**Final Project  
Deliverable 1**

**Software Engineering**

Anas Bin Rashid i220907

Adan Malik i221000

Hasnain Akhtar i221241

Table of Contents

[**Title of the Project** 2](#_Toc191687450)

[**Problem Statement** 2](#_Toc191687451)

[**Features** 3](#_Toc191687452)

[**User Stories** 4](#_Toc191687453)

[**Team Member Roles** 9](#_Toc191687454)

[**Team Agreement** 10](#_Toc191687455)

[**GitHub Repository** 13](#_Toc191687456)

[**Trello Boards** 14](#_Toc191687457)

# **Title of the Project**

***PetroPulse: A Petrol Station Management Website***

# **Problem Statement**

|  |  |
| --- | --- |
| **The problem of** | inefficient and error-prone petrol station operations |
| **affects** | petrol station owners, employees, and customers, |
| **the impact of which is** | loss of revenue, inventory mismanagement, customer dissatisfaction, and security risks, mismanaged service shifts, |
| **a successful solution would be** | a centralized petrol station management system that automates fuel sales, inventory tracking, worker management, and customer loyalty programs while integrating with financial systems. PetroPulse will provide a comprehensive management solution by integrating sales tracking, automated reporting, and real-time fuel level monitoring. The system will ensure smooth operations by incorporating a secure payment mechanism, worker scheduling, and calculations for stock refills. Additionally, it will support customer engagement through a loyalty points program and provide remote access to station owners. |

# **Features**

1. **User Authentication** (Owner, Employee, Customer roles)
2. **Fuel Inventory Management** (Real-time tracking, automated stock refill requests)
3. **Sales Tracking & Reports**
4. **Payment Processing** (Cash, Card)
5. **Worker Management** (Shift management, attendance, check-in system)
6. **Customer Management** (Membership, support, chat assistance)
7. **Loyalty Points System**
8. **Fuel Dispensing Monitoring**
9. **Station Maintenance**
10. **Supplier & Vendor Management** (Order processing, tracking)
11. **Multi-Station Management** (For owners of multiple stations)
12. **Dynamic Fuel Pricing System** (Price adjustments based on market trends)
13. **Remote Access**
14. **Low Fuel Alerts**
15. **Customer Feedback System**
16. **Discounts & Promotions System** (Bulk fuel purchase discounts)
17. **Employee Performance Tracking & Reports**
18. **Financial Reporting Integration** (Tax & compliance reporting)
19. **Security & Fraud Detection**
20. **Analytics & Reporting Dashboard**
21. **Transaction History & Auditing**
22. **Exportable Reports** (CSV, PDF)
23. **Real-time Alerts & Notifications** (Email, SMS)
24. **Role-based Access Control**
25. **API for Third-Party Integration**
26. **User-friendly UI/UX**
27. **Government Fuel Regulations Follow-Up**
28. **Configurable Taxation Module**
29. **Support for Multiple Fuel Types**
30. **Vehicle Registration for Fuel Requests**

# **User Stories**

| **ID** | **User Story** | **Acceptance Criteria** | **Estimation** | **Priority** |
| --- | --- | --- | --- | --- |
| 1 | **User Registration** - As a new user, I want to register on the system so that I can access PetroPulse services. | **Given**: A new user navigates to the registration page. **When**: They provide valid details (name, email, phone, password). **Then**: The system successfully creates an account and sends a confirmation email. | 3 | High |
| 2 | **User Login** - As a registered user, I want to log into the system so that I can access my dashboard. | **Given**: A user is on the login page. **When**: They enter valid credentials. **Then**: The system grants access to their dashboard. | 2 | High |
| 3 | **Password Reset** - As a user, I want to reset my password so that I can regain access to my account if I forget it. | **Given**: A user is on the login page. **When**: They click on "Forgot Password" and enter their email. **Then**: The system sends a password reset link to their email. | 3 | Medium |
| 4 | **Add Fuel Station** - As an owner, I want to add new fuel stations so that I can manage multiple stations in the system. | **Given**: An owner is logged in. **When**: They enter station details and submit the form. **Then**: The system registers the new station and updates the list. | 3 | High |
| 5 | **View Fuel Stations** - As an owner, I want to view a list of my fuel stations so that I can manage their operations efficiently. | **Given**: The owner is on the dashboard. **When**: They navigate to the "Manage Stations" section. **Then**: The system displays a list of owned stations. | 2 | High |
| 6 | **Update Fuel Station Details** - As an owner, I want to update my fuel station details so that I can keep the information accurate. | **Given**: The owner is on the "Manage Stations" page. **When**: They update station details and save changes. **Then**: The system reflects the updated details. | 3 | Medium |
| 7 | **Remove Fuel Station** - As an owner, I want to remove a fuel station so that I can stop managing a closed or sold station. | **Given**: The owner is on the "Manage Stations" page. **When**: They click delete and confirm the action. **Then**: The system removes the station from the database. | 3 | Medium |
| 8 | **Register Workers** - As a station owner, I want to register workers so that they can manage station activities. | **Given**: The owner is on the "Manage Workers" page. **When**: They enter the worker's details and submit. **Then**: The system registers the worker. | 3 | High |
| 9 | **Assign Workers to Stations** - As an owner, I want to assign workers to specific stations so that they can operate in their designated locations. | **Given**: An owner is on the worker management page. **When**: They select a worker and assign a station. **Then**: The system updates the worker's assigned station. | 3 | Medium |
| 10 | **Worker Login** - As a worker, I want to log into the system so that I can manage station operations. | **Given**: A worker is on the login page. **When**: They enter valid credentials. **Then**: The system grants access to the worker dashboard. | 2 | High |
| 11 | **Record Fuel Sales** - As a worker, I want to record fuel sales so that the system tracks all transactions. | **Given**: A worker is logged in. **When**: They enter fuel sale details. **Then**: The system stores the transaction. | 3 | High |
| 12 | **Generate Sales Reports** - As an owner, I want to generate sales reports so that I can analyze revenue and performance. | **Given**: The owner is on the reports page. **When**: They select a date range and generate a report. **Then**: The system displays the sales data. | 3 | High |
| 13 | **Track Fuel Inventory** - As a station owner, I want to track fuel levels so that I can ensure continuous supply. | **Given**: The owner is on the dashboard. **When**: They check the station inventory. **Then**: The system displays real-time fuel levels. | 4 | High |
| 14 | **Alert for Low Fuel Levels** - As a station owner, I want to receive alerts for low fuel levels so that I can order refills in time. | **Given**: A station's fuel level drops below a threshold. **When**: The system detects the low level. **Then**: It sends an alert to the owner. | 3 | High |
| 15 | **Order Fuel Supply** - As an owner, I want to order fuel refills so that my stations never run out of stock. | **Given**: The owner is on the fuel supply page. **When**: They place an order for fuel. **Then**: The system processes and confirms the order. | 4 | High |
| 16 | **View Loyalty Points** - As a customer, I want to view my loyalty points so that I can track my rewards. | **Given**: A customer is logged in. **When**: They navigate to the "Loyalty Points" page. **Then**: The system displays their current balance. | 2 | Medium |
| 17 | **Redeem Loyalty Points** - As a customer, I want to redeem my loyalty points so that I can get fuel discounts. | **Given**: A customer has sufficient points. **When**: They redeem them at checkout. **Then**: The system applies the discount. | 3 | Medium |
| 18 | **Customer Fuel Purchase** - As a customer, I want to purchase fuel from the station so that I can refuel my vehicle. | **Given**: A customer is at a station. **When**: They choose the fuel type, quantity, and payment method. **Then**: The system processes the payment and updates inventory. | 4 | High |
| 19 | **Manage Pricing for Fuel Types** - As a station owner, I want to update fuel prices so that I can adjust them based on market conditions. | **Given**: The owner is on the pricing page. **When**: They update the price of a fuel type. **Then**: The system reflects the new pricing in real time. | 3 | High |
| 20 | **Manage Maintenance Requests** - As a station owner, I want to log maintenance requests so that I can ensure station upkeep. | **Given**: A station has an issue. **When**: The owner submits a maintenance request. **Then**: The system records and tracks the request. | 3 | Medium |
| 21 | **View Transaction History -** As a customer, I want to view my transaction history, so that I can track my fuel purchases. | **Given:** A customer is logged in, **When:** They navigate to the transactions section, **Then:** The system displays all past fuel purchases. | 2 | Medium |
| 22 | **Payment via Cash -** As a customer, I want to pay for fuel using cash, so that I can complete my purchase without a card. | **Given:** A customer selects fuel, **When:** They choose cash as the payment method, **Then:** The system registers the payment and updates records. | 3 | High |
| 23 | **Payment via Card -** As a customer, I want to pay for fuel using my debit/credit card, so that I can make transactions easily. | **Given:** A customer selects fuel, **When:** They choose card as the payment method, **Then:** The system processes the payment and updates records. | 3 | High |
| 24 | **Manage Supplier Information -** As an owner, I want to manage my fuel suppliers, So that I can track orders and deliveries. | **Given:** The owner is on the supplier management page, **When:** They add, edit, or remove a supplier, **Then:** The system updates the supplier database. | 3 | Medium |
| 25 | **Generate Maintenance Reports -** As an owner, I want to track station maintenance, so that I can schedule repairs and ensure smooth operations. | **Given:** The owner is on the reports page, **When:** They request a maintenance report, **Then:** The system provides a list of past and upcoming maintenance tasks. | 3 | Medium |
| 26 | **Employee Attendance Tracking -** As a station owner, I want to track employee attendance, so that I can monitor working hours. | **Given:** A worker logs into the system, **When:** They mark attendance for the shift, **Then:** The system records their working hours. | 3 | Medium |
| 27 | **Secure Access -** As a user, I want to have secure access to system configurations, so that I can manage system settings safely. | **Given:** The user is on the login page, **When:** They enter valid admin credentials, **Then:** The system grants access to the admin panel. | 3 | High |
| 28 | **View Worker Profiles -** As an owner, I want to view worker profiles, so I can manage their information and roles. | **Given:** The owner is on the "Manage Workers" page. **When:** They select a worker profile. **Then:** The system displays the worker's details. | 2 | Medium |

# **Team Member Roles**

***Product Owner*** *– Hasnain Akhtar*

**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.

***Scrum Master*** *– Anas Bin Rashid*

**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 Adan (Developer).
* Track sprint progress and ensure adherence to Agile principles.
* Guide the team in improving workflow efficiency and maintaining high productivity.

***Scrum Team*** *– Adan Malik, Anas Bin Rashid, Hasnain Akhtar*

**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**

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 (Adan) 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**

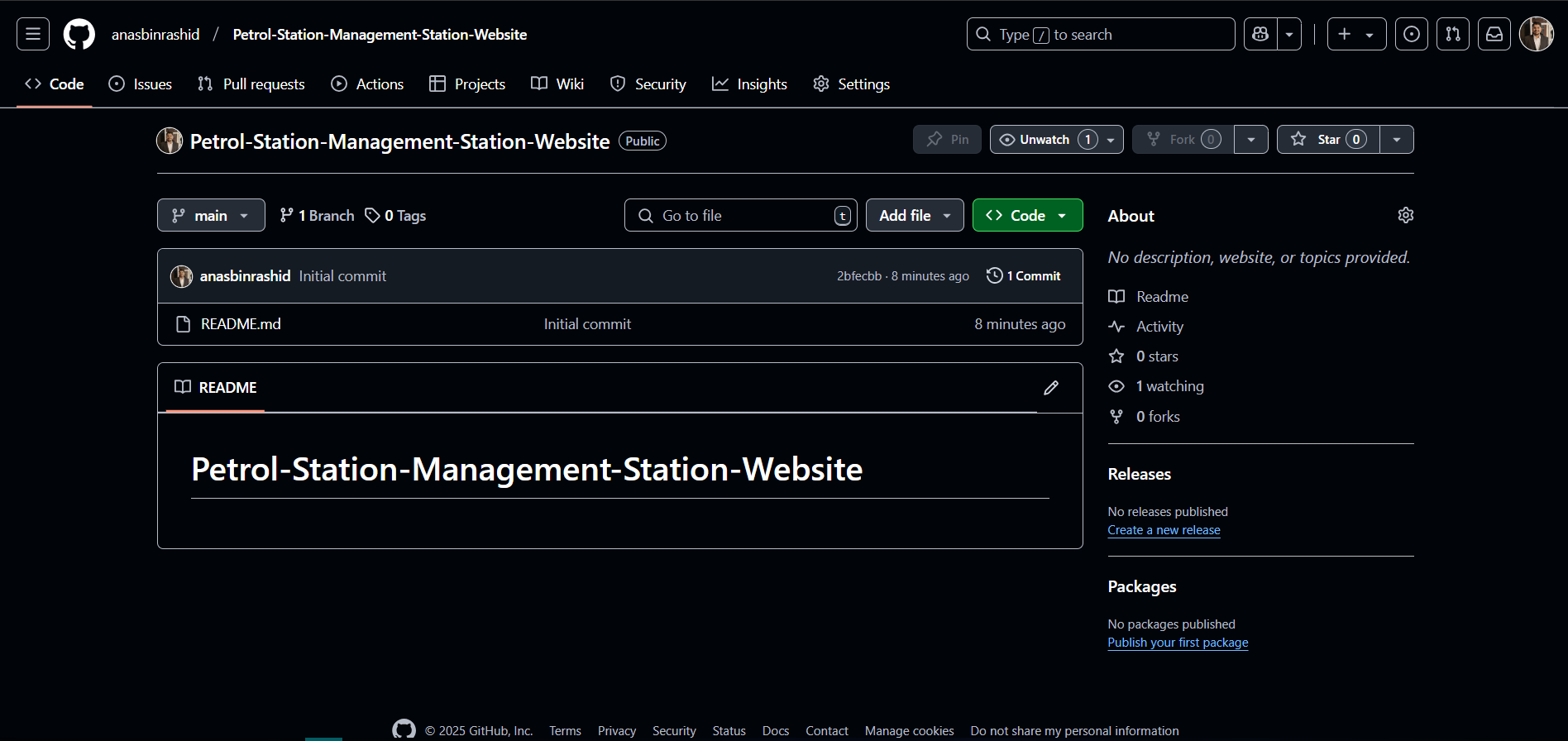
* Deadline: All tasks must be completed at least 3 hours before the sprint ends for review.
* Who submits? The Scrum Master (Anas) will submit final deliverables.
* Who reviews? The Product Owner (Hasnain) will conduct a final review to ensure alignment with business needs.

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 accordingly.
* If a team member is academically dishonest: The incident will be reported following the academic integrity policies.

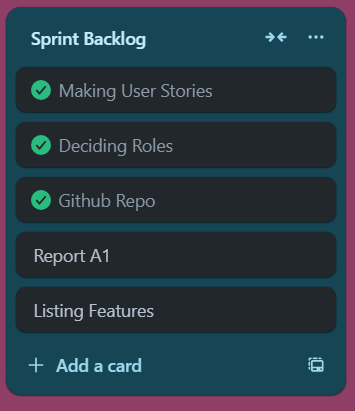
# **GitHub Repository**

**Repository Link:** [*https://github.com/anasbinrashid/Petrol-Station-Management-Station-Website*](https://github.com/anasbinrashid/Petrol-Station-Management-Station-Website)

****

# **Trello Boards**

* 1. **Sprint Backlog**

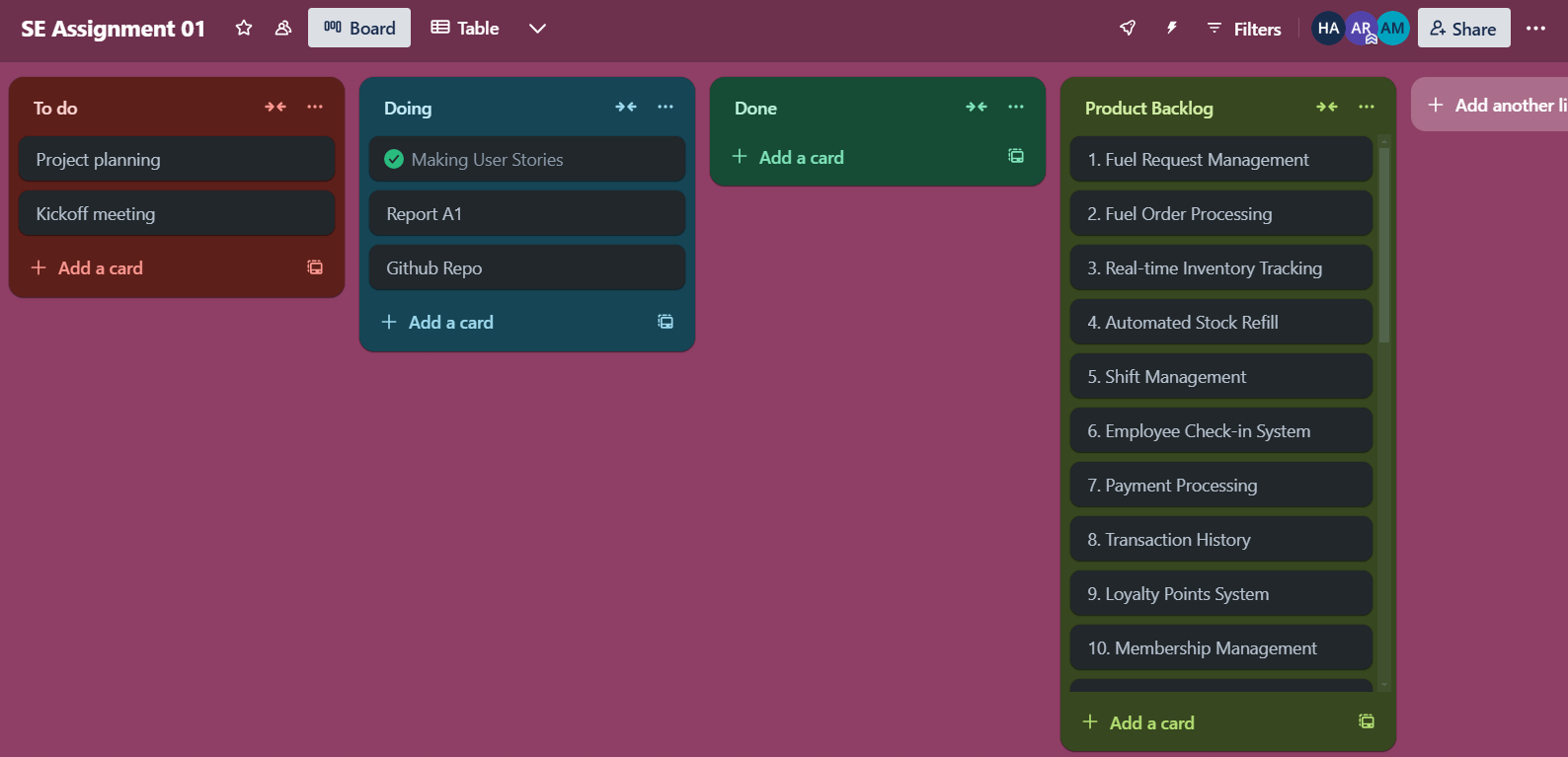
****

* 1. **Product Backlog**

**A screenshot of a phone

AI-generated content may be incorrect.**

* 1. **Complete Trello Board**

****