Доценту кафедры ИКД

Громову В.В.

Лабораторная работа №7

Я, Агарков Вадим Александрович, в ходе лабораторной работы №7 написал Shell-скрипт для скачивания видео с удаленного сервера и их обработки с целью уменьшения их размера и, как следствия, более быстрой их подгрузки на сайт, а также создал страничку на сайте, где и разместил обработанные видео для просмотра.

Видеофайлы были загружены с удаленного сервера при помощи утилиты wget. Для этого я просто разместил ссылки для скачивания в текстовом файле, каждая ссылка на отдельной строчке.

В самом начале выполнения скрипта загружаются видео с ссылок из подготовленного файла, а затем обрабатываются. Также предусмотрен случай, когда видео длится более 10 минут. В таком случае видео делится на фрагменты, которые обрабатываются по-отдельности (в моем случае все видеофрагменты по длительности были менее 10 минут).

Кратко рассмотрим использованные в скрипте утилиты для обработки видео:

* **ffmpeg**: используется для разбиения и конвертации видео. Команда ffmpeg -i используется для чтения входного видеофайла, а -ss и -to используются для указания начального и конечного времени сегмента. -c copy используется для копирования видео без изменения кодека. Второй вызов ffmpeg конвертирует сегменты в формат WebM.
* **ffprobe**: используется для получения длительности видеофайла. Команда ffprobe -i "$video\_file" -show\_entries format=duration -v quiet -of csv="p=0" используется для получения длительности видео в секундах.
* **bc**: используется для арифметических вычислений, особенно для вычисления количества сегментов и временных меток разбиения видео. Например, вычисление числа сегментов происходит так: num\_segments=$(echo "($duration + $segment\_duration - 1) / $segment\_duration" | bc).

Ниже представлен код Shell-скрипта с реализацией вышеописанной логики (см. рисунок 1).

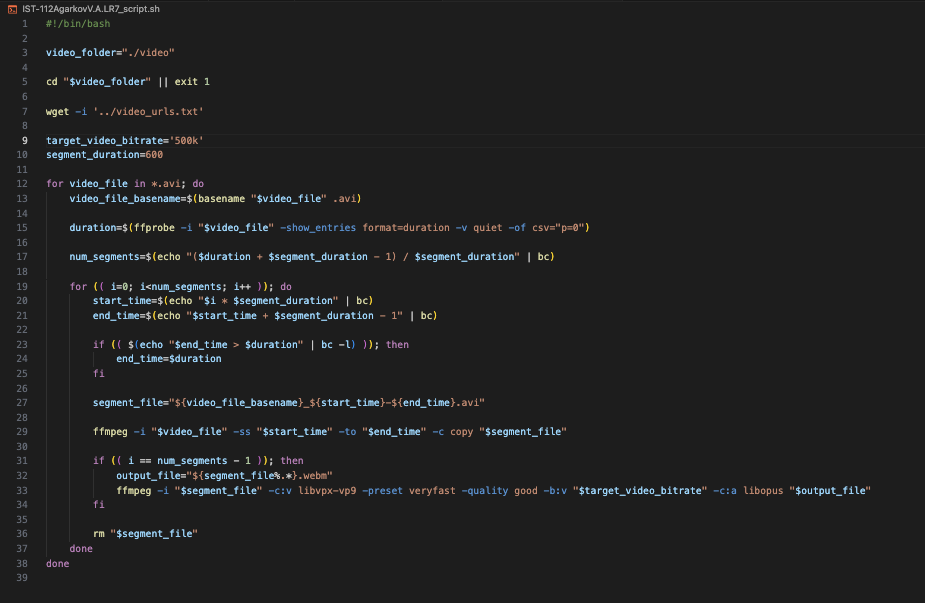


Рисунок 1 – код Shell-скрипта для обработки фотографий

После выполнения данного скрипта, что по времени заняло около 3 часов) в директории video с интернета загрузились видео, а затем были обработаны вышеописанными командами. Обработанные видео были перенесены в директорию сайта по пути /assets/video. В директории /assets/photos были размещены обработанные фотографии из лабораторной работы №6.

Что касается подгрузки фото и видео на сайт, то ее я решил сделать динамической. Логика подгрузки контента была реализована нативным JavaScript без использования сторонних библиотек, что гарантирует стабильность и работоспособность сайта (конечно, при условии, что JavaScript у пользователя в браузере не заблокирован).

Для страниц с фото и видео был написан универсальный скрипт с подгрузкой обоих видов данных. Код скрипта на языке JavaScript представлен ниже (см. рисунок 2).



Рисунок 2 – JS-скрипт для подгрузки фото и видео на сайт.

Также были немного переписана HTML-верстка сайта и CSS-стили к ней. Светлая тема была заменена на темную, а также была сделана простенькая шапка сайта с ссылками для перехода на страницы с фото и видео. В итоге сайт заиграл совершенно новыми красками. Скриншот новой страницы сайта с видео представлен ниже (см. рисунок 3).

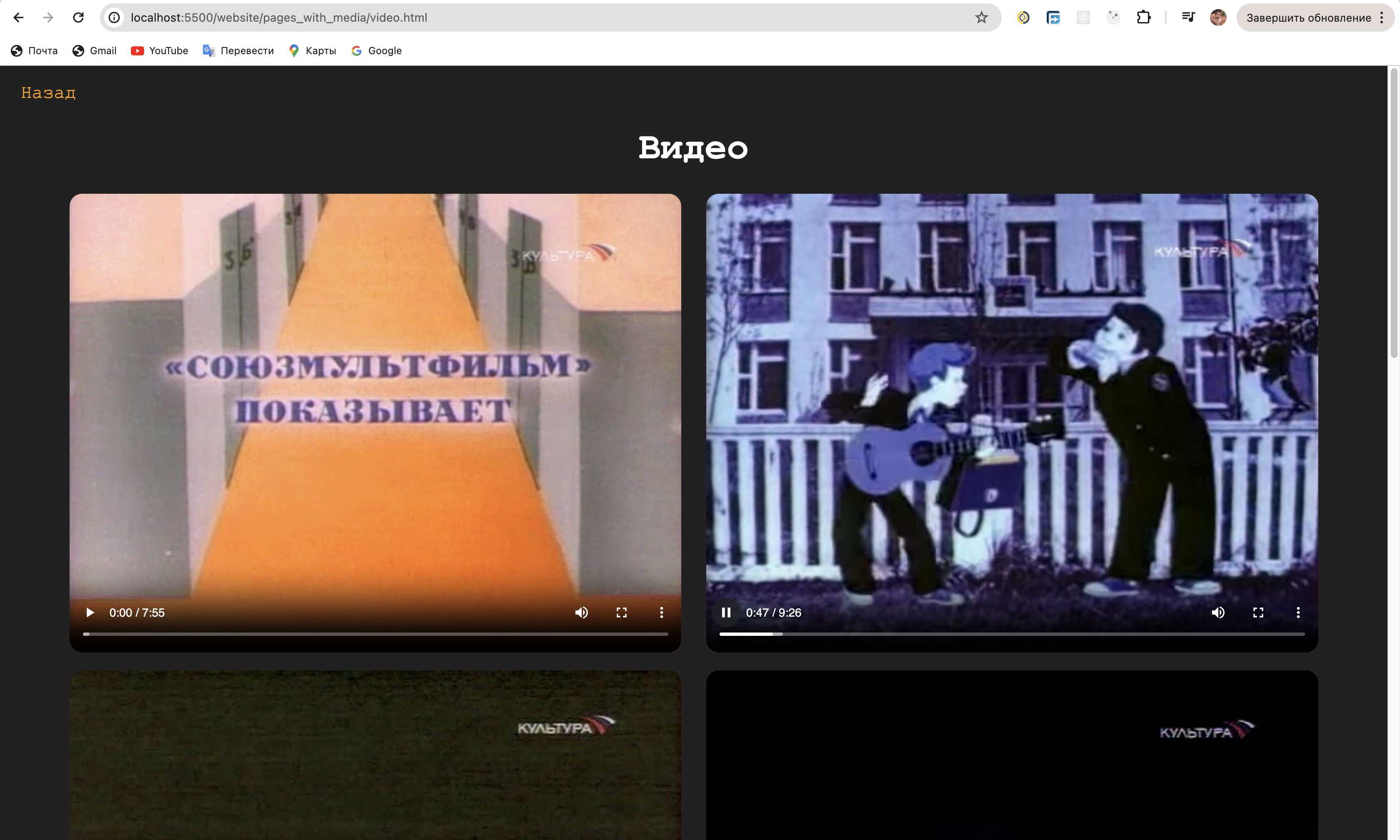


Рисунок 3 – страница с обработанными видео

В ходе выполнения лабораторной работы №7 с помощью Shell-скрипта были загружены, обработаны и выложены на личный сайт видео с ранее заготовленных ссылок.

Стоит отметить, что для данной задачи можно полностью исключить использование языка JavaScript, статически сгенерировав HTML странички с видео или прописав вручную верстку для всех фото и видео. Однако в таком случае решение будет не совсем универсальным и каждый раз при изменении фото- и видеофайлов (имена файлов, их добавление/удаление). А JavaScript как раз предназначен для динамической подгрузки данных на сайт и не требует для такой тривиальной задачи каких-либо зависимостей, так как интерпретатор JavaScript уже встроен в браузер, не требует дополнительных инсталляций и позволяет подгружать данные стандартными средствами языка.

|  |  |
| --- | --- |
| Студент группы ИСТ-112 | Агарков В.А. |