

Техническое ТЗ (high level) sub.osipenkov.ru

1. Цели проекта

- Цель сайта: посадочная + личный кабинет для подписчиков, выдача бесплатного и платного контента, сбор базы подписчиков и связка с Telegram (бот + канал).
- Основной сценарий пользователя:
 1. Заходит на sub.osipenkov.ru.
 2. Авторизуется через Telegram Login Widget.
 3. Попадает в ЛК с меню: Подписка в Telegram, Бесплатный контент, позже Платный контент.
 4. Подписывается на бота и на канал @clicksider.
 5. Получает доступ к таблице бесплатного контента, выбирает элементы и нажимает Отправить, чтобы бот прислал материалы.
 6. Если отписывается от канала/бота – доступ к кнопкам закрывается до повторной подписки.

2. Функциональные требования

2.1. Авторизация через Telegram

- Telegram Login Widget на главной:
- Отображается кнопка логина.
- После успешного логина виджет возвращает на сайт данные пользователя (id, username, first_name, last_name, photo_url, auth_date, hash).
- Проверка подлинности:
- Бэкенд проверяет подпись данных по официальному алгоритму Telegram (HMAC-SHA256 с бот-токеном).
- Создание/обновление пользователя в БД:
- Если пользователя с таким telegram_id нет – создаётся запись.
- Если есть – обновляются имя, username, avatar и время последнего логина.
- Сессия:
- После валидации сервер создаёт сессию (cookie / JWT), связывающую HTTP-сессию с telegram_id.

2.2. Личный кабинет

- Доступ:
- Только авторизованные пользователи.
- Левое меню (на всех страницах ЛК):
- Подписка в Telegram (или Подписка на бота).
- Бесплатный контент.
- Платный контент (можно как заглушку на первом этапе).

2.3. Вкладка «Подписка в Telegram»

- Цель: провести пользователя через 2 шага:
- Подписка на бота.
- Подписка на канал @clicksider.
- Функционал:
- Кнопка/ссылка «Открыть чат с ботом» (deep link https://t.me/имя_бота?start=<token>).
- Инструкция: «Нажмите Start в боте».
- Кнопка/ссылка «Подписаться на канал @clicksider».
- Инструкция: «Подпишитесь на канал».
- Статусы:
- Показывать статус «Подписка на бота: да/нет».
- Показывать статус «Подписка на канал: да/нет».
- Кнопка «Проверить подписку»:
- Отправляет запрос на сервер.
- Сервер через Telegram Bot API проверяет:
- Наличие пользователя в БД с пометкой bot_subscribed = true.
- Статус участника в канале через getChatMember.
- Обновляет статусы в БД и возвращает их фронтенду.

2.4. Вкладка «Бесплатный контент»

- Отображение:
- Таблица/список с колонками:
- Название.
- Тип (книга, чек-лист, видео и т.д. – опционально).
- Описание/краткий текст.
- Кнопка Отправить (если подписки ок).
- Если пользователь:
- Не подписан на бота или канал → вместо кнопок Отправить показывать сообщение/ссылку на вкладку подписки.
- Логика кнопки Отправить:
- По нажатию фронтенд вызывает API /api/free-content/send.
- Передаёт content_id.
- Сервер:
- Проверяет, что пользователь авторизован.
- Проверяет статусы bot_subscribed и channel_subscribed (и при необходимости, свежесть проверки, иногда обновляет через Bot API).
- Если всё ок:
- Отправляет соответствующий контент пользователю в Telegram через Bot API:
- sendMessage, sendPhoto, sendVideo, sendDocument и т.д.
- Логирует в БД факт выдачи (user_content_log).
- Если не ок:
- Возвращает ошибку, фронтенд показывает сообщение и предлагает перейти во вкладку подписки.

2.5. Вкладка «Платный контент» (MVP как заглушка)

- На первом этапе:
- Просто список анонсов, без покупки.
- Ссылки на страницы лендингов платных курсов (на основном сайте или внешней платформе).
- В будущем:
- Можно добавить оплату, статусы покупки, защищённый доступ и т.п.

2.6. Telegram-бот

- Регистрация/связка:
- Обрабатывает /start с deep-link payload (например, start web) — фиксирует, что пользователь запустил бота.
- Сохраняет/обновляет данные:
- telegram_id, username, first_name, language_code.
- Ставит флаг bot_subscribed = true (до тех пор, пока бот не получит indication об удалении/блокировке).
- Проверка подписки на канал:
- Сервер (или сам бот по запросу) делает getChatMember(@clicksider, telegram_id):
- Статусы, которые считаем «подписан»: member, creator, administrator.
- Статусы, которые считаем «не подписан»: left, kicked.
- Отправка контента:
- По запросу веб-сервера бот отправляет:
- Текст, фото, видео, документы, ссылки – в зависимости от типа контента.
- Обработка ошибок:
- Если 403 / «bot was blocked by the user» – обновить bot_subscribed = false в БД.
- Если пользователь не подписан на канал, бот может также прислать информативное сообщение.

2.7. Админ-часть (минимум для старта)

- Может быть отдельной защищённой страницей (базовый логин/пароль).
- Функции:
- CRUD по бесплатному контенту:
- Название.
- Тип.
- Описание.
- Тип отправки (текст/фото/видео/документ/ссылка).
- Поле с payload (текст, URL файла, ID медиа в Telegram и т.д.).
- Просмотр списка пользователей:
- telegram_id, username, дата регистрации, статус подписок.
- Просмотр логов выдачи контента (по пользователям и по контенту).

3. Нефункциональные требования

- Производительность:
 - На старте достаточно поддержки нескольких сотен одновременных пользователей.
 - Ответы API по ЛК $\leq 300\text{--}500$ мс.
- Безопасность:
 - Обязательная валидация подписи Telegram Login.
 - Все запросы к сайту – по HTTPS.
 - Защита админки (сильный пароль, IP-ограничение при желании).
- Надёжность:
 - Логирование ошибок сервера и бота.
 - Обработка timeouts/ошибок Telegram Bot API с повторными попытками (на отправку рассылок, контента).
- Масштабирование:
 - Возможность вынести бот и веб-сайт на разные сервисы/контейнеры.
 - Возможность использовать очередь для массовых рассылок.