При первом запуске приложение запрашивает данные для авторизации:

* Фронтенд-разработчик:
  + Реализовать экран авторизации по макету. //добавить картинку макета в описание
* Бэкенд-разработчик:
  + Разработать API для аутентификации (проверка логина и пароля).
  + Настроить связь с базой данных для хранения информации о пользователях.

После авторизации от имени оператора программа отображает:

* Фронтенд-разработчик:
  + Реализовать интерфейс для отображения информации о скважинах на главном экране.
  + Реализовать кнопку для добавления новой скважины.
* Бэкенд-разработчик:
  + Создать структуру базы данных для хранения информации о скважинах.
  + Настроить микросервис для передачи данных о скважинах на фронтенд.

Приложение должно показывать более подробные данные:

* Дизайнер:
  + Разработать макет окна с подробной информацией о скважине (включая списки работников и ошибок).
* Фронтенд-разработчик:
  + Реализовать страницу для отображения подробной информации о скважине.
  + Настроить возможность клика по скважине для открытия этой страницы.
  + Реализовать механизм динамического отображения данных о скважине.
* Бэкенд-разработчик:
  + Реализовать API для получения списка работников и ошибок для конкретной скважины.
  + Реализовать API для уведомления сотрудника (о чём? 25 -> 38 Суббота, 2 ноября).

Программа должна предоставлять возможность отключения скважины:

* Дизайнер:
  + Нарисовать кнопку «Отключить скважину» на экране скважины.
* Фронтенд-разработчик:
  + Реализовать функционал для отключения скважины через интерфейс приложения.
* Бэкенд-разработчик:
  + Настроить API для отключения скважины и изменения её статуса в базе данных.

Программа должна предоставлять возможность ограничения потребления энергии скважиной:

* Дизайнер:
  + Добавить интерфейс для настройки лимитов потребления энергии.
* Фронтенд-разработчик:
  + Реализовать функционал для установки и изменения лимитов.
* Бэкенд-разработчик:
  + Настроить API для сохранения ограничений на потребление энергии в базе данных.

Приложение должно работать на винде 7 или новее, MacOS, Linux:

* Тестировщики:
  + Провести тестирование работы приложения на разных платформах.

При введении неверных данных:

* Дизайнер:
  + Разработать окно, которое будет показывать ошибку.
* Фронтенд-разработчик:
  + Реализовать механизм стирания данных при неверной аутентификации.
* Бэкенд-разработчик:
  + Настроить обработку ошибок при неверном вводе логина или пароля на сервере

Изменение цвета границы окна:.

* Фронтенд-разработчик:
  + Реализовать функционал изменения цвета границ окна в зависимости от состояния скважины.

Приложение должно работать на автономном сервере:

* DevOps-инженер:
  + Настроить архитектуру сервера (Трёхуровневая архитектура “Клиент-сервер”).

Реализовать скроллинг скважин вниз:

* Фронтенд-разработчик:
  + Реализовать скроллинг по списку скважин.

Найти сервер:

* Процессор: от 4 ядер
* RAM: от 16 ГБ
* Хранилище: от 1 ТБ
* Сетевой интерфейс: от 1 Гбит/сек
* ОС: Linux.
* DevOps-инженер:
  + 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; …).

Реализовать в безопасности:

* DevOps-инженер:
  + Настроить резервирование серверных ресурсов (Дифференциальное резервное копирование. Хранение информации в головном офисе компании).
  + Настроить систему обнаружения вторжений(DDoS-атак)
* Бэкенд-разработчик:
  + Реализовать шифрование данных с использованием шифрования и метода TDE(прозрачное шифрование данных)