

Пункт управления (ПК)

В этой папке находится десктопный интерфейс для управления ровером и запуска анализа камней. Интерфейс построен на базе Electron и включает следующие файлы:

- **index.html** – HTML-разметка и стили для пользовательского интерфейса, обеспечивающие отображение элементов управления, видео-потоков и телеметрии.
- **main.js** – Основной JavaScript-файл для Electron, который создает окно, устанавливает соединение с сериал-портом для получения данных с контроллера, а также реализует вызов Python-скрипта **yaan.py**.
- **preload.js** – Скрипт, который с помощью contextBridge безопасно предоставляет API для взаимодействия между основным и рендер-процессами.
- **package.json** – Конфигурация для Electron Forge, описывающая зависимости, скрипты сборки и запуска.
- **yaan.py** – Python-скрипт для анализа камней, вызываемый из основного окна приложения.

Установка и запуск

1. Установите Node.js и npm (если ещё не установлены).
2. В терминале перейдите в папку **Пункт_управления**:

```
cd Пункт_управления
```

3. Установите зависимости:

```
npm install
```

4. Запустите приложение:

```
npm start
```

5. Для запуска анализа камней воспользуйтесь кнопкой "Запустить анализ" в интерфейсе. При сборке приложения скрипт **analyst.py** будет включён в asar-архив (или использован напрямую при разработке).
6. Для сборки приложения:

```
npm run make
```

Функциональность • Управление движением: Интерфейс позволяет задавать линейную и угловую скорость посредством ползунков и клавиатурных команд (W, A, S, D). • Телеметрия: Отображаются данные одометрии ровера, получаемые через ROS-соединение. • Управление сервоприводами: Два режима управления сервоприводами для камеры и манипулятора. • Отображение данных контроллера: Джойстики, кнопки и потенциометры – все данные обновляются и отображаются в режиме реального времени.

Зависимости

- Electron, Electron Forge
- Socket.IO (для связи с ROS-сервером)
- Python (для запуска yaan.py)