# Группа ЭФБО-01-22

## Структура и описание проекта “Web-Видеоредактор”

### Основная идея проекта

Предполагается использование библиотеки *FFmpeg.wasm* для операций с медиа файлами. Для описания возможностей создаётся прототип продукта с упрощённым интерфейсом и функциями.

### Структура проекта

В корне папки находится файл *server.js*, который используется для создания сервера. Сервер отправляет содержимое папки *front*. Для запуска *server.js* необходимо установить *Node.js* и выполнить команды

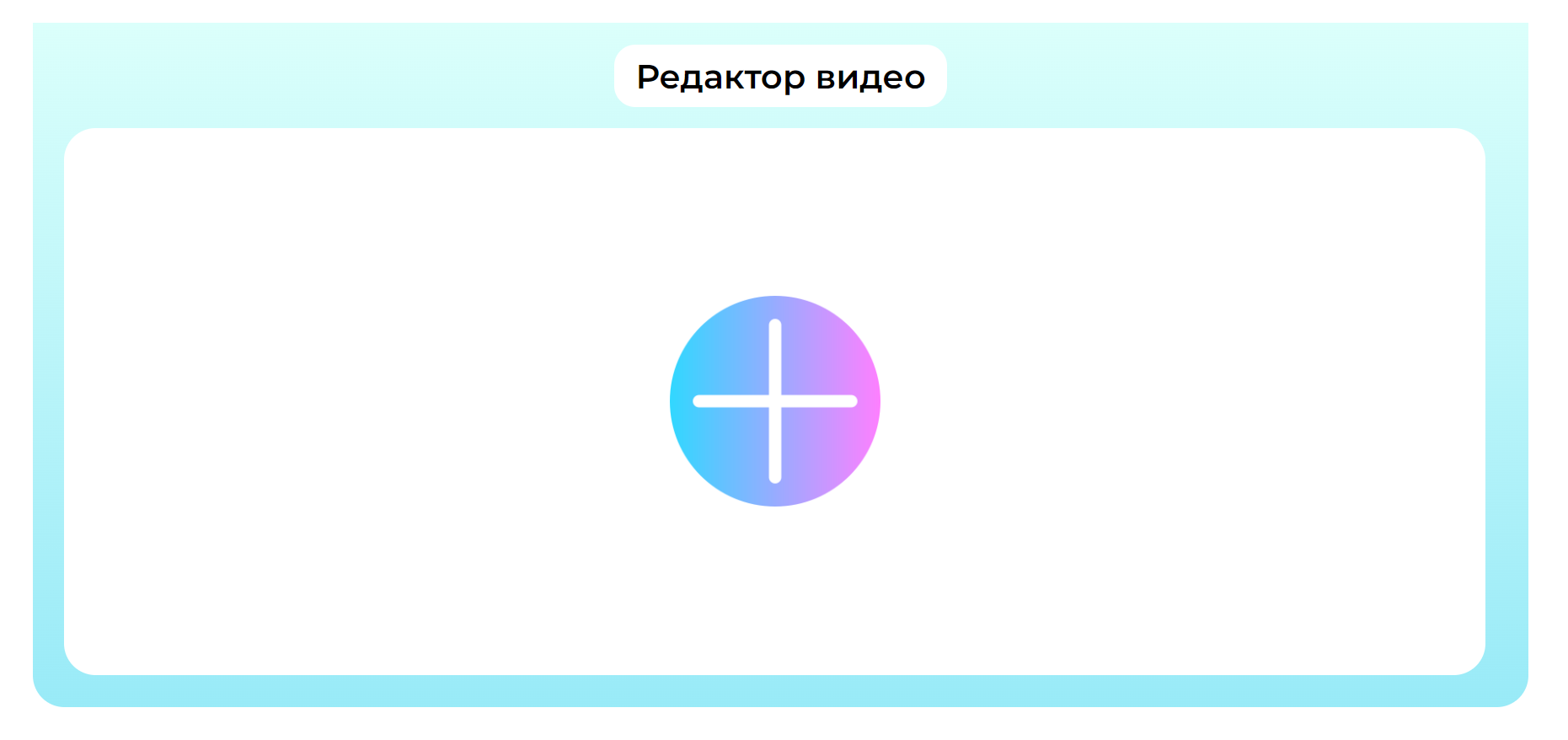
*npm install*

*node server.js*

после этого страничка с продуктом будет доступна по адресу [*http://localhost:5000/pages/index.html*](http://localhost:5000/pages/index.html)*.*

в папке front содержатся папки *pages* с вёрсткой, *styles* со стилями и *scripts* с логикой, а также папка *node\_modules*, которая содержит библиотеку *ffmpeg*.

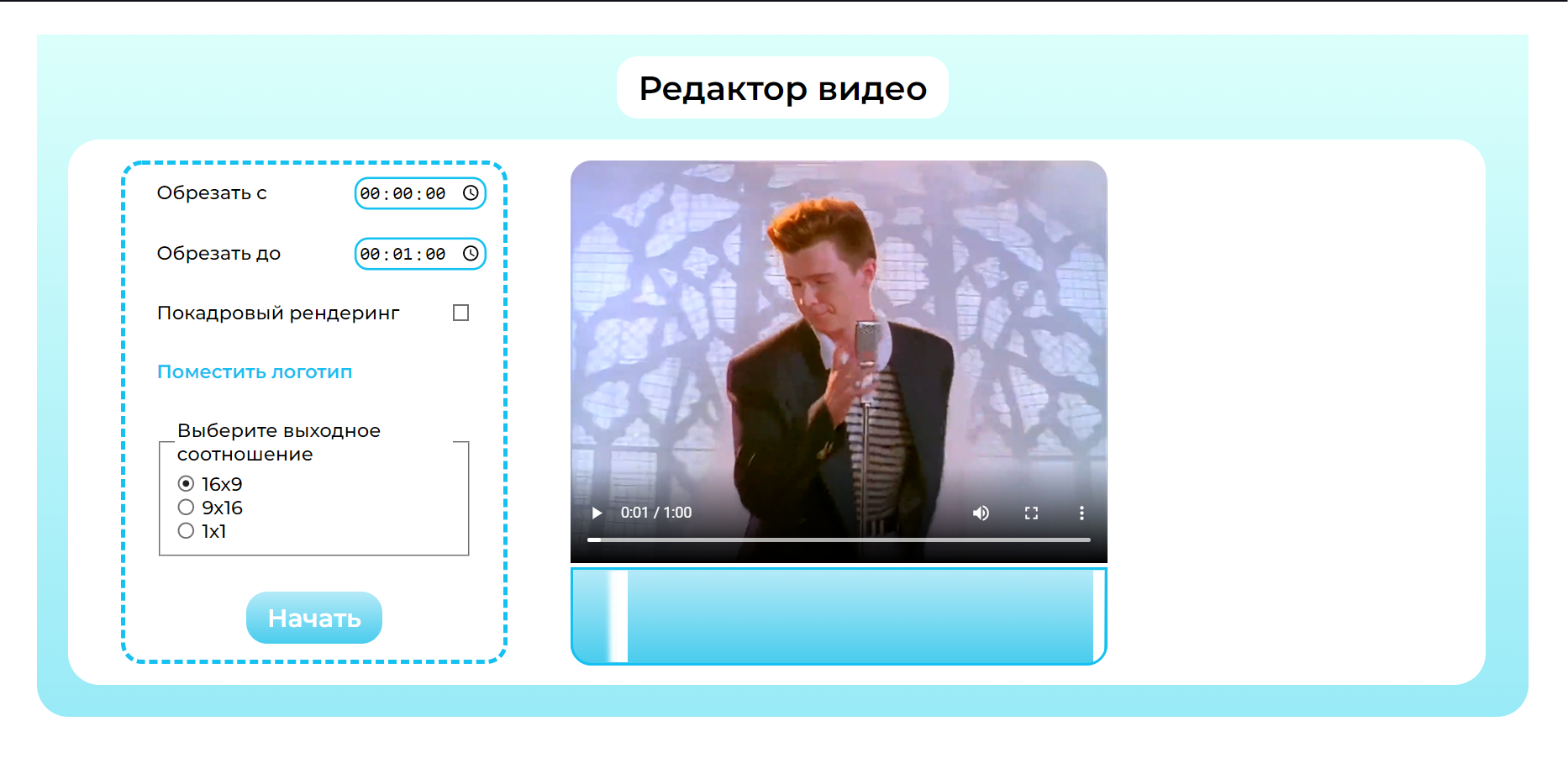
### Вёрстка, стили и внешний вид



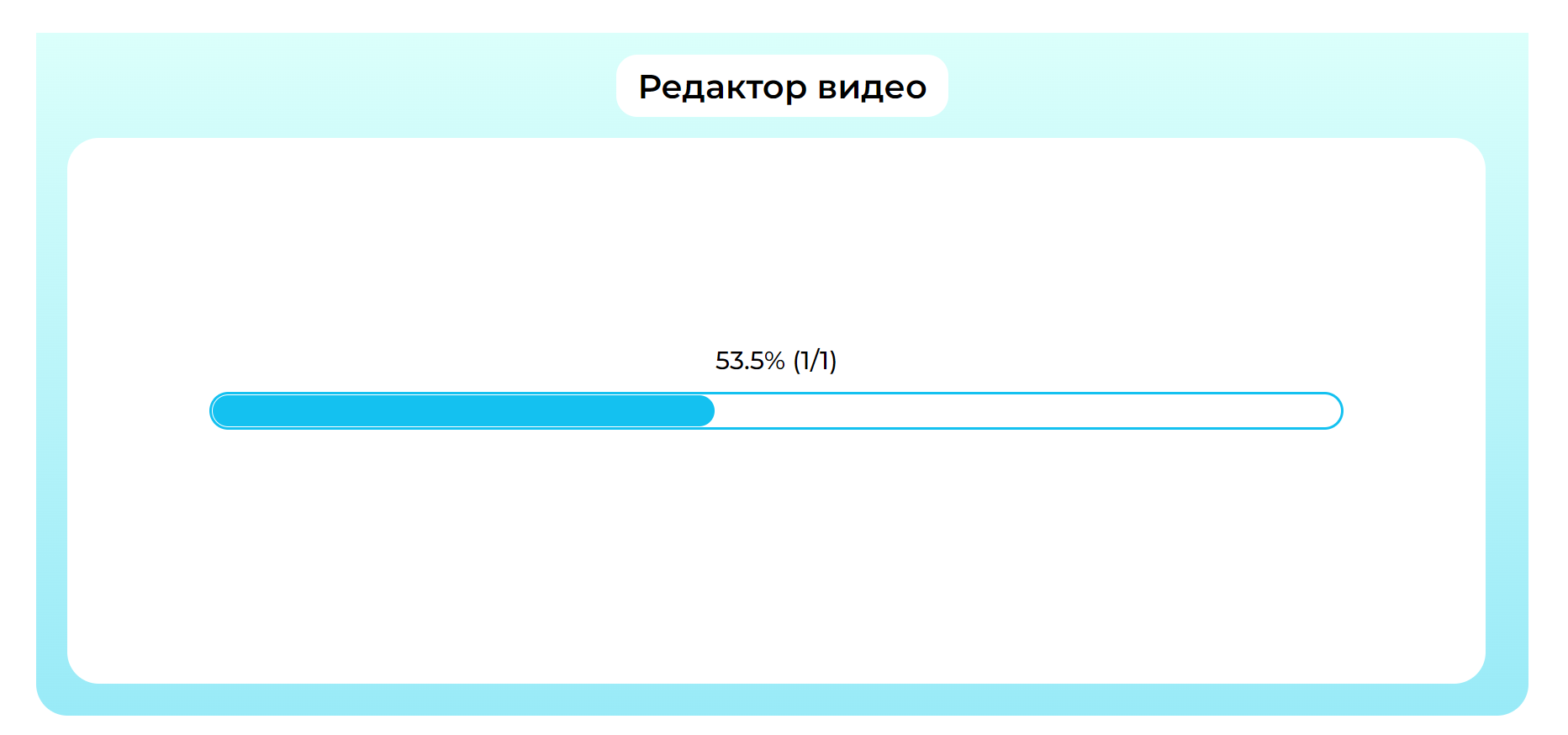
Сразу после перехода на страничку пользователь видит единственный интерактивный элемент – <*input>,* с помощью которого можно загрузить одно видео.

Список доступных форматов:

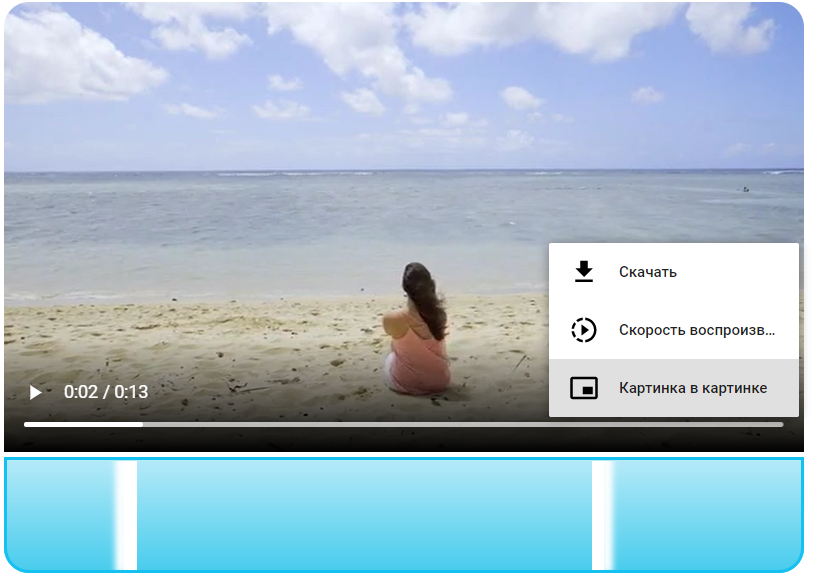




После загрузки видео пользователь может указать отрезок обрезки с помощью ползунков внизу или через *input [type = “time”].* Также доступен покадровый рендеринг – если выбрать его, то обрезка будет точнее (не будет сдвигов дорожки звука относительно видео, артефактов или зависаний кадров), но в таком случае рендеринг будет длительным. Также есть возможность изменить выходное соотношение (сейчас эта функция работает, но некорректно – она меняет соотношение всего видео, не добавляя чёрных полос. Всё же, возможно это будет кому-нибудь полезно, если исходное видео неправильного формата и его нужно исправить). Также можно добавить логотип – он будет в левом верхнем углу видео.



Во время обработки пользователь видит полоску загрузки.

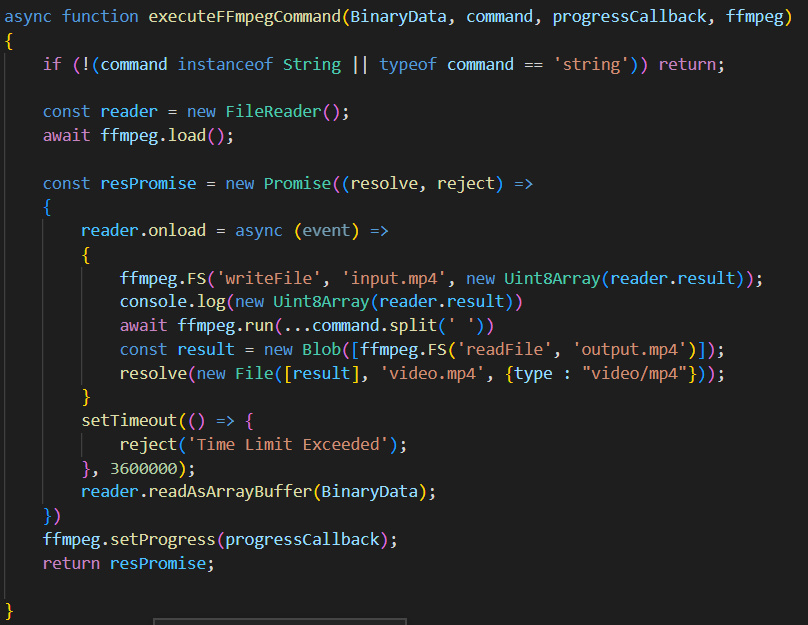


Скачать видео можно с помощью встроенного в браузер выпадающего меню, нажав кнопку «скачать».

### Логика (JavaScript)



Страница слушает событие *“click”* кнопки *«начать»* и считывает введённые пользователем данные. Данные конвертируются и отправляются в функцию *executeFFmpegCommand()*, которая получает на вход команду и видео в бинарном формате и передаёт её *ffmpeg*.



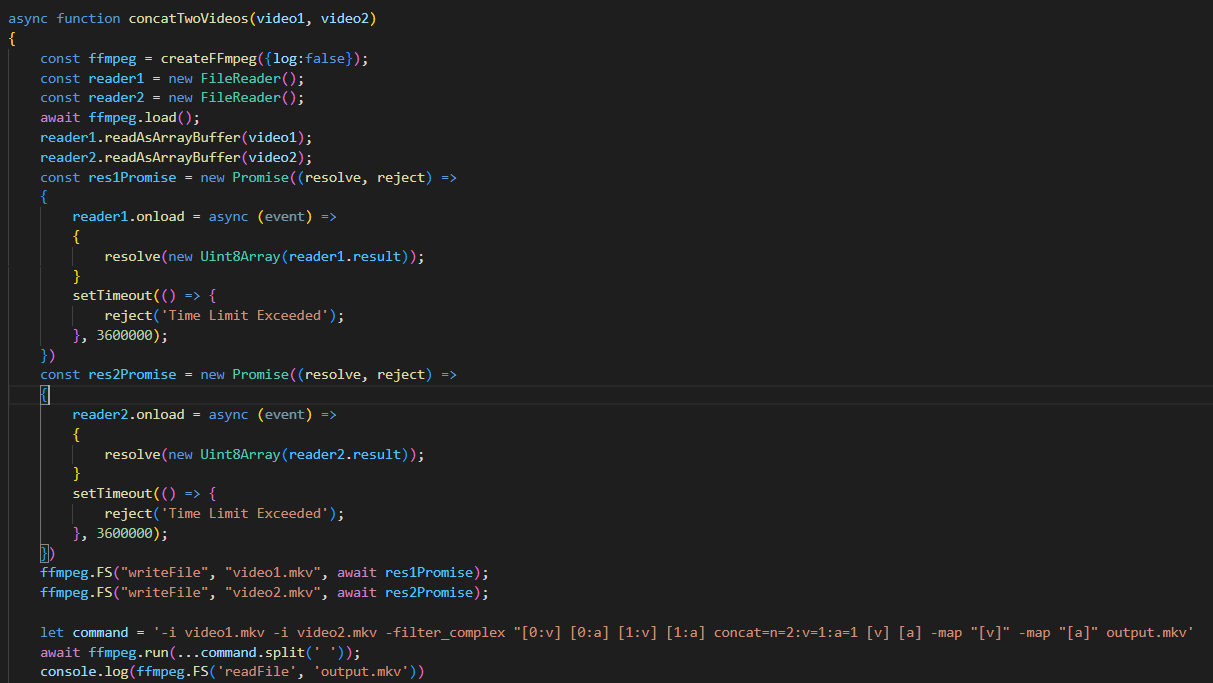
Функция *putLogoOnVideo()* выполнит ещё один проход и добавит логотип в случае, если пользователь указал таковой.



Функция *progressCallback()* получает данные из библиотеки *ffmpeg* и рендерит полоску загрузки на страничке.



Также была идея добавить возможность конкатенации двух видео. Был достигнут промежуточный результат – с помощью функции *concatTwoVideos()* получилось объединить два видео формата *.mkv* (с другими форматами, например *.mp4*, всё сложнее). Но на текущий момент функция сырая и было решено не включать её в текущий продукт.



### Примечания

Чтобы библиотека в принципе работала в браузере, необходимо включить следующие заголовки ответов (иначе SharedArrayBuffer будет заблокирован на уровне браузера):

Cross-Origin-Opener-Policy : same-origin

Cross-Origin-Embedder-Policy : require-corp

### Ссылки

Репозиторий GitHub: <https://github.com/PuriPuri-bot/web-video-cutter.git>

На хостинге Heroku (до ноября 2022 года): <https://web-video-cutter.herokuapp.com/pages/index.html>