PetroPulse

Petroleum Management System

22i-0907, 22i-1241, 22i-1000
FAST NUCES ASSIGNMENT 1

PETRO PULSE

A Petroleum Management System

Problem Statement

The problem of	inefficient fuel management and service station operations
affects	fuel station owners, employees, and customers
the impact of which is	delays in fuel requests, inventory shortages, mismanaged service shifts, and a lack of real-time analytics
a successful solution would be	a comprehensive Petroleum Management System that allows efficient fuel request handling, real-time inventory tracking, optimized shift management, seamless payment processing, customer profile management, and analytical reporting. The system should enhance operational efficiency, minimize fuel shortages, and provide users with a smooth and transparent experience.

Envisioned Features

1. Fuel Request Management
2. Fuel Order Processing
3. Real-time Inventory Tracking
4. Automated Stock Refill Requests
5. Shift Management
6. Employee Check-in System

7. Payment Processing
8. Transaction History
9. Loyalty Points System
10. Membership Management
11. Analytics & Reporting Dashboard
12. Fuel Delivery Tracking
13. Real-time Alerts & Notifications
14. Customer Feedback System
15. Employee Performance Reports
16. Vehicle Registration for Fuel Requests
17. Role-based Access Control
18. Dynamic Fuel Pricing System
19. Bulk Fuel Purchase Discounts
20. Fuel Station Maintenance Management
21. Supplier & Vendor Management
22. Emergency Fuel Request Handling
23. Multi-station Management for Chain Owners
24. Tax & Compliance Reporting
25. Customer Support & Chat Assistance

USER STORIES

---

1. *Fuel Request Management*

User Story:

- As a **customer**,
- I want to **request fuel online**,
- So that **I can refuel conveniently without waiting in long queues**.

Acceptance Criteria:

- And I know I am done when **I receive a confirmation of my request**.

Type: Manage Data

---

2. *Fuel Order Processing*

User Story:

- As a **station manager**,
- I want to **approve or reject fuel requests**,
- So that **fuel is allocated efficiently**.

Acceptance Criteria:

- And I know I am done when **requests are processed and the customer is notified**.

Type: Workflow

---

3. *Real-time Inventory Tracking*

User Story:

- As an **inventory manager**,
- I want to **monitor fuel levels in real-time**,

- So that **I can reorder stock before running out**.

Acceptance Criteria:

- And I know I am done when **inventory alerts are generated when fuel is low**.

Type: Manage Data

---

4. *Automated Stock Refill Requests*

User Story:

- As an **inventory manager**,
- I want to **set automatic reorder thresholds**,
- So that **fuel is always available**.

Acceptance Criteria:

- And I know I am done when **the system generates orders based on low stock**.

Type: Workflow

---

5. *Shift Management*

User Story:

- As a **station manager**,
- I want to **schedule employee shifts**,
- So that **I can ensure efficient staffing**.

Acceptance Criteria:

- And I know I am done when **shifts are assigned, and employees receive notifications**.

Type: Workflow

---

6. *Employee Check-in System*

User Story:

- As an **employee**,
- I want to **log my shift check-in and check-out**,
- So that **my working hours are recorded accurately**.

Acceptance Criteria:

- And I know I am done when **I see my shift summary**.

Type: Manage Data

---

7. *Payment Processing*

User Story:

- As a **customer**,
- I want to **pay using multiple methods (cash, card, digital wallets)**,
- So that **I have flexible payment options**.

Acceptance Criteria:

- And I know I am done when **the payment is processed and I receive a receipt**.

Type: Payment

---

8. *Transaction History*

User Story:

- As a **customer**,
- I want to **view my past purchases**,
- So that **I can track my expenses**.

Acceptance Criteria:

- And I know I am done when **I can see a list of all my previous transactions**.

Type: Report/View

---

9. *Loyalty Points System*

User Story:

- As a **customer**,
- I want to **earn loyalty points on purchases**,
- So that **I can redeem rewards**.

Acceptance Criteria:

- And I know I am done when **my loyalty balance updates after each purchase**.

Type: Manage Data

---

10. *Membership Management*

User Story:

- As a **new customer**,
- I want to **apply for a membership**,
- So that **I can access exclusive benefits**.

Acceptance Criteria:

- And I know I am done when **I receive membership approval and an ID**.

Type: Workflow

---

11. *Analytics & Reporting Dashboard*

User Story:

- As a **station owner**,
- I want to **view a dashboard with sales, inventory, and fuel usage data**,
- So that **I can make informed business decisions**.

Acceptance Criteria:

- And I know I am done when **I see updated analytics in real-time**.

Type: Report/View

---

12. *Fuel Delivery Tracking*

User Story:

- As a **station manager**,
- I want to **track incoming fuel deliveries**,
- So that **I can ensure timely restocking**.

Acceptance Criteria:

- And I know I am done when **I receive notifications on delivery status**.

Type: Workflow

---

13. *Real-time Alerts & Notifications*

User Story:

- As a **system administrator**,
- I want to **receive alerts for low fuel levels or system failures**,
- So that **I can act proactively**.

Acceptance Criteria:

- And I know I am done when **I receive automated alerts**.

Type: Manage Data

---

14. *Customer Feedback System*

User Story:

- As a **customer**,
- I want to **submit feedback about my experience**,
- So that **the service can be improved**.

Acceptance Criteria:

- And I know I am done when **I receive confirmation that my feedback is recorded**.

Type: Manage Data

---

15. *Employee Performance Reports*

User Story:

- As a **station manager**,
- I want to **generate reports on employee performance**,
- So that **I can evaluate productivity**.

Acceptance Criteria:

- And I know I am done when **I can view employee ratings and work logs**.

Type: Report/View

---

16. *Vehicle Registration for Fuel Requests*

User Story:

- As a **customer**,
- I want to **register my vehicle in the system**,
- So that **I can request fuel more efficiently**.

Acceptance Criteria:

- And I know I am done when **my vehicle information is saved**.

Type: Manage Data

---

17. *Role-based Access Control*

User Story:

- As a **system administrator**,

- I want to **manage user access levels**,
- So that **only authorized users can perform specific actions**.

Acceptance Criteria:

- And I know I am done when **users have restricted access based on their roles**.

Type: Workflow

---

18. *Dynamic Fuel Pricing System*

User Story:

- As a **station owner**,
- I want to **update fuel prices dynamically**,
- So that **prices reflect current market conditions**.

Acceptance Criteria:

- And I know I am done when **customers see updated prices**.

Type: Manage Data

---

19. *Bulk Fuel Purchase Discounts*

User Story:

- As a **corporate customer**,
- I want to **receive discounts on bulk fuel purchases**,
- So that **I can save on operational costs**.

Acceptance Criteria:

- And I know I am done when **discounts are applied during checkout**.

Type: Payment

---

20. *Fuel Station Maintenance Management*

User Story:

- As a **station manager**,
- I want to **schedule and track maintenance activities**,
- So that **fuel pumps and equipment remain in good working condition**.

Acceptance Criteria:

- And I know I am done when **maintenance tasks are assigned, completed, and logged**.

Type: Manage Data

---

21. *Supplier & Vendor Management*

User Story:

- As a **station owner**,
- I want to **manage fuel suppliers and vendors**,
- So that **I can ensure timely procurement of fuel and supplies**.

Acceptance Criteria:

- And I know I am done when **supplier details, contracts, and orders are stored and accessible**.

Type: Manage Data

---

22. *Emergency Fuel Request Handling*

User Story:

- As a **customer**,
- I want to **request emergency fuel assistance**,
- So that **I am not stranded due to unexpected fuel shortages**.

Acceptance Criteria:

- And I know I am done when **the system prioritizes emergency requests and sends alerts**.

Type: Workflow

---

23. *Multi-station Management for Chain Owners*

User Story:

- As a **chain owner**,
- I want to **oversee multiple fuel stations from a single dashboard**,
- So that **I can manage operations efficiently across locations**.

Acceptance Criteria:

- And I know I am done when **I can access reports, inventories, and sales data for all stations**.

Type: Report/View

---

24. *Tax & Compliance Reporting*

User Story:

- As a **station owner**,
- I want to **generate tax and regulatory compliance reports**,
- So that **I can ensure legal adherence and avoid penalties**.

Acceptance Criteria:

- And I know I am done when **I can download accurate tax and compliance documents**.

Type: Report/View

---

25. *Customer Support & Chat Assistance*

User Story:

- As a **customer**,
- I want to **contact customer support via chat**,
- So that **I can resolve issues quickly and efficiently**.

Acceptance Criteria:

- And I know I am done when **I receive real-time assistance through the chat feature**.

Type: Workflow

ROLES

1. Hasnain – Product Owner

Responsibilities:

- Define and prioritize user stories for the Petroleum Management System.
- Ensure that features such as fuel request handling, inventory tracking, and payment processing align with stakeholder needs.
- Communicate business requirements clearly to the development team.
- Accept or reject completed user stories based on their adherence to acceptance criteria.
- Continuously refine the backlog based on feedback from stakeholders and the development team.

2. Anas – Scrum Master

Responsibilities:

- Facilitate Agile ceremonies such as Sprint Planning, Daily Standups, Sprint Review, and Retrospective.
- Identify and remove blockers affecting the development of features like shift management, employee check-in, and real-time alerts.
- Ensure smooth collaboration between Hasnain (Product Owner) and Adam (Developer).
- Track sprint progress and ensure adherence to Agile principles.
- Guide the team in improving workflow efficiency and maintaining high productivity.

3. Adan – Developer/Scrum Team

Responsibilities:

- Develop and implement features such as fuel request processing, analytics dashboard, and customer profile management.
- Ensure code quality, security, and efficiency while integrating various modules.
- Conduct unit testing and debugging to ensure smooth system functionality.
- Collaborate with Hasnain (Product Owner) to clarify feature requirements and address feedback.
- Participate in sprint planning and daily standups to track development progress.

Team Agreement – Petroleum Management System

To ensure smooth collaboration and efficient progress in our project, we have established the following team agreement. This will serve as a guideline for how we communicate, work, and handle challenges as a team.

---

1. Methods of Communication

- **Primary Communication:** WhatsApp (for quick discussions, updates, and file sharing).
- **Secondary Communication:** Email (for formal communication, documentation sharing, and reporting).
- **Phone Calls:** Only for emergencies or when detailed discussions are needed.

2. Communication Response Times

- **WhatsApp:** Within **1-2 hours** during working hours.
- **Emails:** Within **24 hours** unless marked urgent.
- **Phone Calls:** Immediate response if available; if missed, return within **2 hours**.

---

3. Meeting Attendance

- **Sprint Planning, Sprint Review, and Retrospectives** are mandatory for all members.
- **Daily Standups:** Attendance is expected unless there's an emergency. Updates can be given asynchronously if someone is unavailable.
- **Ad-hoc Meetings:** Attendance is encouraged but can be flexible based on availability.

---

4. Running Meetings

- **When:** Sprint meetings will be held on Mondays; Retrospectives on Fridays.
- **Where:** Mostly online (Google Meet/Zoom); in-person meetings when necessary.
- **Minutes/Notes:** The Scrum Master (Anas) will document key decisions and action points.

---

5. Meeting Preparation

- Team members should review the agenda before meetings.
- Any assigned tasks must be completed or have a clear progress update ready.
- The Product Owner (Hasnain) should come prepared with backlog updates and priorities.

---

6. Version Control

- **Repository:** We will use GitHub for version control.
- **Commit Guidelines:**
    - Clear and descriptive commit messages (e.g., Added fuel request API, Fixed payment processing bug).
    - No committing untested or broken code.
    - Feature branches should be used before merging to the main branch.
- **What NOT to Commit:**
    - Personal credentials or sensitive data.
    - Debug logs, temporary files, or compiled binaries.

---

7. Division of Work

- Tasks will be assigned based on expertise and workload balance.
- The **Product Owner (Hasnain)** will define priorities, and the **Scrum Master (Anas)** will ensure fair task distribution.
- The **Developer (Adam)** will be responsible for implementation but will collaborate with the team on technical decisions.
- Stakeholders: Product Owner, Scrum Master, Developers, End-users (fuel station managers, suppliers).

---

8. Submitting Assignments

- **Deadline:** All tasks must be completed **at least 3 hours before the sprint ends** for review.
- **Who submits?** The Scrum Master (Anas) or a designated member will submit final deliverables.
- **Who reviews?** The Product Owner (Hasnain) will conduct a final review to ensure alignment with business needs.

---

9. Contingency Planning

- **If a team member drops out:** Responsibilities will be redistributed based on workload and expertise.
- **If a team member misses meetings consistently:** A one-on-one discussion will be held to address concerns. If the issue persists, the team will escalate it accordingly.
- **If a team member is academically dishonest:** The incident will be reported following the academic integrity policies.
