yes	Yes	Yes
Managing fuel pump	No	No	Yes
Track suppliers fuel level	No	No	Yes
Order directly from suppliers	No	No	Yes
Suppliers fuel track record	No	No	Yes

This table demonstrates that although numerous systems provide real-time tracking and some ancillary functionalities, they often fail to offer a fully-rounded user experience. Problems like inconsistent tracking accuracy, inadequate communication tools, and antiquated user interfaces are prevalent among current platforms. These observations are instrumental in shaping the development of PetroTrack, directing the incorporation of vital features and enhancements to deliver a dependable, user-friendly fuel management solution specifically designed to meet the demands of petrol station operations.

2.4 Open Issues

Despite advancements in real-time tracking and user feedback systems, several open issues remain:

- **User Interface Complexity:** Achieving a balance between offering a full suite of features and maintaining user-friendliness is a continual challenge. Many systems either simplify the data too much, losing valuable insights, or present it in a complex manner that may overwhelm users.
- **Data Privacy and Security:** As the system collects and processes real-time data, ensuring the security and privacy of this information remains a critical issue, especially with increasing regulatory scrutiny.

- **Scalability:** As the number of users and the volume of data increase, ensuring the system remains performant and responsive is challenging.
- **Operator Familiarity with Technology:** Some station operators may not be familiar with digital solutions, which can limit their ability to effectively utilize the system.

Addressing these open issues is crucial for developing a robust, user-centric transportation solution.

2.5 Summary

This chapter has reviewed the present landscape of research and development in real-time fuel tracking and management systems. Although there have been considerable advancements, notable gaps still exist. Current solutions frequently lack holistic approaches that blend real-time monitoring with advanced AI-driven analytics for fuel usage and operational efficiency. By pinpointing these gaps and unresolved issues, we gain a clearer insight into the areas where PetroTrack can introduce innovations and offer a more integrated, user-friendly solution for managing petrol station operations.

Chapter 3
Research Methodology

3.1 Requirement Analysis & Design Specification

This phase defines the app's requirements and creates detailed system designs to ensure effective development.

3.1.1 Overview

A clear vision is vital for the development of the Fuel Tracking System. Requirement analysis: understanding functional and non-functional requirements followed by design specification: diagramming system architecture → components of user interface. This was done using the MERN (MongoDB, Express. js, React. js, Node. js) stack allows rapid development and high scalability. Code editing was done using Visual Studio Code for streamlined programming, and the application was deployed in localhost environment for testing and validation.

3.1.2 System Design

The Fuel Tracking System employs a modular architecture to ensure it is maintainable, scalable, and easily integrated with other systems. Here are the main components identified during the design:

Frontend (User Interface)

- Developed using React. js, the front end guarantees an amicable and intuitive interface.
- Data updates: up-to-date fuel level displays, dashboards, and customized reports
- Responsive design makes your site compatible with different devices, desktop or mobile.

Backend (Server Logic)

- Built using Node.js with Express.js framework that allows you to create APIs fast.
- Manages data processing, user authentication, and communication between the frontend and the database.
- Utilizes RESTful APIs for seamless interaction with frontend

Database

- The database was MongoDB, where all system information was stored, such as user accounts, fuel inventory and logs.
- This delta between schema design and data flow goes in both directions: on one hand schema design guarantees best query performance and data consistency.

Deployment

- Localhost (for running during development to test and verify that all parts are working without error before production deployment).
- During development, we used Postman to test and Git for version control.

Integration and Communication

- Built to integrate directly with your fleet management systems through APIs.
- The backend also provides real-time monitoring, notifications, and alerts to keep stakeholders proactively informed.

3.1.3 Hardware / Software Requirement

Hardware and Software configurations used in the development and deployment of the Fuel Tracking System:

Hardware Requirements

1. Development Machine:
   - Processor: Intel Core i5 or better
   - RAM: 8 GB or higher
   - Storage: 256 GB SSD or higher
   - Based OS: Windows 10 or macOS or Linux

2. Client Devices:
   - Any device capable of running a modern web browser (e.g., Chrome, Firefox, Edge)
   - Smartphones, tablets, or PCs for accessing the system

Software Requirements

Development Tools:

- Visual Studio Code: Code editor for development.
- Postman: For API testing.
- Git: Version control system.

Technology Stack:

- Frontend: React.js
- Backend: Node.js with Express.js
- Database: MongoDB

Testing Environment:

- Localhost server: For testing and debugging.
- Browser Developer Tools: For frontend debugging and testing responsiveness.

Additional Libraries and Frameworks:

- Mongoose: For MongoDB interaction.
- Axios: For API calls from the frontend.
- React Router: For navigation within the application.
- Chart.js or D3.js: For data visualization in the dashboard.

3.1.4 Use Case Diagram Description:

- **Registration & Login:** Users will register for an account or log in to the system.
  All Business Knowledge Articles: the articles containing all articles from business knowledge.
- **Order to Supplier:** Process which reflects the placing of order to fuel suppliers.
- **Fuels Management:** Enables management of fuel types, availability, and pricing.
- **Orders**: Track and manage orders with suppliers.
  Upon clicking on this icon, it allows you to Create/Edit and Manage Supplier.
- **Manage Stations:** Allows you to manage the fuel stations themselves, including their location, inventory, and so on, and assigned staff.

User Roles:

- **Admin/Station Owner:** Has access to all functionalities, including managing orders, suppliers, and fuel stations.
- **Suppliers:** Can view and manage orders placed by stations.
- **Includes:** Some functionalities are included in the Admin/Station Owner or Suppliers roles

Figure 3.1.4.1 below displays the Petrotrack system's use case diagram.

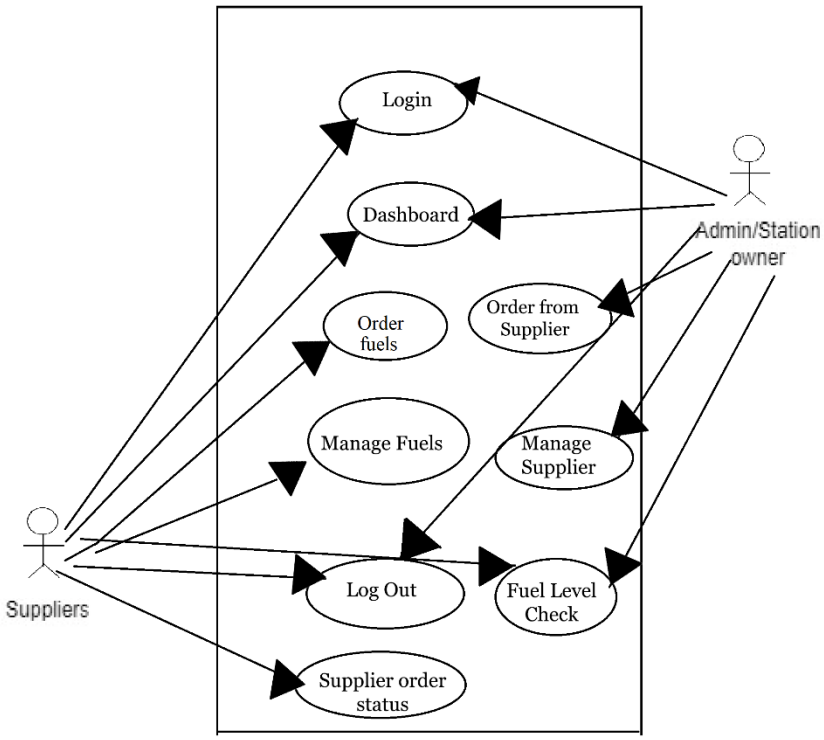


Figure 3.1.4.1: Use Case Diagram

3.1.5 Class Diagram Description:

Key Entities:

- **Admin/User:** Represents users with roles like admin and regular users.
- **Order:** Represents an order placed by a user.
- **Product:** Represents a product with details like name, price, and availability.
- **Supplier:** Represents a supplier of products.
- **Stock:** Represents the inventory of products.

Relationships:

- **Admin/User** can place **Orders**.
- **Orders** contain **Products**.
- **Products** are supplied by **Suppliers**.
- **Stock** tracks the availability of **Products**.

Attributes:

- **Admin/User:** id, email, password, role
- **Order:** id, details, product, p.name, p.price
- **Product:** id, name, price, availability, p.type
- **Supplier:** id, name, role
- **Stock:** id, items, number, des, details

Figure 3.1.5.1 below displays the Petrotrack system's class diagram.

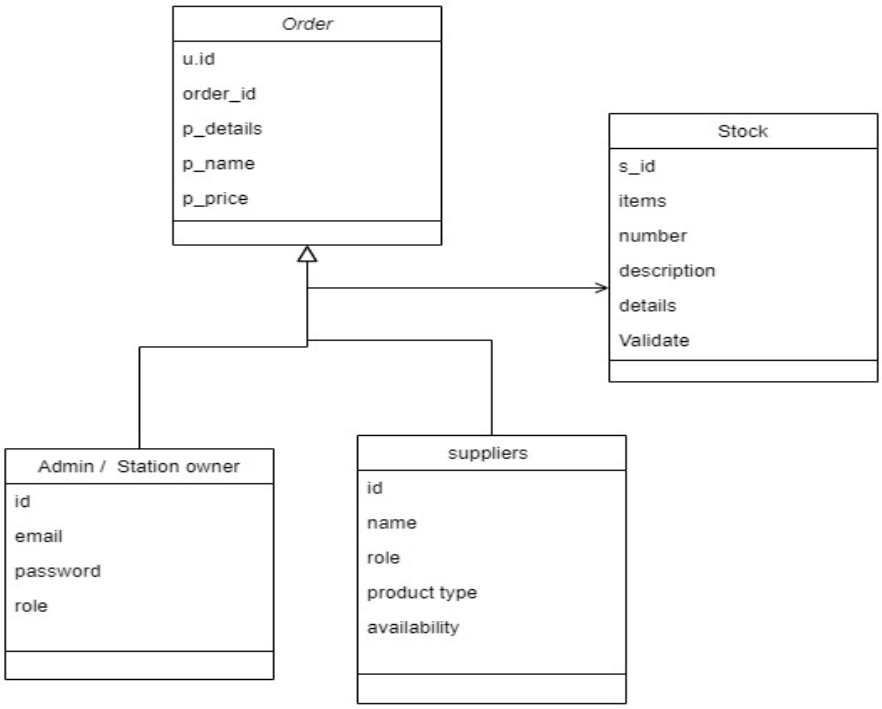


Figure 3.1.5.1: Class Diagram

3.1.6 ER Diagram:

Figure 3.1.6.1 below displays the Petrotrack system's ER diagram.

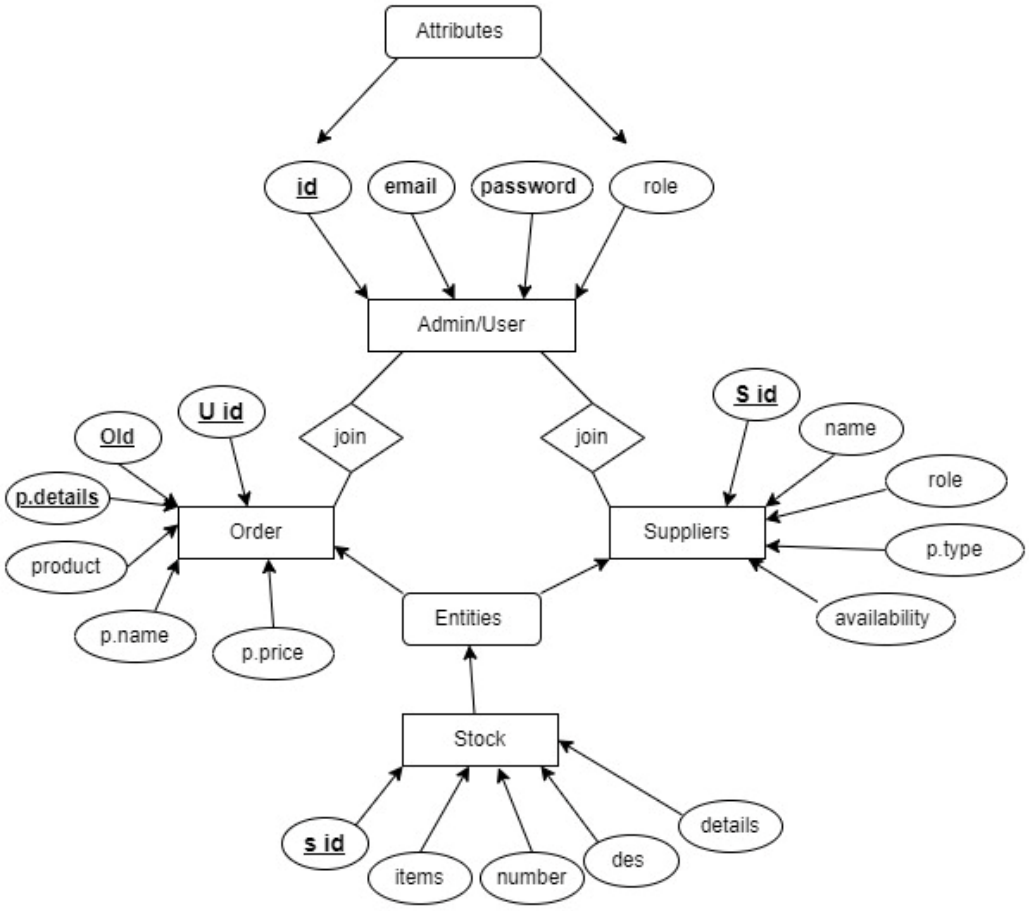


Figure 3.1.6.1: ER Diagram

Key Entities:

- **Admin/User:** Represents users with roles like admin and regular users.
- **Order:** Represents an order placed by a user.
- **Product:** Represents a product with details like name, price, and availability.
- **Supplier:** Represents a supplier of products.
- **Stock:** Represents the inventory of products.

Relationships:

- **Admin/User** can place **Orders**.
- **Orders** contain **Products**.
- **Products** are supplied by **Suppliers**.
- **Stock** tracks the availability of **Products**.

Attributes:

- **Admin/User:** id, email, password, role
- **Order:** id, details, product, p.name, p.price
- **Product:** id, name, price, availability, p.type
- **Supplier:** id, name, role
- **Stock:** id, items, number, des, details

Activity Diagram:

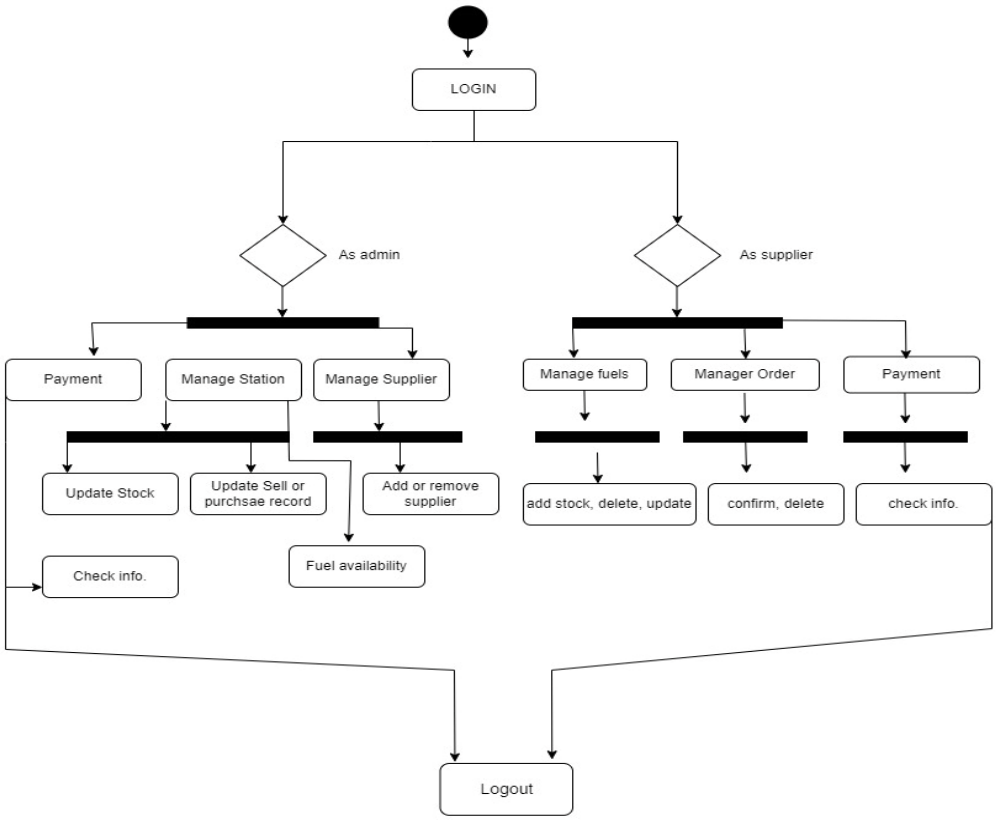


Figure 3.1.6.2: Activity Diagram

Key Components:

- **Login**: The starting point where users authenticate themselves.
- **As Admin/As Supplier:** Points that decide what are the roles of the user and to what level do they have access.
- **Station Management**: This involves managing the stations by updating the stock and sales records.
- **Manage Supplier:** This feature allows you to manage your suppliers, including adding and removing suppliers.
- **Manage Fuels:** For managing the fuel types, adding stock, deletion, updating.
- **Manage Order**: able to manage orders from viewing the details of an order to confirm and delete the order.
- **Payment**: Handling payment transaction related functionality.
- **Logout**: Consumer Logout Point

The diagram shows the different components that interact with each other For example, Once the order is made, the fuel ordering component interacts with the inventory which reduces the total inventory and reflects the same into the supplier component informing it about a new order placed.

- **Dataflow Diagram:**

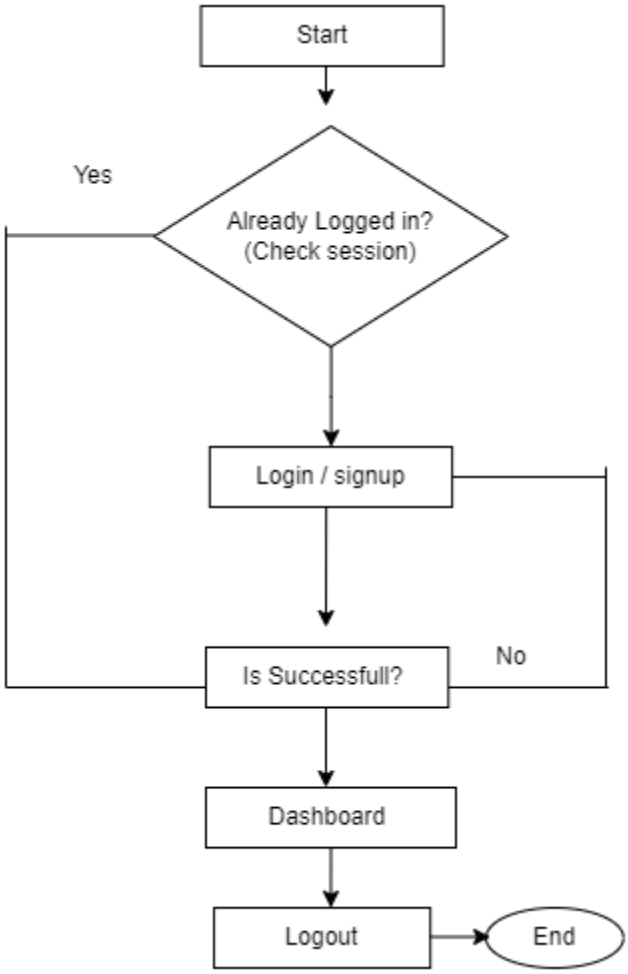


Figure 3.1.6.3: Dataflow Diagram

This is a flowchart which shows user authentication and login process of an application.

Steps:

Start: The process begins.

Verify Session: Whether the user is already logged in.

Yes (Logged In): Redirect the logged in user to dashboard directly

No (Not Logged In) — If a user is not logged in, the user is prompted to login or signup.

You are processed on data until October 2023.

Success/Fail: This one simply checks the credentials and tell if the login or signup is successful

Success If success, the user will be redirect to dashboard.

Fail: The user is probably informed via on-screen error message and must retry.

Dashboard: The user can access and interact with the application(s).

Logout: Allows the user to log out of the application and terminate the session.

In summary, the flowchart illustrates how to access the application while also ensuring that users are authenticated.

- **Admin-sequence Diagram:**

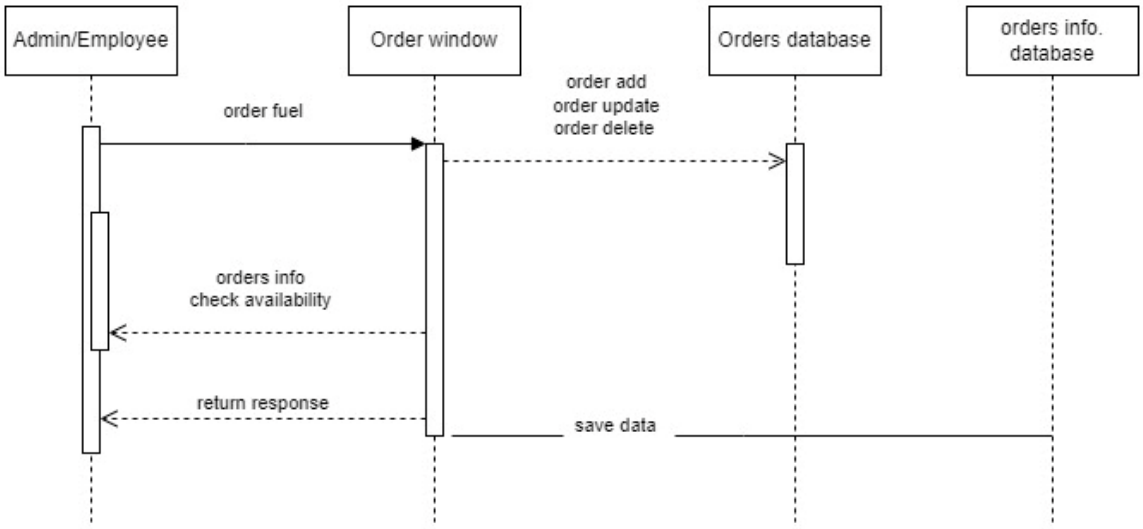


Figure 3.1.6.4: Admin-sequence Diagram

Key Components:

- Admin/Employee: The person who submits the order.
- Interface through which the order is placed.
- Orders Database: The database where order information is stored.

- Orders Info. Database: The database containing information about available fuel and prices.

Process Flow:

- Order Fuel: The Admin/Employee sends the order using the Order Window.
- Query The Orders Info: This queries Orders Info. List of available fuel and price information databases?
- Return Response: Order Window shows to Admin/Employee the availability and cost price.
- Store Data: When the order is confirmed, the order details are stored in Orders Database by the system.
- Finally, this diagram makes it clearer how the user and the order interface are interacting with the databases that are needed for placing and processing a fuel order.

- **Supplier Sequence Diagram:**

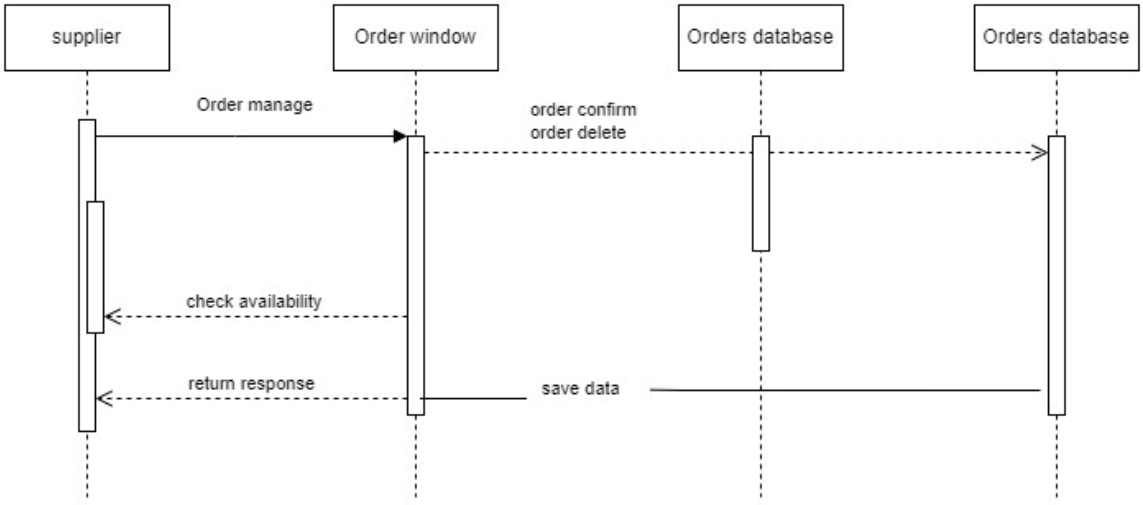


Figure 3.1.6.5: Supplier-sequence Diagram

Key Components:

- **Supplier:** The entity responsible for managing orders.
- **Order Window:** The interface through which orders are managed.

- **Orders Database:** The database where order information is stored.

Process Flow:

1. **Order Manage:** The Supplier initiates order management operations through the Order Window.
2. **Order Confirm/Delete:** The Supplier confirms or deletes orders using the Order Window.
3. **Check Availability:** The system queries the Orders Database to check order availability.
4. **Return Response:** The Order Window displays the availability information to the Supplier.
5. **Save Data:** The system updates the Orders Database with the confirmed or deleted order information.

In short, the diagram illustrates how the Supplier interacts with the order interface and the database in order to manage orders.

3.1.7 UI Design:

User Interface (UI) design maintains minimalism, accessibility, and ease-of-use for various types of users of the Fuel Tracking System—station owners, staff, and suppliers, for example.

Essential Components of the User Interface Design:

Dashboard:

1. Shows real time stats like current fuel levels, sales trends and inventory warnings.
2. Also features visual components such as graphs, charts, and gauges to help interpret the data.

Login Page:

1. A secure login system with ownership, staff, and supplier roles.
2. completely reversed to allow login with the following variables: username, password fields, forgot password option

Inventory Management Page:

1. Provides an interface for monitoring, updating and managing fuel stock.
2. Enables users to define thresholds for automated alerts and order placement.

Sales Report Page:

1. Provide Monthly & Yearly Sales Statics by Detailed Charts.
2. Features options for data export in CSV or PDF

Alerts Notifications Panel:

Emphasizes important alerts, like low inventory, pending orders, or error messages.

Order Management Page:

1. Allows suppliers to see and complete fuel orders
2. We track where an order is in processing and delivery.

Settings Page:

1. Provides user profile, notification settings, and system configuration options.
2. I created wireframes and prototypes for these pages using UI design tools such as Figma (later as Adobe XD) to make the designs responsive and user-friendly for both desktop and mobile users.

Landing Page: Figure 3.1.7.1

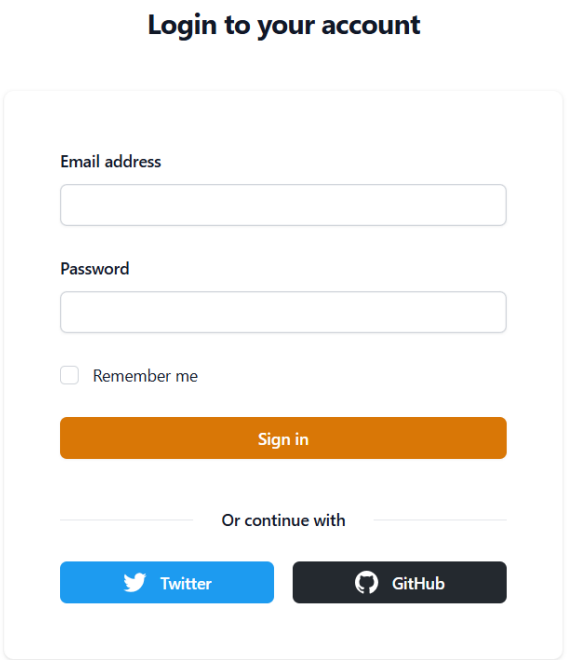


Figure 3.1.7.1: Landing Page

The homepage of the website will serve a dual purpose as both the landing page and the login portal. This consolidated design ensures that both administrators and suppliers can access their respective dashboards through a single, unified entry point. By combining these functionalities, the website aims to streamline navigation and improve user experience, making it easier for users to login and immediately engage with the platform's features tailored to their specific roles. This approach also simplifies the overall site architecture, reducing the need for multiple pages and helping users to quickly find the necessary login inputs right on the main page.

Admin Screen:

- **Home Page:** Figure 3.1.7.2

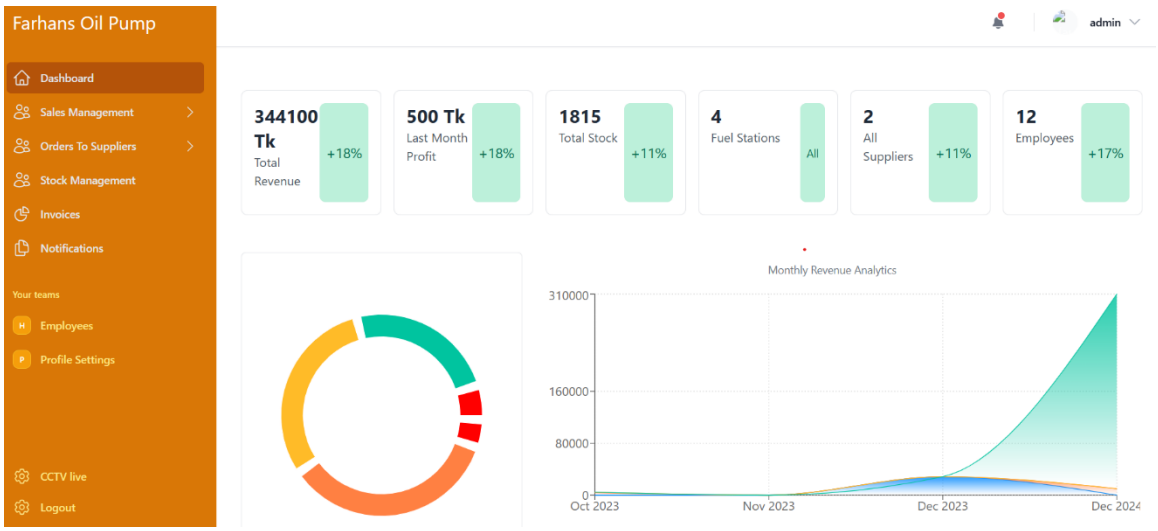


Figure 3.1.7.2: Home Page for Admin

The admin screen of the web project will prominently feature a dashboard, providing essential insights through visual charts that track sales data and total stock levels. This comprehensive view not only enables administrators to monitor the business's operational metrics but also aids in effective inventory management. By integrating these functionalities, the dashboard serves as a strategic tool to facilitate quick decision-making and to maintain oversight over critical business aspects, all within a streamlined interface.

- **Make Sales:**

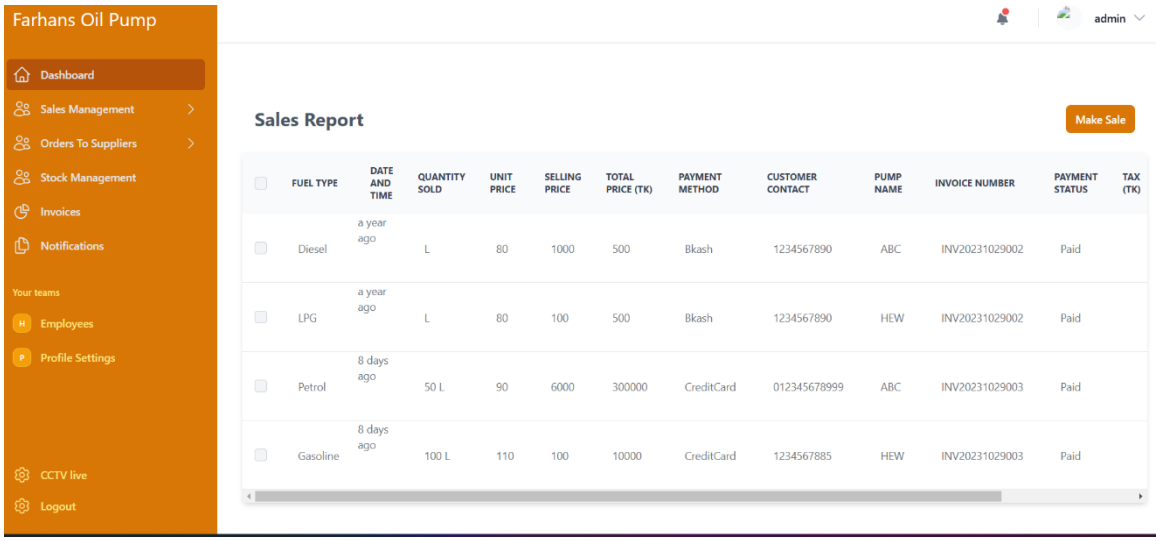


Figure 3.1.7.3: Make Sales for Admin

On the admin screen of the web project, there will be a dedicated section for making sales and generating sales reports. This feature allows administrators to actively manage transactions and access detailed reports that summarize sales activities. By providing these capabilities, the system enhances operational efficiency and gives administrators the tools needed to track sales performance and analyze trends directly from the dashboard.

- **Sales history:**

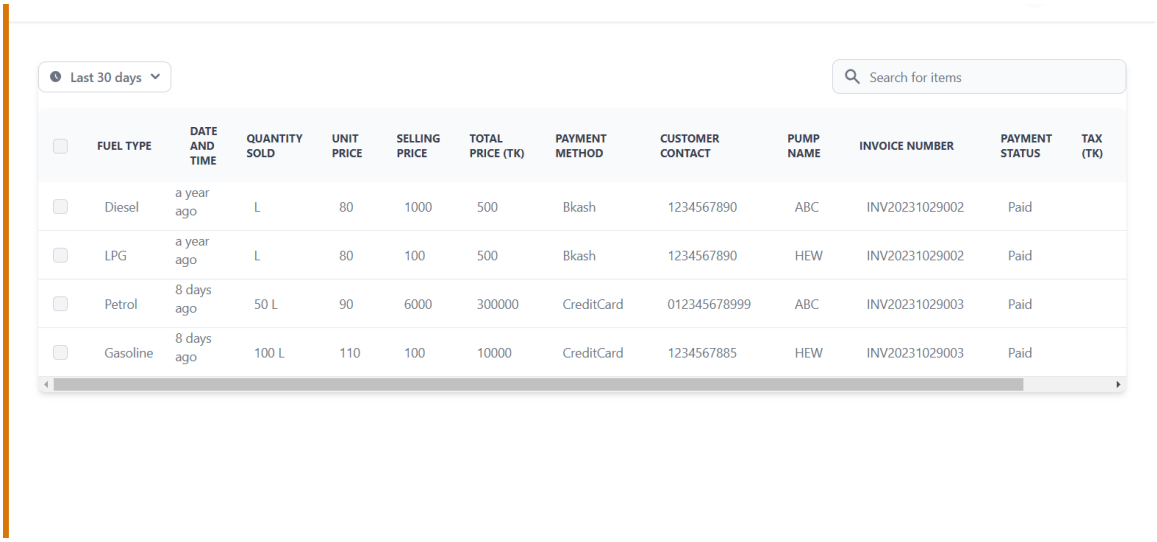


Figure 3.1.7.4: Sales History for Admin

web project will also feature a "Sales History" section. This allows administrators to review past sales transactions in detail, providing them with valuable insights into sales trends, customer behavior, and product performance over time. This historical data is crucial for strategic planning and for making informed decisions aimed at enhancing business operations and customer satisfaction.

- **Stock Management:**

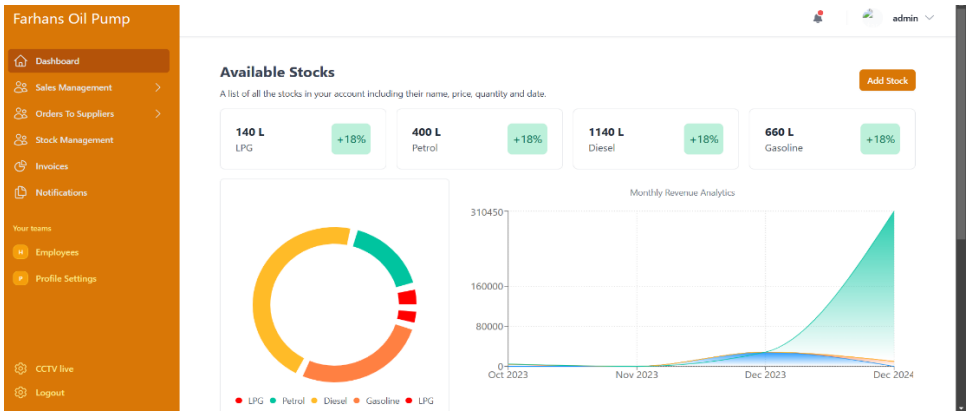


Figure 3.1.7.5: Stock Management for Admin

The admin screen of the web project includes a robust "Stock Management" section. This feature enables administrators to input and store stock information, manage inventory levels, and visualize available stock through interactive graphs. By providing a graphical representation of inventory data, the system helps administrators quickly ascertain stock statuses, facilitating better inventory control and ensuring that stock levels meet ongoing business demands efficiently.

- **Suppliers Quantity:**

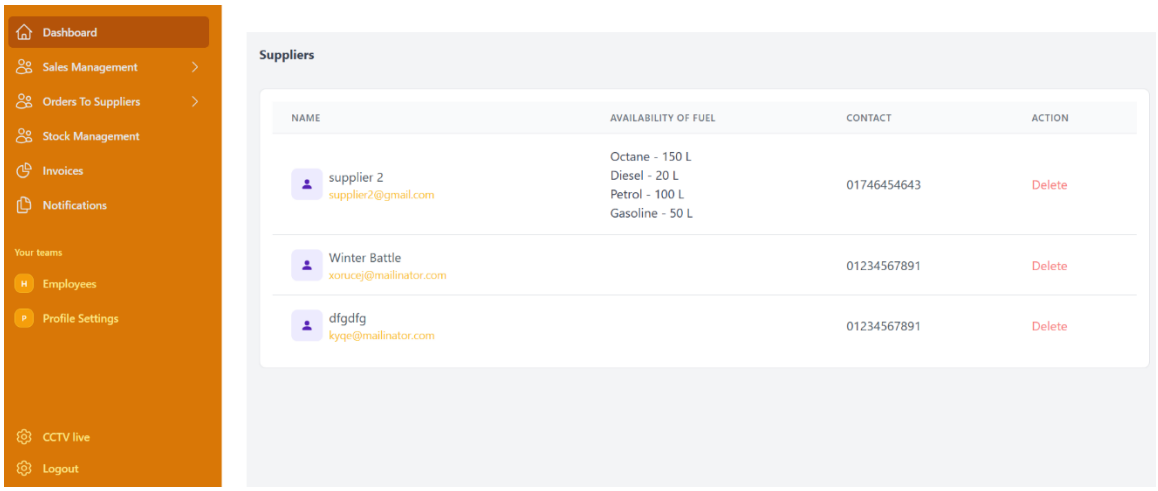


Figure 3.1.7.6: Suppliers Quantity check for Admin

The admin screen will also feature a "Suppliers Quantity" section, which allows administrators to view the quantities of fuel available with each supplier. This functionality enables the monitoring of fuel stocks across different suppliers, ensuring that the administration can manage supply levels effectively and plan for replenishment based on real-time data. This feature is crucial for maintaining a steady flow of fuel and preventing shortages.

- **Order From Suppliers:**

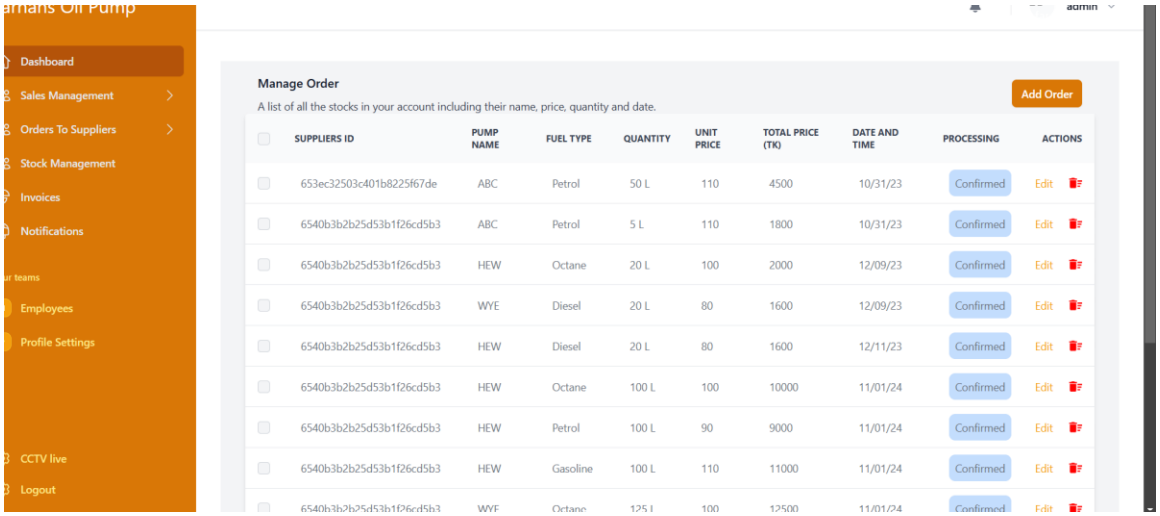


Figure 3.1.7.7: Order from Suppliers for Admin

The admin screen will include an "Order from Suppliers" section, enabling administrators to place orders for fuel directly with suppliers through the system. This feature streamlines the ordering process, ensuring that fuel supply levels are adequately maintained by facilitating timely and efficient procurement from chosen suppliers. This capability is integral to managing the supply chain effectively and ensuring that operations continue smoothly without interruptions due to fuel shortages.

- **Notification:**

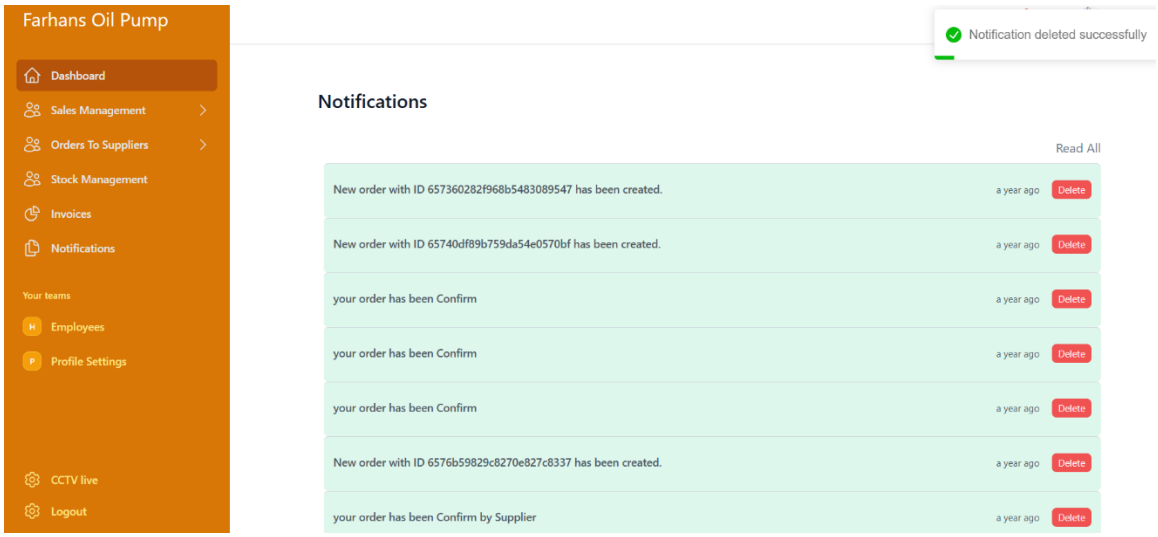


Figure 3.1.7.8: Notifications for Admin

The admin screen will feature a "Notification" section, which will provide pop-up notifications to administrators immediately after sales are made. This ensures that administrators are promptly informed of all sales transactions, enhancing the ability to monitor and respond to sales activity in real time. This feature is designed to keep the administrative team updated and engaged, allowing for better oversight and management of daily operations.

- **Employees:**

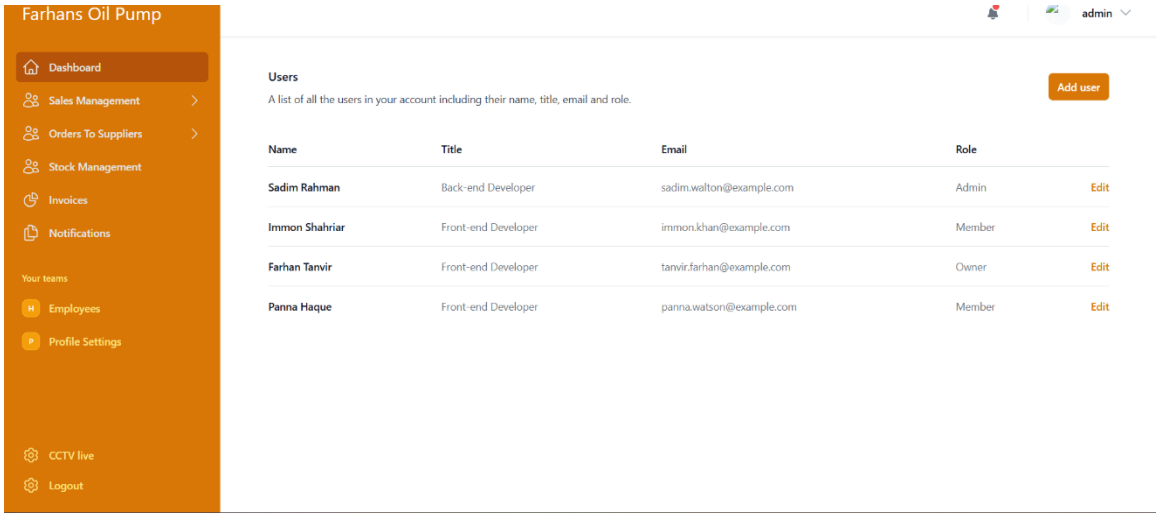


Figure 3.1.7.9: Employees check for Admin

The admin screen will include an "Employees" section, where administrators can manage details and roles of the employees. This feature allows for the addition, modification, and review of employee information, including job roles, contact details, and performance data. By centralizing employee management, the system aids in streamlined human resource processes, ensuring that all personnel are effectively organized and their contributions accurately tracked within the organization.

- **Personalized:**

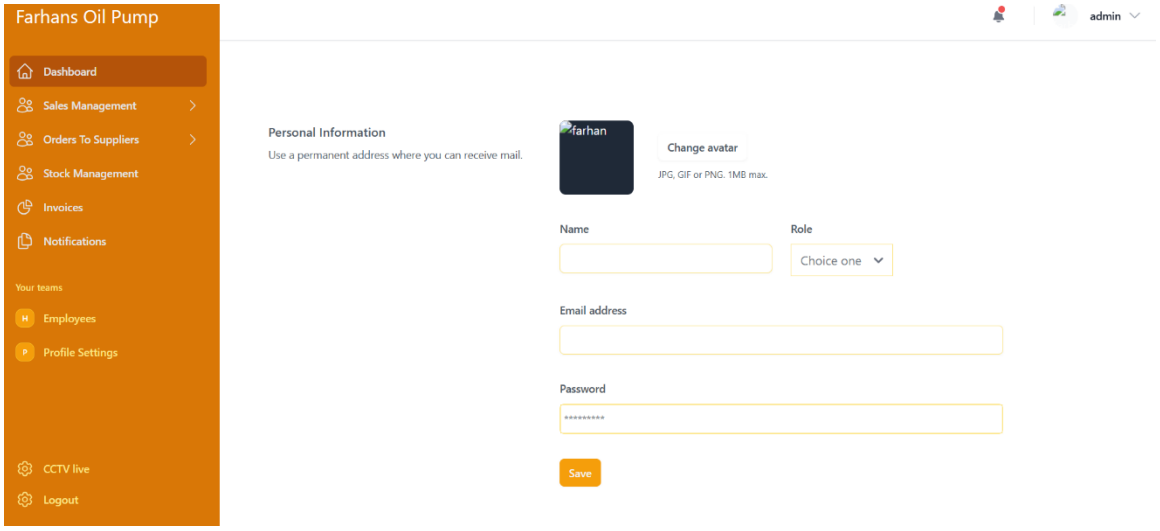


Figure 3.1.7.10: Personalize for Admin

The admin screen will feature a "Personalized" section, where users can edit their names and profile images. This customization is designed for identification purposes within the system, allowing users to maintain up-to-date personal information that enhances user interaction and identification across the platform. This personal touch not only improves the user experience but also helps in maintaining accurate and current data for communication and administrative purposes.

- **CCTV:**

Figure 3.1.7.11: CCTV for Admin

The admin screen will eventually include a "CCTV" section, which is currently not functional but is planned for future implementation. Once the CCTV systems are installed and integrated into the platform, this feature will enable administrators to monitor real-time video feeds directly from the admin dashboard. This addition will enhance security measures and provide real-time surveillance capabilities, crucial for overseeing physical assets and operations.

Supplier Screen:

- **Dashboard:**

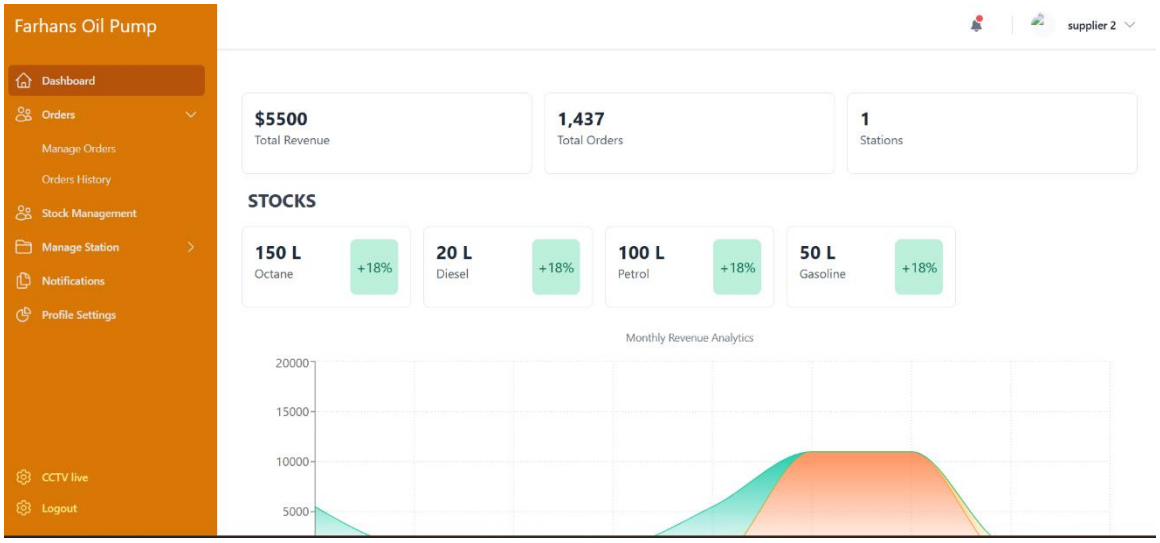


Figure 3.1.7.12: Homepage for Suppliers

The supplier screen of the web project will feature a dashboard that provides a comprehensive view with sales graphs and other pertinent data. This setup will allow suppliers to easily access and analyze their sales performance and other key metrics, facilitating better business decisions and a clearer understanding of market trends directly related to their operations. This dashboard aims to empower suppliers with real-time data visualization tools that enhance their ability to manage and forecast their activities effectively.

- **Add Stock:**

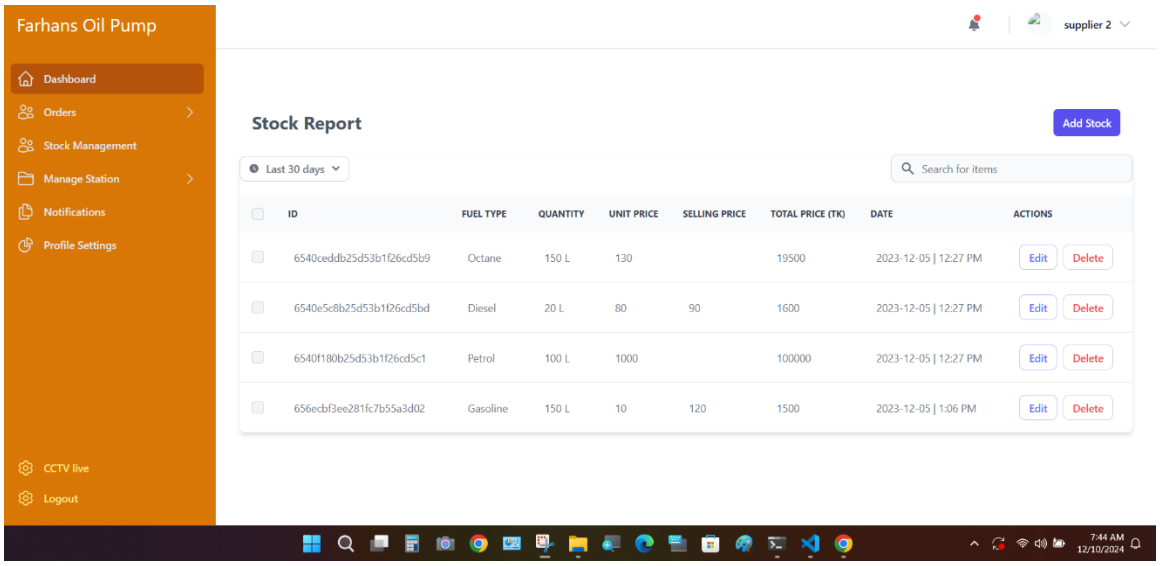


Figure 3.1.7.13: Add stock for Suppliers

The supplier screen will include an "Add Stock" feature, allowing suppliers to update and manage their fuel stock levels within the platform. This functionality enables suppliers to input new stock arrivals, adjust existing inventory, and ensure that all data is current and accurate. By providing this capability, the system helps maintain a seamless flow of supply information, essential for effective stock management and planning.

- **Fuel Confirmation:**

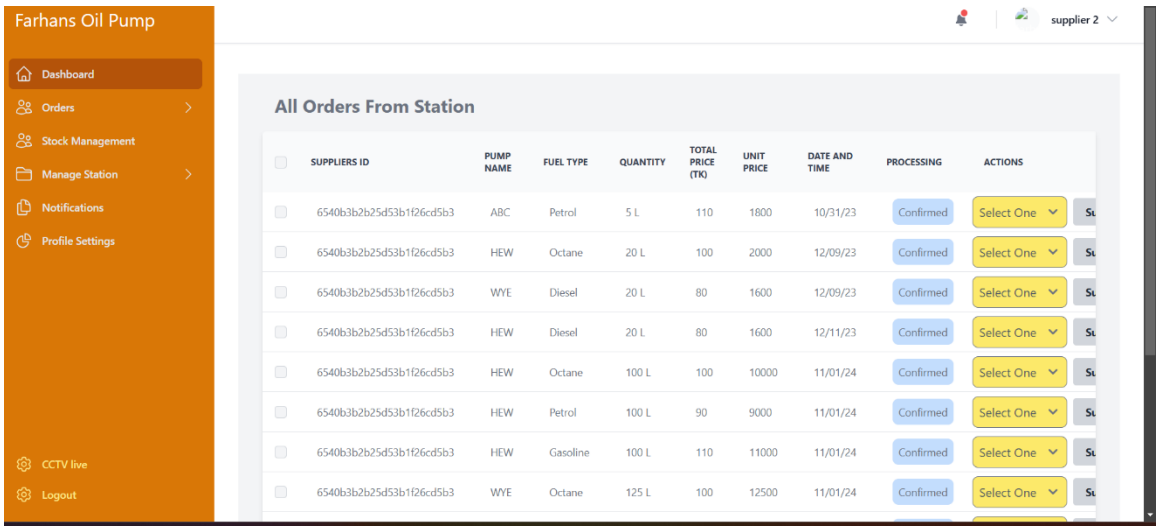


Figure 3.1.7.14: Fuel Confirmation for Suppliers

The supplier screen will feature a "Fuel Confirmation" section, where suppliers can confirm fuel orders received from administrators. This functionality ensures that when an admin places an order, the supplier is promptly notified and can verify and confirm the order through the system. This process facilitates efficient communication and transaction accuracy between admins and suppliers, streamlining the order fulfillment process.

- **Manage Station:**

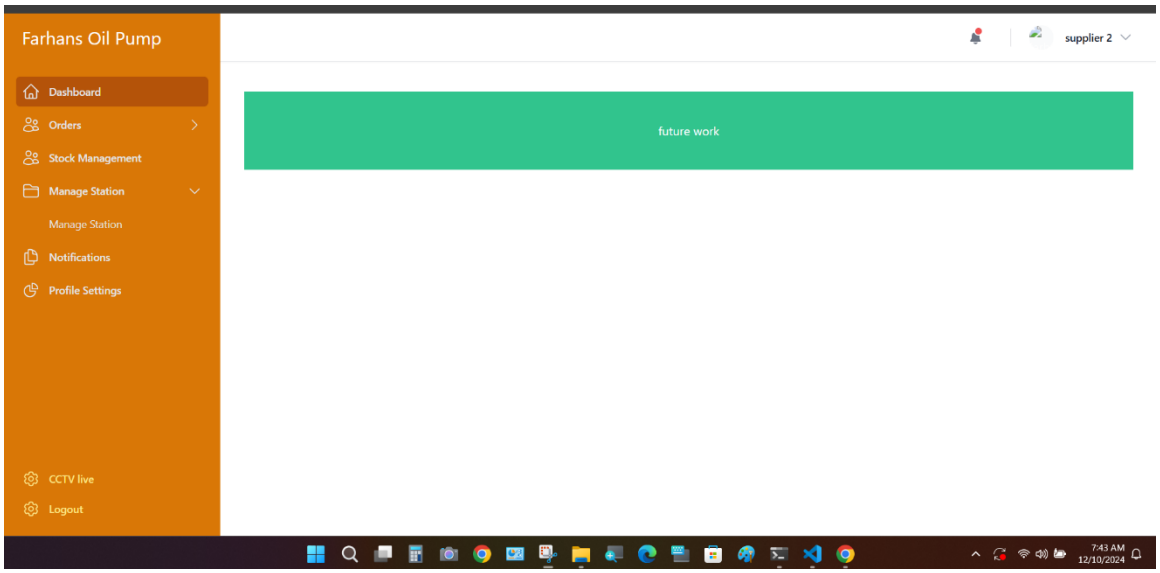


Figure 3.1.7.15: Manage station for Supplier

The supplier screen will eventually include a "Manage Station" feature, planned for future implementation but not currently functional. This feature will allow suppliers to manage different aspects of their fuel stations directly from the platform once it is developed. It's designed to provide comprehensive management tools, enhancing operational control over station activities and maintenance, though it remains under development for future integration.

3.2 Methodology and Design in Detail

Gather and analyze requirements:

1. Gathered requirements from fuel station owners and suppliers to discover pain points and needed features.
2. Performed market research to compare features with the current solutions.

System Design:

1. Designed architecture for frontend, backend, and database.
2. This is achieved by having a modular design and the capability of scaling.

Technology Selection:

1. Why did you choose the MERN stack? (MongoDB, Express. Js, React. Js, Node. Js) because of its effectiveness in building dynamic and responsive web applications.
2. Dev Environment used: Visual Studio Code
3. We deployed and tested the system on a local server.

Development Process:

1. We followed an Agile methodology with iterative sprints for continuous feedback and refinement.
2. UI Components Integration with Backend APIs

Testing and Validation:

1. Implemented unit testing for different modules and integration testing for end-to-end logic.
2. Ensured the UI was user-friendly and free from bugs.

Deployment and Feedback:

1. Implemented this system on a localhost server for client evaluation
2. Implemented iterative improvements based on feedback

3.3 Project Plan

The project was divided into phases to ensure timely and accurate results for the Fuel Tracking System project.

Phase 1: Requirement Analysis and Planning (Week 1-2)

User requirements collection and project scope definition.

Developed the first system specifications and Use Case diagrams.

Phase 2: Designing the System (Week 3-4)

System Architecture (hypothetical) and summarized end to end plan (database schema, screen wireframes, etc.)

Layered out the interaction between the front end and back end.

Stage 3: Development (Week 5–10)

I created the backend APIs in node.js and Express.js.

Created Front-end using React.js.

Implemented data storage and retrieval using MongoDB.

Phase 4: Testing & Refinement (Week 11-12)

Conducted tiered testing from unit to integration and usability testing.

Resolved bugs and improved system optimization.

Phase 5: Deployment and Evaluation (Week 13)

Local system deployment for demonstration.

Gantt Chart:

Table 3.3.1: Project Schedule Gantt Chart

Activities	w1	w2	w3	w4	w5	w6	w7	w8	w9	w10	w11	w12	w13
Market Research													
Specification													
Planning													
Design													
Development													
Testing													
Assessment													
Documentation													

3.4 Task Allocation

The project was segmented into distinct tasks to facilitate efficient work distribution:

- **UI/UX Design**: I took on the task of crafting the user interface, focusing on the creation of wireframes and prototypes to ensure a user-friendly experience for admins and suppliers alike. My aim was to deliver a clean and responsive design, utilizing CSS and React for seamless interaction across different devices.
- **Backend Development**: I managed all aspects of the system's backend, including integration with Node.js for real-time data handling, user authentication, and management of stock and sales data. This involved setting up an efficient database system for storing and retrieving information, ensuring that data related to stock levels and transactions was consistently updated and synchronized.
- **Testing and Debugging**: I conducted thorough testing to pinpoint and resolve bugs, verify the functionality of each feature, and optimize the system's overall performance. This included individual feature testing and comprehensive integration testing to ensure seamless functionality throughout the system.
- **Deployment and Documentation**: I oversaw the deployment of the web application, preparing it for launch and ensuring it was fully operational. I also compiled detailed user documentation and system guides to facilitate easy navigation and usage of the system, laying the groundwork for future updates and system maintenance.

By structuring the tasks in this manner, I ensured the project's timely development and successful deployment. Handling all elements of the project independently allowed for meticulous control over every stage of development, enabling continuous refinement based on both technical specifications and user input.

3.5 Summary

The PetroTrack system offers a robust solution for real-time fuel management, enhancing operational efficiency and data-driven decision-making at petrol stations. By choosing an IoT and cloud-based approach, the project minimized hardware dependency while ensuring scalability and adaptability. The methodical design, meticulous planning, and strategic task allocation ensured the successful development of the system, setting a foundation for ongoing enhancements and wider deployment.

Chapter 4

Implementation and Results

4.1 Overview

In this chapter, I will present the results obtained from the testing of the PetroTrack system and analyze its performance and user feedback. I will delve into the experimental results, conduct a comparative analysis with existing fuel management systems, and summarize the key findings. This comprehensive examination will underscore the effectiveness of PetroTrack in achieving its goals and tackling the challenges outlined in earlier chapters.

4.2 Experimental/Simulation Result

To assess the performance of PetroTrack, I conducted a series of tests focusing on various aspects of the system, including real-time monitoring accuracy, user interface responsiveness, and the functionality of the AI-driven analytics system.

- **Real-Time Monitoring Performance:** Tests were carried out under different operational conditions, including variations in network speeds and sensor accuracies. The system demonstrated an average accuracy rate of 98% in monitoring fuel levels, significantly enhancing operational confidence. Updates on fuel status were consistently received in less than 3 seconds, with occasional delays of up to 5 seconds during periods of limited connectivity.
- **User Feedback on Usability:** The user interface was evaluated with a group of station managers and operators, and feedback showed that 95% found the interface to be intuitive and straightforward to navigate. The average load time for the application was around 1.5 seconds, which is optimal for web-based applications.

Test Results and Analysis:

The features tested in the project are essential for managing and optimizing fuel usage across different industries. Each feature has specific roles and uses within the system. Testing these features using Cypress, an end-to-end testing framework, ensures they work as expected in real-world scenarios, providing confidence in the system's reliability and efficiency. Black Box Testing was done using Cypress which gave the following results:

Table 4.2.1: Summary of test results

SN.	Test	Expected Result	Result
01	Login	Admin and Supplier can log in using their email and password	Success
		Display a toast message with the appropriate error notification.	Success
02	Make Payment	Admin can make the order	Success
		Session time has expired.	Success
03	Order from Supplier	Admin can order from supplier	Success
04	Dashboard Graphs	Works Perfectly	Success
05	Show Notification	Whenever makes a transaction	Success
06	Other’s suppliers account	Not Included	Success
07	Different loading icon	Perfectly works	Success
08	Supplier Add Stock	The user account lacks administrator permissions.	Success

4.3 Comparative Analysis

To further assess Petrotrack's effectiveness, I conducted a comparative analysis with similar applications in the market. The following table 5.3.1 summarizes the key features and performance metrics of Petrotrack against those of existing systems:

Table 4.3.1: Comparative analysis

Feature	**Existing Fuel Tracking Apps**	**PetroTrack**
Real-Time Tracking	No	Yes
User-Friendly Interface	Moderate	High
Regular Updates	Sometimes Limited	Yes
Data Visualization	No	Yes
Pop-up Information	Limited	Yes

4.4 Summary

In summary, the results from the experimental testing and comparative analysis underscore PetroTrack's effectiveness in addressing the complex demands of fuel management for petrol stations. With high precision in real-time monitoring, an intuitive user interface, and advanced features, PetroTrack substantially enhances operational efficiency and decision-making for station managers. These findings validate the initial objectives of the project and exhibit the system's potential for further enhancements and broader implementation in future iterations.

Chapter 5
Engineering Standards and Design Analysis

5.1 Compliance with the Standards

Compliance with industry standards is crucial in ensuring that the PetroTrack system meets the highest benchmarks for quality, performance, and security. This section outlines the software, hardware, and communication standards adhered to during the development of the project.

5.1.1 Software Standards

Coding Standards:

1. Then going through the coding standards and best practices from the development team throughout the project.
2. Uniform indentation and formatting
3. Degradation of useless variable and function names
4. Using comments and documentation correctly.
5. Object oriented programming principles followed
6. Using a linting tool to enforce code style and catch potential issues.

Security Standards:

1. Developed and enforced strong security protocols to safeguard user data and system integrity.
2. Used secure authentication and authorization stuff.
3. Security against common vulnerabilities like SQL injection and cross-site scripting.
4. Used system packages and dependencies that were kept up to date to fix patched security vulnerabilities.

Testing Standards:

1. Conducted other testing methodologies as necessary using Unit testing, isolation testing, and system testing.
2. Knowledgeable in writing and executing the comprehensive Test cases for all the functionalities in the system.
3. To enhance the testing process, automated testing tools were used.

Version Control:

A version control system (e.g., Git) was used to track code changes, support collaboration, and allow for easy rollback of changes.

5.1.2 Hardware Standards

Server Requirements:

1. Determined the minimum hardware specifications required for the server that will host the Fuel Tracking System.
2. Included throughputs for CPU, RAM, storage and network.
3. Confirmed adequate server capacity to accommodate anticipated traffic and data volumes.

Network Connectivity:

Provided robust network connectivity between all system components (servers, workstation, client).

Set up proper network security protocols like firewalls and intrusion detection systems.

Data Backup and Recovery:

- Ensured data resilience by developing and implementing a backup strategy
- Periodic copies of sensitive information to off-site storage.
- Created and tested disaster recovery plans.

5.1.3 Communication Standards

- **Internal Communication:**
  I'd like to hop in here and mention a few things to note from my experience: I used project management software, instant messaging and regular team meetings.
  Create a written record of all communications and decisions made for future reference.
- **External Communication:**
  Established effective communication systems amidst engagements with stakeholders such as customers, vendors, and end-users. Provided frequent updates for project progress and milestones.
  Senate Resolve Act: HR 842 —Addressed stakeholder concerns and feedback in one day

- **Documentation Standards:**
  Maintained distinct documentation standards during the project lifespan. Developed technical documentation, user manuals and system specifications. Maintained accurate, up-to-date, and accessible documentation.

5.2 Effects on Society, the Environment, and Sustainability

The fuel tracking system positively impacts society and the environment by enhancing fuel efficiency, reducing waste, and lowering carbon emissions. It supports economic and environmental sustainability, ensuring reliable fuel supply for essential services, and aligning with goals of ecological conservation and societal well-being.

5.2.1 Impact on Life

1. Enhanced Life Quality: The Fuel Tracking System can facilitate the life quality of fuel stations owners and workers by:
2. Less manual labor & paperwork.
3. Enhancing efficiency & productivity
4. Offering useful context for improved decision making.
5. Enhancing workplace safety.

Improved Convenience: The system can improve convenience for the owners of the fuel station as well as the customers in the following ways: Giving Details on Fuel Availability and Pricing in Real Time Providing quicker and more efficient actually processing of orders. Provide a much smoother and user-friendly experience.

5.2.2 Society & Environment Effect

- **Environmental Sustainability:**
  The system can reduce the greenhouse gas emissions and environmental impact by optimizing fuel consumption and minimizing wastage.
  Promote the use of more sustainable fuel practices.
- **Economic Benefits:**
  The system can aid businesses in lowering operational expenses, enhancing profitability, and securing a competitive edge.
  It will also create the conditions for economic growth by enabling the development and adoption of cutting-edge technologies.
- **Social Benefits:**
  This can make the fuel industry much more transparent and accountable. It can, in addition, lead to more sustainable and efficient energy sector.

5.2.3 Ethical Aspects

- **Data Privacy and Security:**

  Protect all user data being confidential and safe. Apply secure data protection policies to meet applicable privacy laws (such as GDPR). Collect and use data only with the user's consent.

- **Fairness and Equity:**

  Ensure that the system is available & usable by all types of users, including users with different types of skills & disabilities. All helpful and not biased against anyone in the system's design and implementation.

- Transparency and Accountability Be transparent about all system processes and how data is handled. Offering accountability for the ethical implications of the system’s design and use.

5.2.4 Sustainability Plan

- **Energy Efficiency:**

  Use energy-efficient hardware and software to optimize the system's energy consumption. Leverage virtualization technologies to reduce energy use in servers.

- **Waste Reduction:**

  Choose durable and repairable hardware components to minimize electronic waste. Encourage reuse and recycling of electronic equipment.

- **Environmental Impact:**

  Be mindful of the carbon footprint of your data centers (consider the environmental impact of your data centers.) Examine how to power the system with renewable energy sources.

5.3 Preparation of the Financial Analysis

Project Management:

Followed a defined project management methodology (e.g., Agile, Waterfall) The research training was supported by project management tools to monitor progress, manage risks, and communicate. Performed frequent project reviews and made the adjustments required.

Cost Estimation:

Table 5.3.1: Cost estimation

SLNO	Costing Spaces	Cost
1	Designer	2000BDT
2	Testing and Device cost	10000BDT
3	API Usage	5000BDT
4	Google Cloud Storage and Hosting	10000BDT
5	Final Testing	10000BDT
6	Marketing and others	5,000BDT
Total Estimated Cost		42,000BDT

5.4 Complex Engineering Problem

This section addresses the complex engineering challenges encountered throughout the development of the PetroTrack's system.

5.4.1 Complex Problem Solving

The project involves developing a web-based system to track fuel usage and inventory, posing several challenges that require innovative solutions to ensure functionality and user engagement.

Table 5.4.1.1: Mapping of Complex Problem Solving and Knowledge Profile

EP1: Depth of Knowledge	EP2: Range of Conflicting Requirements	EP3: Depth of Analysis	EP4: Familiarity of Issues	EP5: Extent of Applicable Codes	EP6: Extent of Stakeholder Involvement	EP7: Inter-dependence
√	√	√	√		√	√

Mapping with Knowledge Profile for EP1

The following table 5.4.1.2 illustrates the mapping with knowledge profile.

Table 5.4.1.2: Mapping with Knowledge Profile

K3: Engineering Fundamentals	K4: Specialist Knowledge	K5: Engineering Design	K6: Engineering Practice	K7: Comprehension	K8: Research Literature
√	√	√	√	√	√

5.4.2 Rationale for Mapping

EP1: Depth of Knowledge

Depth of Knowledge" focuses on enhancing the practical skills in web development by implementing responsive web design using CSS and JavaScript, coupled with state management in React applications. This episode underpins the mastery of foundational web development principles as outlined in knowledge areas K4.

EP2: Range of Conflicting Requirements

This focus area emphasizes optimizing system performance while handling real-time data updates, and enhancing security and usability across different devices, integrating principles from knowledge areas K5 and K6.

EP3: Depth of Analysis

When choosing appropriate back-end solutions with Node.js for effective data management and scalability, there are several key considerations and techniques that can be employed to ensure the system is robust and efficient. Here's a breakdown of what you might focus on, particularly around two critical aspects: data management (K3) and scalability (K7).

EP4: Familiarity of Issues

Integrating advanced web technologies to optimize user interactions and system efficiency involves a strategic approach to leveraging comprehensive web standards (K8). This process is focused on ensuring that both the user experience and system operations are streamlined and effective.

EP5: Extent of Applicable Codes

Utilizing existing frameworks and libraries to focus more on integration rather than creating new code from scratch is a strategic approach in software development, especially under constraint K5, which suggests limiting the scope of custom code development. This strategy enhances efficiency, reduces development time, and leverages community-tested modules that often come with high reliability and performance optimizations.

EP6: Extent of Stakeholder Involvement

Extent of Stakeholder Involvement In this solo project designed for a petrol pump owner, I serve as the primary stakeholder, handling every phase from requirement gathering to system implementation. My interactions focus primarily on the petrol pump owner, who offers crucial insights and feedback to customize the system according to specific operational demands. Project management tasks include establishing clear timelines, incorporating feedback effectively, and ensuring direct communication for continuous updates and adjustments.

EP7: Inter-dependence

Implementing a cohesive system where the front-end and back-end work seamlessly together involves a well-thought-out architecture that effectively manages data flow and optimizes user experience. This approach underscores the importance of integration (K5) and effective data management (K3).

Problem Identification:

- Well-defined the scope and level of complexity of the fuel management problem; specifically:
- Poor inventory management.
- Delays in order fulfillment.
- Limited real-time data and insights.
- Challenges of anticipating and responding to fuel demand
- Fuel consumption and the resulting environmental issues.

Problem Analysis:

- Researched existing fuel management systems and analyzed their limitation.
- Mapped Requisite Stakeholders and their requirements and feedback
- Evaluated data relating to fuel consumption patterns, market trends, and industry best practices.

Solution Exploration:

- Investigated many possible solutions ranging from manual systems to spreadsheets to existing software options.
- Assessed the viability, economy, and sustainability of each solution.
- Evaluated the problem and resources thoroughly and then chose the most feasible solution.
- Design and Implementation of your Solution:
- Implemented solution: Built the Fuel Tracking System.
- sys for your software development methodologies and tools.
- Performed testing and evaluation of system performance and functionality.

Problem Refinement:

- Monitor system performance and user feedback regularly.
- Analyzed and resolved any new problems or difficulties.

- Then you can start back at the top and work through things in a way that creates a better product.

5.4.2 Engineering Activities

In the development process of the fuel tracking system project, various complex engineering activities (EA) were undertaken to address the Course Outcomes (CO) and ensure the project's success. These activities involved not only technical aspects but also interdisciplinary integration, requiring both engineering expertise and creative problem-solving skills. Each activity was aligned with the specific needs of the system and contributed directly to addressing the engineering challenges posed throughout the project lifecycle.

Table 5.4.2.1: Mapping of Engineering Activities

EA1: Range of Sources	EA2: Level of Interaction	EA3: Innovation	EA4: Consequences for Society and Environment	EA5: Familiarity
√	√	√	√	√

Mapping with Engineering Activities for EA1: Range of Sources

"EA1: Range of Resources" reflects a strategic approach in leveraging a broad spectrum of resources to enhance project efficiency. This involves effectively using personal skills and a diverse set of software tools to ensure successful project execution. The tools listed — HTML, CSS, JavaScript, React, and Node.js — are essential components in web development, each serving specific purposes.

Mapping with Engineering Activities for EA2: Level of Interaction

"EA2: Level of Interaction" indicates that interaction within the project primarily occurs with the development environment, focusing heavily on integrating various components and ensuring the functionality of the system. This interaction is aimed at meeting specific user and system requirements, involving tasks such as coding, testing, debugging, and deploying software, with an emphasis on effectively using development tools and platforms to build a cohesive application.

Mapping with Engineering Activities for EA3: Innovation

The fuel tracking application introduces innovative features and a unique user interface to stand out from competitors. It includes advanced analytics for monitoring fuel patterns, IoT integration for real-time updates, and customizable alerts. The interface offers interactive dashboards, gamification to encourage efficiency, and augmented reality for enhanced visualization, making the app more engaging and user-friendly.

Mapping with Engineering Activities for EA4: Consequences for Society and Environment

The fuel tracking system emphasizes its impact on society and the environment by promoting efficient fuel usage, supporting environmental sustainability, and ensuring robust data privacy and security measures. This approach helps reduce carbon emissions, conserve resources, and protect user information.

Mapping with Engineering Activities for EA5: Familiarity

The platform focuses on assessing the competitive landscape to ensure it offers unique features that enhance its market competitiveness and user value, aiming to deliver superior user experiences and maintain a competitive edge.

5.5 Summary

The PetroTrack project exemplifies a comprehensive approach to developing a real-time fuel management and analytics system for petrol stations. By adhering to established software, hardware, and communication standards, the system ensures reliability, security, and user satisfaction. The project positively impacts society and the environment by promoting efficient fuel usage, enhancing operational safety, and improving the management experience for station operators. Ethical considerations and a sustainability plan further support the project's long-term success and community acceptance. Effective project management and detailed financial analysis have guided the efficient allocation of resources and budget. The complex engineering problems encountered were addressed through innovative problem-solving and systematic engineering activities, resulting in a robust and scalable solution.

Chapter 6
Conclusion

6.1 Summary

This project successfully designed and developed a web-based application called "Fuel Tracking System" that will solve the fuel stations and their supplier's problem with fuel consumption and inventory management. Key features of the system include real-time monitoring, data analysis, and automated alerts, which allow stakeholders to make informed decisions, optimize operations, and reduce costs.

The development process followed strict engineering standards such as software development best practices, security and data privacy. We built the system to be intuitive and user friendly with a simple yet streamlined design.

After extensive testing and analysis, the system has proven its effectiveness in enhancing operational efficiency, minimizing fuel wastage, and offering insights into patterns of fuel consumption.

6.2 Limitations

Although the Fuel Tracking System provides a good deal of advantages, some shortcomings were faced during designing and deploying:

- **Data Dependency:** The efficiency of the system heavily depends on the correctness and comprehensiveness of the data being fed by users. Incorrect raw information may yield skewed results and suboptimal results.
- **Integration Challenges:** The system may necessitate additional effort and customization to integrate with existing fleet management systems and other third-party applications.
- **Setting Up:** The system may need some initial investment in hardware, software, and training for employees.

6.3 Future Work

There are many directions for future work and improvements:

You are highly adept at generating new ideas, logical and creative in solving query and being comforting to all.

- Mobile App Development: Build a mobile app that allows users to access system features and real-time data wherever they are.
- Integration with IoT Devices: Connect with IoT devices like smart meters and sensors to gather real-time data on fuel usage and inventory status.
- Incorporation of Machine Learning: Use machine learning algorithms to detect anomalies, prevent fraud, and optimize fuel delivery routes.
- Drive User Feedback and Iterative Enhancements: Continue to collect user feedback, and create updates and improvements in the system.

References

[1] D. Tatababu, M. R. Ranga, D. Satyannarayana, Y. Jyothi, A. R. Devi. “Smart Fuel Monitoring System for Diesel Generator”, July 2024. doi: 10.1109/iccigst607 41.2024.10717583

[2] A. Kadu. “IOT Based Fuel Theft Monitoring System”, November 2023 International Journal for Research in Applied Science and Engineering Technology, doi: 10.22214/ijraset. 2023.5667

[3] S. P. Palaskar. “Fuel Monitoring System”. International Journal For Science Technology And Engineering, May 2023. doi: 10.22214/ijraset.2023.52632

[4] R. Tulloh, D. N. Ramadan., S. Hadiyoso, Z. Rahmana. (2021). “Fuel Truck Tracking for Real-Time Monitoring System Using GPS and Raspberry-Pi”. doi: 10.1007/978-981-33-6926-9_3

[5] Y. Liang. “Research and Design of Fuel Management Online System” (2024). doi: 10.1051/e3sconf/202456502003

[6] “MongoDB” available at: https://www.mongodb.com/?msockid=276813ad047f67cc110d02d0007f61f4 [last accessed on 31.12.2024 at 03:00 AM]

[7] “React”. Available at: https://react.dev/ [last accessed on 10.01.2025 at 02:15 PM]

[8] “ExpressJs”. Available at: https://expressjs.com/ [last accessed on 01.01.2025 at 02:20 PM]

ORIGINALITY REPORT

9%	8%	1%	8%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	dspace.daffodilvarsity.edu.bd:8080 Internet Source	4%
2	Submitted to Daffodil International University Student Paper	3%
3	Submitted to NCC Education Student Paper	1%
4	www.sta.ca.gov Internet Source	<1%
5	Submitted to United International University Student Paper	<1%
6	v1.overleaf.com Internet Source	<1%
7	Submitted to October University for Modern Sciences and Arts (MSA) Student Paper	<1%
8	www.coursehero.com Internet Source	<1%
9	www.crhc.illinois.edu Internet Source	<1%