**Дизайнер:**

1. Разработать макет окна с подробной информацией о заправке (включая списки работников и ошибок).
2. Нарисовать кнопку «Отключить заправку» на экране заправки.
3. Добавить интерфейс для настройки лимитов потребления топлива.
4. Разработать окно, которое будет показывать ошибку.

**Фронтенд-разработчик:**

1. Реализовать экран авторизации по макету.
2. Реализовать интерфейс для отображения информации о заправках на главном экране.
3. Реализовать кнопку для добавления новой заправки.
4. Реализовать страницу для отображения подробной информации о заправке.
5. Настроить возможность клика по заправке для открытия этой страницы.
6. Реализовать механизм динамического отображения данных о заправке.
7. Реализовать функционал для отключения заправки через интерфейс приложения.
8. Реализовать функционал для установки и изменения лимитов.
9. Реализовать механизм стирания данных при неверной аутентификации.
10. Реализовать функционал изменения цвета границ окна в зависимости от состояния заправки.
11. Реализовать скроллинг по списку заправок.

**Бэкенд-разработчик:**

1. Разработать API для аутентификации (проверка логина и пароля).
2. Настроить связь с базой данных для хранения информации о пользователях.
3. Создать структуру базы данных для хранения информации о заправках.
4. Настроить микросервис для передачи данных о заправках на фронтенд.
5. Реализовать API для получения списка работников и ошибок для конкретной заправки.
6. Реализовать API для уведомления сотрудника.
7. Настроить API для отключения заправки и изменения её статуса в базе данных.
8. Настроить API для сохранения ограничений на потребление топлива в базе данных.
9. Настроить обработку ошибок при неверном вводе логина или пароля на сервере.
10. Реализовать шифрование данных с использованием шифрования и метода TDE (прозрачное шифрование данных).

**DevOps-инженер:**

1. Настроить архитектуру сервера (Трёхуровневая архитектура “Клиент-сервер”).
2. Обеспечить настройку серверной инфраструктуры в соответствии с минимальными требованиями (реализовать виртуализацию – рационально использовать вычислительные ресурсы; реализовать СУБД; реализовать эффективную передачу между клиентом и сервером – сервисы DNS/DHCP; …).
3. Настроить резервирование серверных ресурсов (Дифференциальное резервное копирование. Хранение информации в головном офисе компании).
4. Настроить систему обнаружения вторжений (DDoS-атак).

**Общие требования:**

1. Найти сервер:
   * Процессор: от 4 ядер
   * RAM: от 16 ГБ
   * Хранилище: от 1 ТБ
   * Сетевой интерфейс: от 1 Гбит/сек
   * ОС: Linux.