ERP – Базовый модуль

Тип

Инфраструктурный модуль для приёма, структурирования и начальной агрегации сырых данных на собственной инфраструктуре.

Назначение

Непрерывно собирает и сохраняет данные о транспортных средствах, водителях, событиях, финансах и обслуживании. Служит единственным источником достоверных данных для всех сырых потоков, питающих аналитические модули, модули допуска и отчётности — при этом инфраструктурные расходы остаются ниже \$1 000/мес. для парка из ~500 машин.

« Структура подсистем

Подсистема	Функция	Ключевые данные
transport_registry	Регистрация TC, учёт автопарка, профиль и статус	VIN, марка/модель, документы, пробег
telemetry_engine	Сбор данных с датчиков (CAN/OBD/GPS)	скорость, координаты, расход топлива, ошибки
driver_module	Профили водителей, документы, рейтинги, отзывы	driver_id, страховка, обратная связь
event_log	Учёт поездок, инцидентов, нарушений	trip_id, дата/время, штрафы, ремонты
maintenance_layer	График и история обслуживания, расходы	дата ТО, список запчастей, стоимость
finance_block	Управление финансовыми потоками	депозит, удержания, выплаты
supply_core	Управление запасами, списания, закупки	остатки, заказы, аналитика
investor_interface	Отчётность перед инвесторами, графики выплат	asset_id, оценка, условия контракта
access_control	Правила доступа TC/водителей, блокировки	статус аренды, лог блокировок, причина
personnel_tasks	Задания для сотрудников парка, КРІ	task_id, роль, исполнитель, срок
interface_layer	Веб- и мобильные интерфейсы для всех ролей	См. ниже
analytics_kpi_engine	Расчёт и агрегация метрик водителей/парка	индекс сна, штрафы, флот, загрузка и т.д.

Питерфейсы

operator portal – веб-приложение для диспетчеров: задачи, мониторинг, оповещения, аварийная блокировка

- **driver_mobile_app** мобильное приложение для водителя: маршруты, статус, загрузка документов
- investor dashboard веб-портал для инвесторов: активы, доходность, выплаты
- admin_console админка: конфигурация, управление пользователями, кнопка "заглушить" (relay через Vialon)

🔁 Режим работы и поток данных

Сбор

- Потоки в реальном времени (5–15 сек): телеметрия, события
- Пакеты (каждые 5–15 мин): финансы, ТО, запасы

2. Хранение

- TimescaleDB/PostgreSQL на 2 VPS (4 vCPU, 8 ГБ RAM, 200 ГБ SSD)
- Срок хранения: сырые данные ≤ 7 дней, агрегаты ≤ 90 дней
- 3. Фичепайплайн

4. Потребители:

Модули: Коридор нормальности, Риск-движок, Коуч, FSM-допуск, IMS, интерфейс инвестора

🌓 Данные на выходе (Feature Store)

Поле	Описание	Источник	Частота обновления
driver_sleep_index	Индекс качества сна водителя	telemetry_engine	real-time / batch
driver_penalty_rate	Штрафы за период	event_log	batch
fleet_size	Размер парка	transport_registry	batch
fleet_active_count	Активные ТС	event_log	batch
avg_rev_vehicle	Средняя выручка с ТС	finance_block	batch
repair_rate	Доля ТС в ремонте	maintenance_layer	batch
idle_rate	Доля простаивающих ТС	registry + log	batch

† АРІ-сценарии

Версия: v1.0 (обратная совместимость) Авторизация: OAuth2 (JWT Bearer)

1. Получить события водителя:

GET /api/v1/event_log/driver/{driver_id}?date=YYYY-MM-DD

2. Последняя телеметрия ТС:

GET /api/v1/telemetry/{vehicle id}?limit=10

3. Статус допуска водителя:

GET /api/v1/access/{driver id}/status

4. Финансовый отчёт по парку:

GET /api/v1/finance/park/{park id}?period=2025-Q2

5. Удалённая блокировка ТС:

```
h
Копировать Редактировать
POST /api/v1/access/{vehicle_id}/immobilize
Content-Type: application/json
{
   "method": "vialon_relay",
   "reason": "safety_lockdown"
}

• Ответ: 202 Accepted → { job_id, status }
• Проверка: GET /api/v1/access/{vehicle id}/immobilize/{job id}
```

Ф SLA, производительность и батчи

- Задержка ingestion: ≤ 30 сек
- API (p95): $\leq 200 \text{ Mc}$
- Пропускная способность: ≥ 1000 RPS
- Обработка батчей: ≤ 5 мин / пакет
- Задержка блокировки: ≤ 15 сек, до 3 повторов при сбое
- Масштабируемость: Docker-реплики, cron-окна, VPS-апгрейды

П Качество данных

Подсистема	Валидация	Дедупликация
telemetry_engine	sanity-чек, фильтры порогов	по таймстампу
event_log	проверка схемы и бизнес-правил	по хешу события
driver_module	уникальность и целостность документов	по driver_id
finance_block	сверка балансов	по transaction_id
maintenance_layer	г логика графика и статусов	по maintenance_id

🔐 Безопасность и защита данных

- Аутентификация: OAuth2 (JWT Bearer)
- Авторизация: RBAC (оператор, водитель, инвестор, админ)

- Только админ / спец-оператор могут блокировать
- Шифрование: TLS 1.2+ (в пути), AES-256 (в покое)
- Уязвимости: SAST/DAST, сканы зависимостей, пентесты
- Реагирование на инциденты: playbooks, патчи
- Аудит: все блокировки логируются (user_id, time, VIN, причина)
- GDPR / PII: псевдонимизация, удаление по запросу

⊗ Наблюдаемость и СІ/СD

- Логи и трассировка: Fluentd → ELK, OpenTelemetry
- Метрики и дашборды: Prometheus + Grafana
- CI/CD: GitHub Actions или Jenkins (lint → test → build → deploy)
- Оркестрация: Docker Compose, cron ETL

📜 Аудит и управление данными

- Логи: immutable-аудит всех операций (Postgres WAL + файлы)
- Хранение: логи и $PII \ge 7$ лет, secure-erase
- COOTBETCTBUE Basel III / ESG / SDG
- SHAP: объяснимость
- Каталог данных: https://confluence.company.com/erp-data-catalog
- Lineage: https://datahub.company.com/lineage

🔁 Бэкапы и восстановление

- Полные бэкапы: ежедневно (pg dump), инкременты каждые 4 ч
- Хранение: ≥ 30 дней
- RPO: < 24 ч; RTO: < 1 ч
- Репликация: резервный VPS + снепшоты
- DR-тесты: раз в квартал с отчётом

🖈 Ключевые сценарии

1. Регистрация ТС

POST /api/v1/transport registry $c\ VIN$ и данными ightarrow 201 Created

2. Оповещение о высоком idle

idle rate > 0.3 в течение 15 мин \rightarrow оповещение Grafana \rightarrow PagerDuty

3. Ежедневный отчёт по парку

Cron: GET /api/v1/finance/park/{park id}?period=today \rightarrow PDF no email

4. Удалённая блокировка

Клик в admin console \rightarrow POST /immobilize \rightarrow relay \rightarrow status-проверка

Проверка состояния

GET /healthz →

```
json
КопироватьРедактировать
{ "status": "ok", "uptime": "72h", "lag ms": 120 }
```

- Правила оповещений (Grafana):
 - avg (api p95) > 200 ms в течение 5 мин → Severity P1 → PagerDuty
 - CPU > 80% дольше 10 мин \rightarrow уведомление в Slack

✓ ERP как инфраструктурная основа цифрового доверия

Представь, что у тебя десятки машин, водителей, операторов, инвесторов — и каждый день что-то происходит: поездки, ремонты, штрафы, выплаты, простои. Это превращается в хаос, если события не фиксируются и не структурируются.

ERP — это система, которая превращает поток действий в проверяемую, наблюдаемую цифровую реальность.



🦴 Что делает ERP

- Собирает и агрегирует все ключевые данные: телеметрию, действия водителей, финансовые потоки, техобслуживание, инциденты.
- Фиксирует каждое действие, связанное с извлечением дохода из актива: от начала поездки до выплаты инвестору.
- Создаёт единый поток событий, верифицированный и пригодный для автоматизированного анализа.
- Даёт возможность дистанционного воздействия например, заглушить двигатель при нарушении условий.
- Обеспечивает управляемость активов и прозрачность операций на всех уровнях.

🥰 Роль в системе Equinomix

В архитектуре **Equinomix** ERP выполняет базовую инфраструктурную функцию:

- Это основной слой фиксации событий физического мира телеметрии, статуса автомобилей, финансовых транзакций, действий людей.
- Все данные, поступающие в систему, проходят через ERP и становятся проверяемыми фактами.
- На их основе модуль коридор нормальности, строит интерпретации траектории поведения и сигнализирует о рисках, аномалиях и отклонениях — уже по наблюдаемым данным.

• Именно ERP создаёт основу доверия, на которой можно принимать решения о допуске, управлении активами и расчётах.

© Что даёт ERP

- Прозрачность все действия фиксируются цифрово, а не на словах.
- Управляемость система знает, что происходит с каждым активом в каждый момент времени.
- **Безопасность** в случае нарушения условий можно ограничить доступ или остановить транспорт.
- Справедливость любые выводы (о рисках, доходности, эффективности) основаны на проверяемых данных.
- **Инвесторская надёжность** можно в любой момент показать, куда ушли деньги, какой актив сколько заработал, что с ним происходит.

Что меняется

До ERP C ERP

Условный учёт, бумажки и таблицы Структурированные потоки событий
Данные недоступны для анализа Всё агрегировано и готово к расчётам
Оценки субъективны Выводы опираются на реальное поведение
Риски обнаруживаются постфактум Возможна ранняя реакция благодаря сигналам
Неясно, кто что делает и почему Полная наблюдаемость действий



ERP — это не просто система учёта, а цифровой фундамент, на котором строится управление токенизированными активами.

Без неё невозможна ни оценка поведения, ни расчёт доходности, ни предиктивная сигнализация.

Это слой наблюдаемости, который делает Equinomix честной, управляемой и защищённой системой.