**Universidad de Antioquia.**

**Facultad de Ingeniería**

**DESAFÍO 2**

**Autores:**

Esteban Rodríguez Monsalve

Melany Sierra Torres

**Asignatura:**

Informática 2

**Docente:**

Aníbal Guerra

**Fecha:**

12 de Octubre 2024

**1. Análisis del Problema:**

El problema principal radica en modelar y gestionar una red compleja de estaciones de servicio, cada una con sus propias características y múltiples interacciones. Se requiere un sistema que permita:

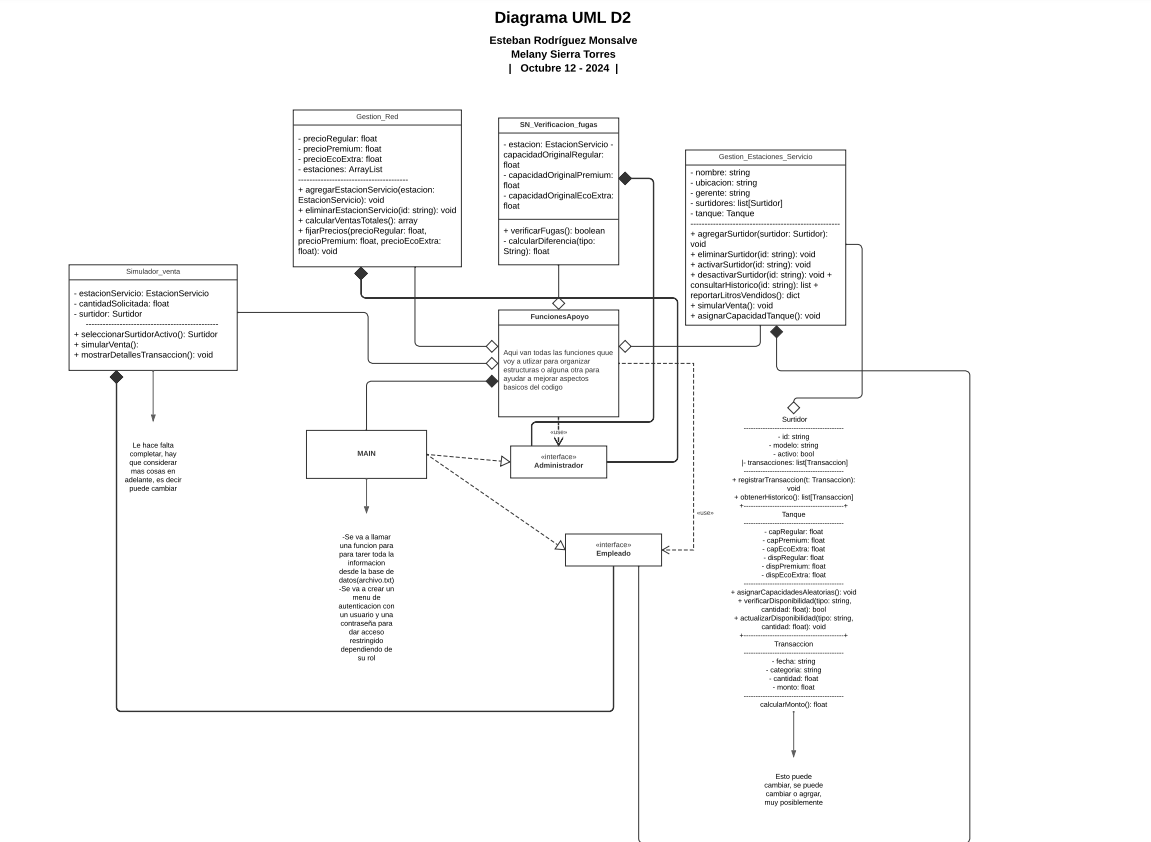
* **Gestión de estaciones:** Crear, modificar y eliminar estaciones, incluyendo sus atributos como ubicación, gerente y capacidad de tanques.
* **Gestión de surtidores:** Asociar surtidores a estaciones, registrar ventas y actualizar el inventario de combustible.
* **Simulación de ventas:** Realizar ventas de combustible considerando la disponibilidad en los tanques y los diferentes métodos de pago.
* **Verificación de fugas:** Implementar un mecanismo para detectar posibles fugas de combustible en los tanques.

### **2. Identificación de Requisitos**

* **Requisitos Funcionales**:
  + **Gestión de la Red**:
    - Agregar y eliminar estaciones de servicio.
    - Calcular el monto total de ventas en cada E/S discriminado por categoría.
    - Fijar precios del combustible.
  + **Gestión de Estaciones de Servicio**:
    - Agregar y eliminar surtidores.
    - Activar y desactivar surtidores.
    - Consultar el histórico de transacciones.
    - Reportar litros vendidos por categoría de combustible.
    - Simular ventas y gestionar la capacidad del tanque.
  + **Verificación de Fugas**:
    - Detectar fugas de combustible según el inventario y las ventas.
  + **Simulación de Ventas**:
    - Asignar surtidores activos para gestionar transacciones aleatorias.
* **Requisitos No Funcionales**:
  + **Usabilidad**: Interfaz amigable para el usuario.
  + **Escalabilidad**: Capacidad para añadir nuevas estaciones de servicio y surtidores.
  + **Rendimiento**: Respuestas rápidas a las consultas y operaciones.
  + **Seguridad**: Control de acceso y manejo seguro de transacciones financieras.

### **3. Modelado del Sistema**

* **Diagrama UML**:



### **4. Identificación de Casos de Uso**

* Crear casos de uso que describan cómo interactuarán los usuarios (administradores y empleados) con el sistema. Cada caso de uso debe incluir:
  + **Rol**: quién interactúa con el sistema (administrador, empleado).
  + **Escenario**: pasos que se siguen para completar una tarea.
  + **Resultados Esperados**: qué debe ocurrir al final del proceso.

#### **Ejemplo de Caso de Uso: Simulación de Venta**

* **Rol**: Empleado
* **Escenario**:
  1. El empleado selecciona una estación de servicio.
  2. Se asigna un surtidor activo aleatorio.
  3. Se solicita una cantidad de combustible (entre 3 y 20 litros).
  4. Se verifica la disponibilidad en el surtidor.
  5. Se realiza la venta y se registran los detalles.
  6. Se muestran los detalles de la transacción.
* **Resultados Esperados**: Se muestra un mensaje confirmando la venta y detalles de la transacción.

### **6. Validación de Requisitos**

* Revisar si todos los requisitos funcionales y no funcionales han sido considerados en el diseño.
* Realizar reuniones con los interesados (stakeholders) para asegurarse de que los requisitos capturados son correctos y completos.

### **7. Pruebas y Mantenimiento**

* Diseñar pruebas unitarias para cada clase y método.
* Planificar la fase de pruebas de integración y funcionales.
* Establecer un plan para el anejo de errores para actualizaciones futuras y corrección de los posibles errores.