**Техническое задание на разработку «Видеохостинг с инструментом для создания и оптимизации видео-контента»**

**1. Введение**

1.1 **Цель документа**

Данный документ описывает требования к разработке видеохостинга с интегрированным инструментом для создания, редактирования и SEO-оптимизации видео-контента. Система должна предоставлять пользователям возможность загружать, публиковать и просматривать видео, а также взаимодействовать с контентом (ставить лайки, дизлайки, оставлять комментарии и подписываться на каналы). Кроме того, система будет включать встроенные инструменты SEO-оптимизации, которые помогут пользователям улучшать видимость своих видеороликов в поисковых системах и увеличивать их охват, что позволит создать более эффективную платформу для видео-контента.

1.2 **Актуальность и назначение проекта**

С ростом популярности видео-контента на различных цифровых платформах, создание эффективного и удобного видеохостинга становится важной задачей. Видеохостинг будет направлен на предоставление широкого функционала для взаимодействия пользователей с контентом, а также предложит встроенные механизмы SEO-оптимизации для улучшения поиска и продвижения видео.

1.3 **Основные пользователи системы**

Система предназначена для использования следующими категориями пользователей:

* **Неавторизованные пользователи —** могут просматривать видео, но не могут взаимодействовать с контентом (ставить лайки/дизлайки, оставлять комментарии, подписываться на каналы) или загружать свои видео.
* **Авторизованные пользователи** — могут загружать видео, взаимодействовать с контентом (ставить лайки/дизлайки, оставлять комментарии), подписываться на каналы.
* **Администраторы** — имеют полный доступ к системе, могут управлять пользователями, модерацией контента и настройками платформы.

**2. Цели и задачи проекта**

2.1 **Основные цели**

* Разработка видеохостинга с возможностью загрузки, просмотра и продвижения видео.
* Интеграция инструментов SEO-оптимизации для улучшения видимости видеоконтента в поисковых системах.
* Создание удобного и интерактивного интерфейса для пользователей (лайки, дизлайки, комментарии, подписки).
* Обеспечение масштабируемости платформы, способной обслуживать большое количество пользователей и видео.

2.2 **Конкретные задачи**

* Разработка функционала для загрузки, редактирования и управления видео-контентом.
* Внедрение системы рекомендаций для персонализированного контента.
* Интеграция инструментов SEO-оптимизации для повышения видимости видео.
* Реализация эффективной системы поиска и фильтрации видеоконтента.
* Разработка панели администратора для управления пользователями и модерации контента.

*Пример:* Система должна позволять пользователю загружать видео, редактировать его метаданные (название, описание, теги) и изменять настройки видимости (публичное или приватное видео). Пользователи могут архивировать или удалять свои видео. Администратор должен иметь возможность модерации контента, управлять пользователями и удалять видео, нарушающие правила.

**3. Требования к системе**

3.1 **Функциональные требования**

* **Загрузка и управление видео:** пользователи могут загружать видео, редактировать его метаданные (название, описание, теги), удалять и архивировать видео.
* **SEO-оптимизация:** встроенные инструменты для добавления ключевых слов, тегов и описаний.
* **Поиск и фильтрация:** система должна поддерживать поиск, по ключевым словам, категориям, тегам и другим меткам.
* **Интерактивность:** возможность ставить лайки и дизлайки, оставлять комментарии, подписываться на каналы.
* **Подписка на каналы:** пользователи могут подписываться на каналы, получать уведомления о новых видео.
* **Панель администратора:** возможность управления пользователями, модерации контента и настройками платформы.

*Пример*: Система должна позволять пользователям загружать, редактировать и управлять видео, а также добавлять SEO-оптимизированные метаданные (теги, описание). Пользователи могут ставить лайки, дизлайки, оставлять комментарии и подписываться на каналы. Также должна быть возможность поиска и фильтрации контента. Администраторы управляют пользователями, контентом и настройками платформы.

3.2 **Нефункциональные требования**

* **Производительность:** отклик интерфейса не более 2 секунд при просмотре видео.
* **Масштабируемость:** система должна поддерживать 5000+ пользователей и обеспечивать стабильную работу при высоких нагрузках.
* **Безопасность:** поддержка HTTPS для защиты данных, защита контента от несанкционированного доступа.

**4. Требования к пользовательскому интерфейсу**

4.1 **Основные экраны**

* **Главная страница:** отображение популярных видео, новинок, категорий.
* **Страница видео:** подробная информация о видео, возможность ставить лайки/дизлайки, оставлять комментарии, подписываться на канал.
* **Панель пользователя:** интерфейс для загрузки, редактирования и управления своими видео.
* **Панель администратора:** интерфейс для управления пользователями, модерации контента и настройки платформы.

4.2 **Навигация**

Система должна включать меню навигации с разделами:

* «Главная».
* «Категории».
* «Поиск».
* «Панель блогера».
* «Панель администратора».
* «Мои видео».
* «Настройки».
* «Подписки».

4.3 **Юзабилити**

* Простой и интуитивно понятный интерфейс для всех типов пользователей.
* Поддержка адаптивного дизайна для мобильных устройств.
* Удобный и функциональный плеер для воспроизведения видео.

*Пример*: на главной странице видео отображаются в виде списка.

**5. Требования к технической реализации**

5.1 **Языки и технологии**

* **Backend**: Laravel.
* **База данных**: MySQL.
* **Видеоплатформа**: использование облачных решений для хранения видео, таких как AWS S3 или Google Cloud Storage.

5.2 **Архитектура системы**

* Микросервисная архитектура с REST API для взаимодействия между модулями.
* Хранение метаданных в реляционной базе данных.

5.3 **Интеграции**

* Внешние интеграции для SEO-оптимизации (например, через API Google для анализа ключевых слов).
* Интеграция с внешними сервисами для аутентификации (например, Google OAuth).

*Пример*: Система должна интегрироваться с внешними сервисами для SEO-оптимизации и аутентификации, используя API для анализа ключевых слов и Google OAuth для авторизации пользователей, обеспечивая возможность легко масштабировать и обновлять эти интеграции при необходимости.

**6. Требования к безопасности**

6.1 **Аутентификация и авторизация**

* **Аутентификация**: OAuth для обеспечения безопасности учетных записей пользователей.
* **Авторизация**: Разграничение прав доступа между типами пользователей — неавторизованные пользователи, авторизованные пользователи и администраторы.

6.2 **Шифрование данных**

* **HTTPS** для всех передаваемых данных.
* **Хранение паролей**: хэширование через bcrypt.

6.3 **Политики доступа**

* Защита от SQL-инъекций и XSS.
* Ограничение по IP для административного доступа.

*Пример*: Пользователи не могут загружать, редактировать или удалять видео других пользователей, если у них нет соответствующих прав. Администраторы могут управлять всеми пользователями и контентом, в то время как авторизованные пользователи имеют доступ только к своим видео и действиям с ними.

**7. Ограничения и допущения**

7.1 **Технические ограничения**

* Серверная часть должна работать на облачных платформах (например, AWS или Azure).
* Система должна поддерживать видео форматов MP4, AVI, MOV.
* Ограничение на максимальный размер загружаемого видео — 5 ГБ.

7.2 **Финансовые ограничения**

* Бюджет проекта: до 150000 ₽, включая тестирование и документацию.

7.3 **Сроки выполнения**

* Полная реализация проекта — 6 месяцев с начала разработки.

*Пример*: в рамках ограничения по бюджету допустимо использовать только open-source решения для разработки.

**8. Требования к тестированию и приемке**

8.1 **Типы тестирования**

* **Функциональное тестирование**: проверка всех реализованных функций.
* **Нагрузочное тестирование**: тестирование на работе с 5000+ пользователями.
* **Тестирование безопасности**: проверка на уязвимости.
* **Тестирование пользовательского интерфейса (UI)**: проверка удобства и интуитивности интерфейса на различных устройствах.

8.2 **Критерии приемки**

* Полное выполнение всех функциональных и нефункциональных требований.
* Прохождение не менее 95% тестов по результатам приемочных испытаний.

*Пример*: если в процессе тестирования будет выявлено более 5 критических ошибок, система не может быть принята в эксплуатацию.

**9. Требования к документации**

9.1 **Пользовательская документация**

* **Руководство пользователя**: инструкция по регистрации, входу в систему, загрузке видео, редактированию метаданных, настройке видимости видео и SEO-оптимизации.

**Разделы**:

* Начало работы: создание аккаунта, интерфейс и навигация по платформе.
* Загрузка и управление видео: как загружать, редактировать и удалять видео, а также архивировать их.
* SEO-оптимизация: описание встроенных инструментов для добавления тегов, описаний и ключевых слов.
* Взаимодействие с контентом: как ставить лайки, дизлайки, оставлять комментарии и подписываться на каналы.
* Управление подписками: как подписываться на каналы и получать уведомления о новых видео.

9.2 **Техническая документация**

* **Архитектура системы:** описание архитектурных решений и принципов работы платформы, включая взаимодействие компонентов и системы API.
* **Структура базы данных:** схема и описание таблиц базы данных, включая таблицы пользователей, видео, комментариев, подписок и SEO-метаданных.
* **API документация:** описание всех доступных API-интерфейсов для взаимодействия с системой (например, API для загрузки видео, получения информации о канале, комментариях, поиске видео).
* **Процесс развертывания:** инструкции для развертывания системы на сервере, настройка окружения и зависимостей.

*Пример*: Руководство пользователя должно включать разделы «Начало работы», «Управление видео», «SEO-оптимизация» и «Подписки», с четкими и подробными инструкциями, чтобы пользователи могли быстро освоить систему и эффективно использовать все её функции.

**10. План реализации**

10.1 **Этапы разработки**

1. Анализ и проектирование — 1 месяц.
2. Разработка прототипа — 1 месяц.
3. Полноценная разработка и тестирование — 3 месяца.
4. Внедрение и обучение — 1 месяц.

10.2 **Сроки выполнения этапов**

* Дата начала: 1 марта 2024 года.
* Дата завершения: 31 августа 2024 года.

10.3 **Ответственные лица**

* Менеджер проекта: Шаехов М.Ф.
* Ведущий разработчик: Шаехов М.Ф.

*Пример*: Этап «Полноценная разработка и тестирование» включает разработку всех функциональных модулей и последующее тестирование каждого из них.

**11. Приложения**

* **Приложение A**: Детализированные схемы архитектуры.
* **Приложение B**: Примеры интерфейсов.
* **Приложение C**: Технические спецификации для интеграции с внешними системами (например, API для SEO-оптимизации).