

Архитектурная документация

Обзор для due diligence · Kolba ЛИМС

Настоящий документ — сокращённый, ориентированный на заказчика обзор архитектуры ЛИМС «Kolba». Полная внутренняя техническая документация (ADR, схема данных, эксплуатационные регламенты) относится к внутренним материалам Лицензиара и может быть предоставлена для технического аудита на условиях NDA по отдельному запросу.

1. Общая схема

Система построена как модульный монолит на Go с чётким разделением на независимо развёртываемые компоненты:

- **core-api** — серверная бизнес-логика: пробы/образцы, ход исследований, реактивы, протоколы, ВЛК и метрология, персонал/RBAC, чат и уведомления, аналитика.
- **gateway** — единая точка входа (маршрутизация, аутентификация сессий, отдача веб-интерфейса), включая публичный лендинг.
- **erp-connector** — изолированный интеграционный слой обмена с учётными системами клиента (1C); работает как отдельный процесс с собственной моделью данных, чтобы сбой или недоступность ERP-контура клиента не блокировали работу лаборатории (лаборатория — не бухгалтерская/учётная система).
- **SPA (веб-интерфейс)** — React/TypeScript, взаимодействует с core-api/gateway по HTTP API.
- **PostgreSQL** — единая СУБД с политиками Row-Level Security (RLS) как основным механизмом изоляции данных между организациями.

2. Мульти-тенантность и изоляция данных

2.1. Иерархия организационных единиц: **Компания** → **Лаборатория**. Каждая запись домена привязана к `company_id` (и, где применимо, `lab_id`); доступ к данным на уровне СУБД разрешается только при совпадении контекста запроса (роль/компания/лаборатория) — по принципу «запрет по умолчанию» (deny-by-default, fail-closed), а не «список запрещённых операций».

2.2. Вендорский контур (административный доступ поставщика к управлению компаниями, тарифами и фиче-флагами) физически отделён от доменных данных клиентов: роль вендор-администратора не имеет прав доступа к данным лабораторий клиентов на уровне СУБД (не «может, но не должен» — не имеет технической возможности).

2.3. Роли пользователей внутри организации клиента строятся по принципу необходимого минимума в соответствии с требованиями ИСО 17025 (в т. ч. обезличивание проб для исполнителей исследований, где это предусмотрено процессом лаборатории).

3. Прослеживаемость и аудит

Все мутирующие операции фиксируются в аудит-логе (кто/что/до/после/когда) — требование прослеживаемости по ИСО 17025. Оценка соответствия результата исследования установленным нормам выполняется лабораторией (пользователем), а не автоматически Системой; Система предоставляет вспомогательный, не окончательный индикатор сопоставления с нормой там, где норма задана.

4. Развёртывание и эксплуатация

4.1. **Модель поставки** — SaaS (эксплуатация в инфраструктуре Лицензиара) либо On-Premises (развёртывание в инфраструктуре Лицензиата тем же Docker-based кодом под управлением подписанного лицензионного файла) — см. Приложения А/В лицензионно-подписного договора. Мульти-тенантная архитектура не является барьером для On-Premises: на одной инсталляции при этой модели фактически работает одна компания.

4.2. Развёртывание — по образам Docker, с проверкой работоспособности (health-check) перед переключением трафика на новую версию и автоматическим откатом при сбое проверки.

4.3. Резервное копирование данных выполняется на регулярной основе с проверенной процедурой восстановления (restore drill).

4.4. Локация хранения данных для SaaS-модели — Казахстан, город Алматы (Приложение А.4 лицензионно-подписного договора).

5. Опциональные ИИ-модули

Система содержит опциональные, по умолчанию выключенные модули на основе внешней языковой модели (перевод вопроса на естественном языке в параметры дашборда, ассистент первой линии техподдержки) — детальное описание границ доступа к данным см. в «Декларации об использовании ИИ».

6. Локализация

Интерфейс поддерживает русский, английский и казахский языки (RU/EN/KK) без зависимости от сторонних сервисов перевода.