



АНГАР PRO

АнгарПРО

TENT PROJECT

tentproject.ae

KP Engine

KP Engine - Матрица поддержки шаблона

Матрица совместимости placeholder-полей исходного HTML и текущего движка.

Документ KP Engine - Матрица поддержки шаблона	Собран 2026-03-30 01:24	Источник TEMPLATE_SUPPORT_MATRIX.md
---	----------------------------	--

RU-first движок КП мультибренд / мультирынки / интеграции	Документы собраны для продаж, внедрения и руководителя. Они показывают архитектуру, модель переменных, deployment и следующий слой функций без лишнего шума.
--	--

Что это

Этот документ показывает, как KP Engine покрывает placeholder-поля из исходного kp_template_v2_final.html.

Это не общий список полей "на глаз", а слой совместимости между исходным HTML и текущей моделью данных.

Итог по текущей версии

Показатель	Значение
Всего placeholder-полей в исходном HTML	68
Прямые поля	32
Вычисляемые поля	36
Непокрытые поля	0

Как читать статусы

Статус	Что это значит
Прямое поле	значение берется из deal, client, manager, roi или другого runtime-слоя без дополнительной логики
Вычисляется	значение собирается из сметы, brand cases или derived-правил
Gap	для поля нужен новый mapping rule или отдельное поле в модели

Группы полей

Группа	Кол-во	Что внутри
Клиент, менеджер и meta	8	company_name, contact_name, psychotype, manager_phone
Геометрия и параметры проекта	15	width, area, climate_zone, tent_sqm
Референсы и кейсы	12	case_1_name, case_2_city, relevant_case
AI / narrative блок	4	ai_opening_sentence, ai_pain_address
Коммерческие и сметные поля	21	price_frame, price_metal, price_per_sqm_install
ROI и экономика	8	roi_rent_monthly, roi_payback_months

Принцип маппинга

- 1 Из исходного HTML извлекаются все {{placeholder}}.
- 2 Для каждого поля задаются status, sources и resolver.
- 3 Resolver либо берет значение напрямую, либо вычисляет его из estimate, brand.cases или roi.
- 4 Showcase показывает sample values по нескольким кейсам:
- 5 АнгарPRO / RU sale
- 6 RoyalTent / RU rental
- 7 TentProject / AE rental

Что поддерживается напрямую

Прямо покрываются:

- клиент и контакт: company_name, contact_name, psychotype;
- менеджер: manager_name, manager_phone, manager_email;
- мета: kp_date, price_valid_until;
- геометрия и параметры: width, length, height, area, object_type, location, climate_zone, delivery_weeks, production_weeks, install_weeks, tent_color;
- AI / narrative: ai_opening_sentence, ai_pain_address, ai_pain_context, manager_personal_note;
- ROI: roi_rent_monthly, roi_rent_3years, roi_asset_value_3years, roi_downtime_cost, roi_overpay_annual, roi_lost_revenue, roi_payback_months, roi_payback_scenario.

Что вычисляется

Вычисляемые поля опираются на явные правила, а не на ручные догадки:

- кейсы и референсы берутся из brand.cases;
- relevant_case выводится из релевантного кейса бренда;
- tent_sqm, light_qty, gates_qty извлекаются из сметы;

-
- totals и line-items собираются из профильных сметных разделов;
 - удельные показатели считаются из `total / qty` или `total / area`.

Где смотреть полную матрицу

- live-page: `/showcase#template-support`
- JSON API: `/api/showcase/template-support`
- rules: `template_support.py`

Что это дает внедрению

- исходный HTML больше не является черным ящиком;
- sales видит, какие поля можно гарантированно заполнять из CRM;
- интегратор видит, какие derived-правила уже существуют;
- архитектура переменных остается прозрачной при подключении amoCRM, Mango Office и будущего Bitrix24.