**Полное описание проекта «SubSearch»**

*Руководитель проекта: Позднышева Софья Владимировна*

*Разработчики: ученики 10 класса «Н» Рустам Узденов, Иван Петрухин*

**Цель проекта -** создание программного продукта, который позволит в простой и интуитивно-понятной форме любому пользователю вне зависимости от своего уровня подготовки и на любой платформе осуществить поиск контекстных слов субтитрах видеофрагментов (например, размещенных в YouTube) с указанием тайм-кода.

**Актуальность проекта.** Современные обстоятельства обуславливают увеличение потребности у широкого круга лиц в дистанционной работе/обучении. Реализация предлагаемого проекта будет способствовать в более качественном удовлетворении указанной потребности.

Предлагаемый проект **не имеет аналогов в мире**. Есть несколько схожие решения, но существенно ограниченные в сфере своего применения, например:

1. Есть расширение к браузеру (<https://chrome.google.com/webstore/detail/invideo-for-youtube/iacbjlffnpbhgkgknabhkfmlcpdcigab>), но оно активируется с рядом ограничений (только с «десктопного клиента» и при этом только в браузере «Google Chrome»), вследствие чего большинство пользователей не могут пользоваться этим расширением;

2. Среди профессионального сообщества программистов есть ссылка на репозиторий (<https://github.com/linoleum-js/youtube-search>), который теоретически может способствовать в поиске контекстных слов в субтитрах. Однако, это не является единым программным продуктом и не позволяет ею воспользоваться абсолютному большинству пользователей, в силу отсутствия у них необходимых специальных знаний и навыков.

Проект имеет следующие безусловные **преимущества**:

* Кроссплатформенность. Пользователь сможет воспользоваться предлагаемым программным продуктом, используя любое устройство с поддержкой любого интернет-браузера;
* Простота использования. От пользователя требуется наличие простейших навыков использования интернет-браузера, а интерфейс интуитивно-понятный;
* Алгоритм back-end реализуется во front-end (фактически отсутствует back-end), что позволяет предельно снизить нагрузку на сервер программного продукта, а значит недопустить перегрузку сервера и получить бесплатно услуги хостинга;
* Широкая сфера применения: образование, развлечения и т.д.;
* Высокий потенциал расширения функциональных возможностей;
* Бесплатное и безлимитное использование программного продукта.

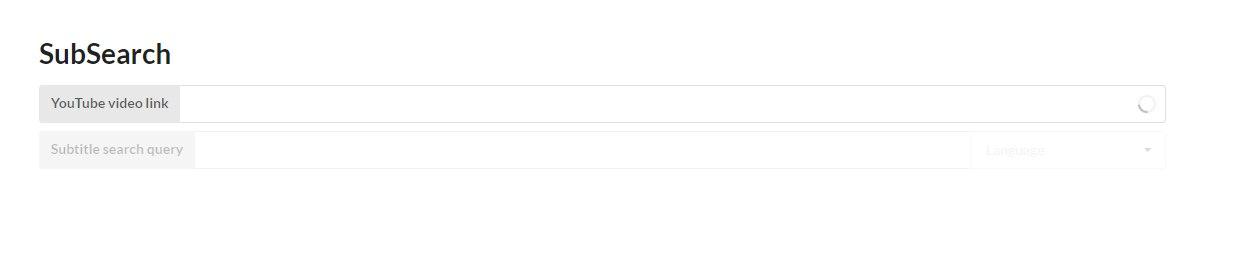
**Описание разработки**

Проект реализуется в форме сайта с нижеуказанными техническими особенностями.

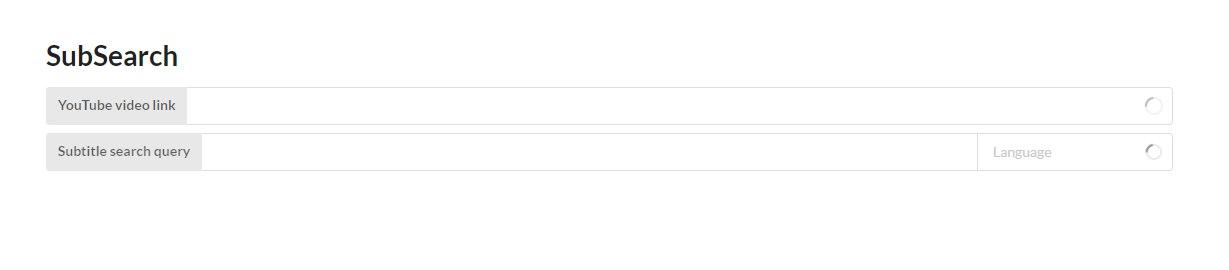
1. **Front-end**

1.1. На главной странице размещается поле для ввода ссылки на видео и поле для ввода поискового слова, совмещенное с полем выбора языка субтитров (см.: *Изображения № 1 - № 3*). Оба поля оснащены индикатором загрузки для отслеживания прогресса поиска.

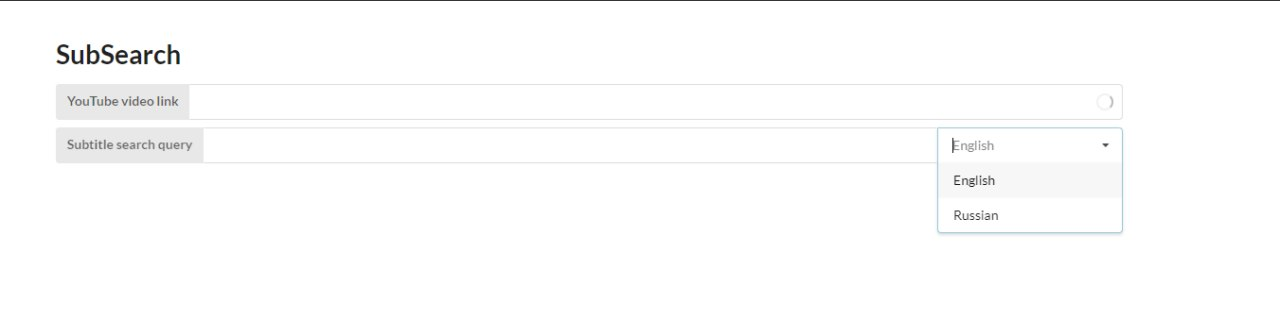
*Изображение № 1*



*Изображение № 2*



*Изображение № 3*



* 1. Порядок ввода и обработки запроса:
     1. После ввода ссылки идет загрузка субтитров на сайт и весь прогресс загрузки отображается в индикаторе *(см.: Изображения № 1)*;
     2. Пользователь выбирает язык субтитров *(см.: Изображения № 2 - 3)*;
     3. Как загрузка субтитров будет закончена, пользователь сможет ввести нужное слово в поисковое поле. Прогресс загрузки отображается уже в другом индикаторе *(см.: Изображения № 2 - 3);*
     4. После поиска будет выведена таблица с двумя столбцами: первый столбец будет с тайм-кодами (гиперссылки на нужное видео и время), а второй столбец - с предложением, где было найдено это слово *(см. Изображение № 4)*;
     5. После поиска будет выведена таблица с двумя столбцами: первый столбец будет с тайм-кодами (гиперссылки на нужное видео и время), а второй столбец - с предложением, где было найдено это слово *(см. Изображение № 4)*;

*Изображение № 4*

Поле ввода

|  |
| --- |
| сто |

Таблица

|  |  |
| --- | --- |
| 1:10 | Сто лет меня не было |
| 11:54 | Просто сделать |
| 17:10 | Стол черный, небо серое |

**2. Back-end**

2.1. Весь код написан на JavaScript и находится во front-end

2.2. Основные компоненты

* Модуль загрузки субтитров из ссылки будет разработан с помощью библиотеки «axios» (<https://www.npmjs.com/package/axios>);
* Парсинг-модуль будет разработан с помощью специальной библиотеки «subtitle» (<https://www.npmjs.com/package/subtitle>).

2.3. Примерное описание работы back-end сайта

2.3.1. Сайт получает ссылку;

2.3.2. Модуль загрузки субтитров делает запрос на http://www.youtube.com/get\_video\_info?video\_id={id видео} (пример ссылки, где в части «id видео» необходимо указать, соответственно, соответствующее значение id видео);

2.3.2. Парсинг-модуль обрабатывает ответ сервера и преобразовает JSON в JavaScript-объект, с которым удобно работать;

2.3.3. Приложение получает ссылки на файлы субтитров и добавляет параметр vtt-формата, чтобы удобно с ним работать;

2.3.4. Выводится пользователю список языков субтитров (сайт получил языки в первом ответе сервера);

2.3.5. Приложение, при выборе языка, загружает субтитры по соответствующей ссылке;

2.3.6. При помощи метода массива «.filter» отфильтровывается массив отрывков субтитров, чтобы остались только те, в которых содержится текст;

2.3.7. Сайт получает поисковое слово и выводит нужную таблицу.