

Техническое задание (MVP для хакатона)

Задать вопросы можно в Telegram-чате

<https://t.me/+yfqjuxZCunU2MmMy>

0. Цель проекта

Создать веб-сервис, который ежемесячно формирует и показывает маркетинговые отчёты для проектов клиентов на основе данных **Яндекс.Метрики** и **Яндекс.Директа**, с возможностью будущего расширения на другие каналы.

Ключевые разделы отчёта:

1. **Общая аналитика** (Метрика) с разбивкой по возрасту: Визиты, Посетители, Отказы, Время на сайте.
2. **Контекстная реклама** (Директ): Показы, Клики, CTR, CPC, Конверсии, CPA, Бюджеты — итоги по всем кампаниям и отдельно по каждой кампании.
3. **SEO**: Посетители, Конверсии, Список запросов с позициями (интеграция на этапе MVP — «заглушка/подготовка», источник позиций — TBD).

Правило периода: всегда три календарных месяца: **текущий** и **два предыдущих**. Счёт периода — календарный месяц с 1-го числа; если проект стартует не 1-го, отчёт строится до конца этого месяца.

Динамика: для каждой метрики рассчитывается $(\text{Текущий} - \text{Предыдущий}) / \text{Предыдущий} * 100\%$. Если $\text{Предыдущий} = 0$ — показывать +0.

Технологический стек:

- **Backend:** Laravel (PHP 8.2+).
 - **Frontend (MVP):** HTML + нативный JS.
 - **Frontend (идеально / v1.1):** ReactJS.
 - **БД:** MySQL 8.x.
-

1. Сущности и модель данных

1.1. Проекты и счётчики

- **projects**
 - id (PK)
 - name (string, required)
 - slug (string, unique)
 - timezone (string, default: проектная TZ, например [Europe/Moscow](#) или конфигурируемая)
 - currency (enum, default: [RUB](#))
 - is_active (bool)
 - created_at, updated_at
- **yandex_counters** (1:N к project)
 - id (PK)
 - project_id (FK → projects)
 - counter_id (bigint, required) — ID счётчика Метрики

- name (string, nullable)
- is_primary (bool, default: false)
- created_at, updated_at

- **direct_accounts** (1:N к project)

- id (PK)
- project_id (FK → projects)
- client_login (string, required) — логин клиента в Директе (для заголовка [Client-Login](#))
- account_name (string, nullable)
- created_at, updated_at

- **direct_campaigns** (N:1 к direct_accounts)

- id (PK)
- direct_account_id (FK → direct_accounts)
- campaign_id (bigint, required)
- name (string)
- status (string)
- created_at, updated_at

- **goals** (N:1 к yandex_counters)

- id (PK)
- counter_id (FK → yandex_counters)
- goal_id (bigint, required)

- name (string)
- is_conversion (bool, default: false) — флаг «считаем как конверсионную»

1.2. Хранилище агрегатов (помесячно)

- **metrics_monthly** (общие метрики Метрики по проекту)
 - id (PK)
 - project_id (FK)
 - year (int), month (tinyint)
 - visits (int)
 - users (int)
 - bounce_rate (decimal(5,2)) — %
 - avg_session_duration_sec (int)
 - conversions (int, nullable) — сумма по отмеченным флагом **is_conversion** целям
 - created_at
- **metrics_age_monthly** (разбивка по возрасту)
 - id (PK)
 - project_id (FK)
 - year, month
 - age_group (enum: 18-24, 25-34, 35-44, 45-54, 55+, unknown)
 - visits, users, bounce_rate, avg_session_duration_sec

- **direct_campaign_monthly** (агрегаты Директа по кампаниям)

- id (PK)
- project_id (FK)
- direct_campaign_id (FK → direct_campaigns)
- year, month
- impressions (int)
- clicks (int)
- ctr_pct (decimal(6,2))
- cpc (decimal(12,2)) — в RUB
- conversions (int, nullable)
- cpa (decimal(12,2), nullable) — RUB
- cost (decimal(14,2)) — RUB
- created_at

- **direct_totals_monthly** (итоги по всем кампаниям проекта)

- id (PK)
- project_id (FK)
- year, month
- impressions, clicks, ctr_pct, cpc, conversions, cpa, cost
- created_at

- **seo_queries_monthly** (TBD источник позиций)

- id (PK)
- project_id (FK)
- year, month
- query (string)
- position (int)
- url (string, nullable)

Примечание: таблицы можно дополнить индексами `(project_id, year, month)` и уникальными ограничениями на `(project_id, month, ...)` для идемпотентности загрузки.

2. Интеграции и доступы

2.1. Яндекс OAuth (общая схема)

- Регистрируем приложение в Яндексе → получаем `client_id` и `client_secret` (хранить в `.env`).
- Получаем **OAuth-токен** через авторизацию (для агентства/админа).
- Для Яндекс.Директа используем заголовок `Client-Login:` `<логин_проекта>` для каждого проекта (у вас «у каждого проекта свой аккаунт в Директе»).
- Для Метрики указываем ID счётчиков проекта.
- Токены и client credentials хранить в таблице `integrations_credentials` или в `.env` + KMS/Secrets Manager (прим.: MVP — `.env`).

Что предоставить:

- `YANDEX_CLIENT_ID`, `YANDEX_CLIENT_SECRET` (для OAuth).
- Список проектов с: названием, **логинем Директа**, перечнем **ID счётчиков Метрики**, списком целей (`goal_id`), где `is_conversion = true`.

2.2. Метрика — эндпоинты и поля

- Отчёты API v3:
 - `stat/v1/data` (метрики: `visits`, `users`, `bounceRate`, `avgVisitDurationSeconds`)
 - Срез по возрасту: dimension `ym:s:ageIntervalName` (или эквивалент)
 - Фильтр по дате: три последних календарных месяца
- Цели: `ym:s:goal<goal_id>Visits / ...Conversions` — суммировать по `goal_id`, где `is_conversion = true`.

2.3. Директ — эндпоинты и поля

- Reports API: `/json/v5/reports` (с заголовком `Client-Login`)
 - Разрез: **CampaignId**, **CampaignName**
 - Метрики: `Impressions`, `Clicks`, `Cost`, `CTR`, `AvgCpc`.
 - Конверсии/CPA: либо из Директа (если настроено счёт целей), либо стыкуем с Метрикой (MVP — из Директа, валюта RUB).

3. Расчёты и правила представления

- Три месяца: `M` (текущий), `M-1`, `M-2`.

- **Динамика %:** $(M - M-1)/M-1 * 100\%$, при $M-1 = 0 \rightarrow$ отображать +0.
 - Возрастные группы: 18–24, 25–34, 35–44, 45–54, 55+, unknown (неопределено).
 - Валюта: **RUB**.
 - Метрики трактуются по определению Яндекс.Метрики/Директа, без нашей корректировки.
-

4. UI/UX (MVP)

4.1. Страницы

- **Список проектов** → карточки проектов.
- **Детальная страница проекта** → три раздела:
 1. **Общая аналитика:**
 - Таблица по месяцам М, М-1, М-2 (Визиты, Посетители, Отказы, Время на сайте, Конверсии).
 - Линейные графики по ключевым метрикам (по месяцам).
 - Блок «Возраст»: таблица/столбчатая диаграмма с группами.
 2. **Контекстная реклама:**
 - Итоговая строка по всем кампаниям (Показы, Клики, CTR, CPC, Конверсии, CPA, Бюджет) за М, М-1, М-2.
 - Таблица по кампаниям: каждая строка — кампания, те же метрики и месяцы.

- Индикаторы динамики (стрелки ↑/↓, цвет).

3. **SEO** (TBD источник):

- Посетители, Конверсии (из Метрики — сегмент Органики).
- Таблица запросов с позициями (если источник подключён); в противном случае — заглушка и CSV-импорт.

4.2. Взаимодействие

- Переключатель периода (по умолчанию — текущий месяц + авто-подбор M-1/M-2).
 - Экспорт PDF отчёта (собираем HTML → wkhtmltopdf/Dompdf).
 - RU-интерфейс, базовый брендированный шаблон (логотип проекта при наличии).
-

5. Backend/инфраструктура

- Laravel 10/11, PHP 8.2+, MySQL 8.x.
- Конфиг через `.env`.
- **Планировщик (cron):**
 - Ежедневно ночью: обновление данных M (текущего месяца).
 - В 01:00 первого числа: фиксация итогов прошедшего месяца, добор M-1, сдвиг окна M/M-1/M-2.
- Очереди: Laravel Queue (database/redis) для API-вызовов и парсинга отчётов.

- Логирование: `storage/logs` + уведомления об ошибках в Telegram/Sentry (опция).
 - Кэш: Redis для кэширования неизменных справочников (кампании и пр.).
-

6. API/контроллеры (внутренние)

- `POST /api/projects` — CRUD проектов (админ).
- `POST /api/projects/{id}/counters` — привязка счётчиков Метрики.
- `POST /api/projects/{id}/direct-accounts` — привязка логина Директа.
- `POST /api/projects/{id}/goals` — регистрация целей (`is_conversion=true`).
- `POST /api/sync/{id}` — форс-синхронизация данных для проекта.
- `GET /api/report/{id}` — агрегированный JSON для фронтенда (три месяца).

Контракт `GET /api/report/{id}` (пример):

```
{
  "projectId": 123,
  "periods": ["2025-11","2025-10","2025-09"],
  "metrica": {
    "summary":
    [{"month":"2025-11","visits":1000,"users":800,"bounce":32.1,"avgSec":75,"conv":35}, ...],
    "age":
    [{"month":"2025-11","age":"25-34","visits":300,"users":250,"bounce":30.0,"avgSec":80}, ...]
  },
  "direct": {
```

```
"totals":
[{"month":"2025-11","impressions":50000,"clicks":2500,"ctr":5.0,"cpc":18.5,"conv":60,"cpa":770,"cost":46250}, ...],
  "campaigns": [{"campaignId": 111, "name": "Brand", "rows":
[{"month":"2025-11","impressions":...}]]
},
"seo": {
  "summary": [{"month":"2025-11","visitors":400,"conv":8}],
  "queries": [{"month":"2025-11","query":"пример","position":12,"url":"/page"}]
}
}
```

7. Безопасность и доступы

- Роли (MVP): **admin** (все проекты), **manager** (выбранные проекты), **client** (только свой проект).
 - Авторизация: Laravel Breeze/Fortify (email+password).
 - Доступ по проектам — через pivot **user_project_roles**.
-

8. Критерии приёмки (MVP)

1. Можно завести проект, привязать **1+ счётчиков Метрики** и **1 Direct-логин**.
2. Отчёт по проекту отображает **М, М-1, М-2** по разделам:
 - Общая аналитика: визиты, посетители, отказы, время, конверсии; есть разбивка по возрасту.
 - Контекст: итоги по всем кампаниям и таблица по кампаниям (показы, клики, CTR, CPC, конверсии, CPA, бюджет).

- SEO: блок доступен, данные по позициям — через заглушку/CSV-импорт, посетители/конверсии — из Метрики (органика).
 - 3. Динамика % корректно считается; при **Previous=0** отображается **+0**.
 - 4. Экспорт отчёта в **PDF**.
 - 5. Ежедневная синхронизация и финализация месяца по крону.
-

9. План работ

9.1. Спринт 1 (MVP)

- БД и сущности (projects, counters, direct-accounts, campaigns, агрегаты).
- OAuth авторизация и базовые клиенты API Метрики/Директа.
- Импорт/синхронизация помесечных данных (M, M-1, M-2).
- Веб-страницы (HTML+JS): список проектов, отчёт проекта.
- Экспорт PDF.
- Авторизация (Breeze) и роли (минимум admin/manager/client).

9.2. Спринт 2 (v1.1)

- ReactJS фронтенд.
- Улучшенные графики и фильтры.
- Интеграция сервиса позиций SEO (Топвизор/SE Ranking) + автообновление.

- Уведомления об ошибках в Telegram/Sentry.
 - Доп. разрезы (устройства, площадки).
-

10. Риски и допущения

- Лимиты API Директа/Метрики → очереди, ретраи, бэкофф.
 - Различия атрибуции/задвоение конверсий между каналами.
 - Изменение схем API.
 - Источник SEO-позиций — будет выбран позже; на MVP — заглушка/CSV.
-

11. Что предоставить от заказчика

- `YANDEX_CLIENT_ID` и `YANDEX_CLIENT_SECRET` (OAuth).
 - Список проектов: `Название`, `Client-Login` (Директ), перечень `Counter ID` (Метрика).
 - Перечень `goal_id`, которые считаются **конверсиями**.
 - Логотипы для брендирования (по желанию).
-

12. Приложение: пример .env

```
APP_ENV=production
APP_KEY=base64:***
APP_URL=https://report.example.com

DB_CONNECTION=mysql
```

DB_HOST=127.0.0.1
DB_PORT=3306
DB_DATABASE=reports
DB_USERNAME=reports
DB_PASSWORD=***

YANDEX_CLIENT_ID=***
YANDEX_CLIENT_SECRET=***
YANDEX_OAUTH_TOKEN=*** # токен агентства/админа
YANDEX_DEFAULT_CURRENCY=RUB
DEFAULT_TIMEZONE=Europe/Moscow

13. Открытые вопросы (для финализации ТЗ)

1. Источник **SEO-позиций** (сервис или CSV) и регион/устройство?
 2. Таймзона проекта по умолчанию (оставляем Moscow или делаем конфигурируемой на уровне проекта)?
 3. Нужна ли авто-рассылка PDF отчёта (email/Telegram) после закрытия месяца?
 4. Список точных **goal_id** по каждому счётчику, которые считаем конверсией.
 5. Нужны ли доп. разрезы по Директу (устройства/площадки) в MVP или отложить во v1.1?
-

Актуализация по итогам уточнений

- **Возрастные группы:** используем **стандартные** из Яндекс.Метрики (как в интерфейсе и API), без кастомных интервалов.

- **Контекст (Директ):** делаем отчёт **только по кампаниям** (доп. разрезы отложены на v1.1).
- **Экспорт PDF:** не требуется в рамках MVP.
- **Доступы:** админ управляет проектами через **Filament**, клиент видит **публичную ссылку отчёта** (только просмотр).
- **Синхронизация:** запуск **по крону 1-го числа каждого месяца в 07:00 MSK** (закрытие прошедшего месяца и обновление окна M/M-1/M-2).
- **Хранение данных: постоянное** (без срока давности), храним помесечные агрегаты, при необходимости расширяем до сырья.
- **Масштаб запуска:** ориентировочно **~30 проектов**.
- **Оформление:** предоставите дизайн/гайд — применим к веб-отчёту.
- **Интеграции:** на MVP **обязательны** Яндекс.Метрика и Яндекс.Директ.
- **Сроки реализации: 3 дня** (хакатон).