

Описание технической архитектуры

Устройство и технологический стек программы

г. Москва · 2026

1. Общая архитектура

Программа построена по трёхслойной архитектуре и работает как единое настольное приложение. Все слои функционируют локально на компьютере пользователя.

- **Слой представления** – графический интерфейс на базе SvelteKit, отображается во встроенном системном компоненте WebView2;
- **Прикладной слой** – ядро приложения на языке Rust (платформа Tauri v2): управление окном, файловыми операциями, лицензией, формированием отчётов и запуском вычислительного модуля;
- **Вычислительный слой** – вычислительный модуль на языке Python: байесовское моделирование, оптимизация, генерация отчётов.

Взаимодействие слоёв – внутреннее: интерфейс обращается к ядру через механизм межпроцессного вызова платформы Tauri, ядро обменивается данными с вычислительным модулем по локальному сетевому интерфейсу (loorback, 127.0.0.1). Внешние сетевые соединения для основной работы не используются.

Единственное обращение программы за пределы компьютера – периодическая проверка наличия новой версии: запрашивается файл описания версий на сервере обновлений правообладателя. Данные пользователя, проекты и результаты расчётов при этом не передаются. При отсутствии подключения проверка не выполняется, и программа сохраняет полную работоспособность.

2. Технологический стек

Компонент	Технология / версия	Назначение
Оболочка приложения	Tauri v2 (Rust, edition 2021)	Настольное приложение, интеграция с ОС
Интерфейс	SvelteKit 2.9, Svelte 5, Vite 6	Графический пользовательский интерфейс
Визуализация	Apache ECharts 5.6	Графики и диаграммы
Вычислительный модуль	Python 3.10+ (упакован в исполняемый файл)	Математические расчёты
Локальный сервис	FastAPI, Uvicorn	Обмен данными ядро ↔ вычислительный модуль
Байесовский движок	PyMC 5.28, PyMC-Marketing 0.19, NumPyro 0.20, JAX 0.7 (CPU)	МСМС-оценка модели MMM
Диагностика	ArviZ 0.23	Показатели сходимости (R-hat, ESS)
Обработка данных	NumPy, pandas, SciPy, scikit-learn	Подготовка данных и оптимизация
Отчёты	python-pptx, openpyxl, XlsxWriter	Формирование PPTX и XLSX
Криптография	AES-256-GCM, HKDF-SHA256, Ed25519	Защита данных и проверка лицензии

3. Вычислительное ядро

Основной метод – байесовская оценка модели маркетингового микса методом MCMC (сэмплер NUTS на движке NumPyro/JAX) с моделью «адсток + насыщение Hill». Для коротких временных рядов применяется откат на метод наименьших квадратов (OLS). Качество оценки контролируется диагностическими показателями сходимости (R-hat, эффективный размер выборки, число дивергенций). Оптимизация бюджета выполняется методами численной оптимизации (SLSQP) с учётом ограничений по каналам.

4. Хранение данных и безопасность

- **Хранение** – проекты и результаты хранятся локально в каталоге пользователя; чувствительные данные кабинетов шифруются алгоритмом AES-256-GCM;
- **Лицензирование** – лицензия проверяется локально по электронной подписи Ed25519 с привязкой к оборудованию;
- **Сетевая изоляция** – сетевое взаимодействие компонентов ограничено локальным интерфейсом (loopback); данные пользователя не передаются на внешние серверы.

5. Используемые сторонние компоненты и лицензии

Программа использует исключительно компоненты с открытым исходным кодом на разрешительных лицензиях. Проприетарные иностранные компоненты, требующие лицензионных выплат, отсутствуют.

Компонент	Лицензия
Tauri, Svelte, SvelteKit, Vite, FastAPI, python-pptx, openpyxl	MIT
PyMC, NumPyro, JAX, ArviZ, Apache ECharts	Apache-2.0
PyTensor, NumPy, SciPy, pandas, Uvicorn, XlsxWriter, Ed25519	BSD (2/3-Clause)
Шрифты Inter	SIL Open Font License 1.1
Инструмент упаковки Python (PyInstaller)	GPL-2.0 с исключением для распространяемых сборок

Итог по лицензиям

Все сторонние компоненты распространяются на разрешительных открытых лицензиях (MIT, Apache-2.0, BSD, ISC, SIL OFL). Выплаты иностранным правообладателям за использование компонентов отсутствуют – критерий реестра о доле иностранных выплат (не более 30 % выручки) соблюдается.