

Здравствуйте, уважаемые эксперты! Вас приветствует команда КЛАН Д со своим решением текстового поиска по медиаконтенту. Основной упор в разработанном решении сделан на поиск, поэтому пользователя на главной странице сервиса встречает поисковая строка.

В эту строку пользователь может ввести любой текст для поиска на русском языке. Для примера введем «Горы». Для осуществления поиска можно нажать либо кнопку Enter, либо на данную кнопку для поиска.

После этого пользователь перейдет на новую страницу с полученными ТОП-10 результатами.

На данной странице есть хэдер, в котором присутствует логотип, поисковая строка, а также кнопка для перехода на страницу для загрузки видео.

В левой части представлено видео, в котором по умолчанию выключен звук. При необходимости его можно включить, поставить видео на паузу, скачать, изменить скорость воспроизведения. В правой части представлено описание видео.

Обратим внимание, что белым текстом отображен `short_description`, показывающий, краткое описание видео, которое сгенерировано нашей моделью. Зеленым текстом – описание, которое ввел пользователь при загрузке данного видео. Если мы не видим зеленого текста, значит пользователь не ввел описание.

В данном примере можем заметить, что пользователь в описании указывал точно не горы, но несмотря на это наша система смогла найти данное видео и корректно его отобразить.

Пролистаем ниже и ознакомимся с другими результатами поиска. Можно удостовериться, что на каждом из видео представлены действительно горы.

Помимо этого, в нашем сервисе реализована мобильная версия, а также возможность поиска с подсказками. Произведем поиск с подсказками и посмотрим за какое время он отработает. Обратим внимание, что поиск отрабатывает не более 500 миллисекунд, как и указано в ТЗ.

Перейдем на страницу загрузки видео. В нашем сервисе видео можно загрузить двумя способами: либо через файл, либо через ссылку.

При этом реализован драг-н-дроп функционал для загрузки с помощью файла. Пользователь может выбрать файл и перетащить его. Также реализована защита от загрузки файлов неподходящих по формату. Для примера загрузим `docx`. Видим окно, сигнализирующее то, что нельзя загрузить данный формат файла. Загрузим корректный файл, выбрав его. Добавим описание. С помощью кнопки опубликовать отправим данное видео в развернутый `Minio` на нашем стенде. После успешной загрузки видео можем увидеть таблицу, в которой представлены данные, которые наша модель собрала по этому видео. Это описания на русском и английском, короткие описания, ...

Перейдем к загрузке с помощью ссылки. Для этого скопируем ссылку на видео, добавим описание. Также можно отследить за какое время произойдет эта загрузка. Нажмем опубликовать. Согласно ТЗ на прием видео и индексацию выделяется 5 минут, видим, что наша система справилась с этим в разы быстрее.

Также хотелось бы продемонстрировать вам реализацию нашего OpenApi. Swagger поделен на два сервиса, первый сервис отвечает за поиск, в нем представлены:

`/search` для поиска видео согласно предоставленной вами спецификации

`/videos/suggest` для подсказок при вводе, которое выполнено в качестве доп.задания.

Второй сервис отвечает за загрузку видео. В нем есть

/Index, который соответствует описанной вами спецификации и служит для загрузки файлов по ссылке.

Predict-by-file для загрузки файла в развернутый Minio на нашем стенде. Используется для фронта

На этом демонстрация нашего сервиса подходит к концу. С остальной информацией о нашем решении можно будет ознакомиться в предоставленной нами документации. Спасибо за внимание.