KMedia Player Api
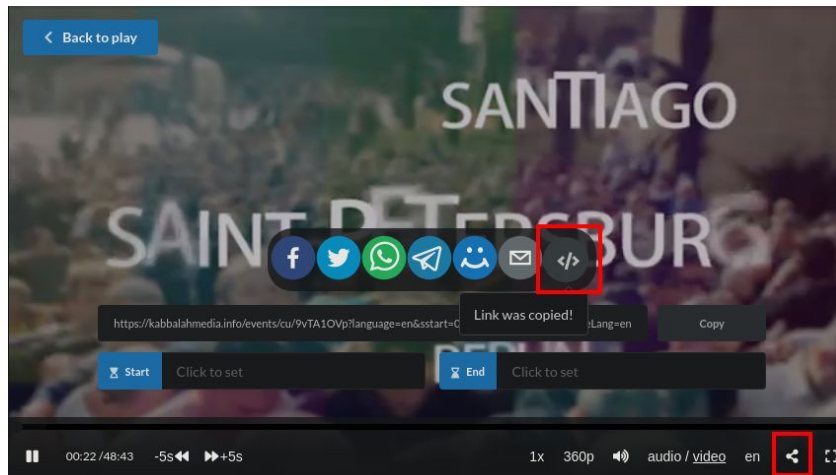
The api enables you to control embedded video from kabbalah media.

Usage:
First share a page with a video from https://kabbalahmedia.info with the </> button:

Put the copied text into your html and add id="player"

```
<html>
        <head>
        </head>
        <body>
                <iframe id="player" width="680" height="420"
src="http://kabbalahmedia.info/en/programs/cu/yFAiAUX8?language=en&sstart=0s&mediaType=video&embed=1&autoPlay=1" frameBorder="0" scrolling="no" allowfullscreen />
        </body>
</html>
```

Include the the api in head:

```
<head>
        <script src="https://kabbalahmedia.info/api/video/kmediaPlayerApi.js"
type="text/javascript"></script>
</head>
```

Add this code to initialize the player api with the iframe id

```
<script>
        function playerCallback(event) {
                console.log(event.data);
        }
        const playerApi = new KMediaPlayerApi('player', playerCallback);
</script>
```

Now you can control the player using the playerApi.

Here is a full working example:

```html
<html>
<head>
        <script src="https://kabbalahmedia.info/api/video/kmediaPlayerApi.js"
type="text/javascript"></script>
</head>
<script>
        function playerCallback(event) {
                console.log(event.data);
        }
        const playerApi = new KMediaPlayerApi('player', playerCallback);
</script>
<body>
        <button onclick="playerApi.play()">play</button>
        <button onclick="playerApi.pause()">pause</button>
        <button onclick="playerApi.stop()">stop</button>
        <button onclick="playerApi.mute()">mute</button>
        <button onclick="playerApi.muteUnmute()">muteUnmute</button>
        <button onclick="playerApi.getInfo()">info</button>
        <div>
                <iframe id="player" width="680" height="420"
src="http://kabbalahmedia.info/en/programs/cu/yFAiAUX8?language=en&sstart=0s&mediaType=video&embed=1&autoPlay=1" frameBorder="0" scrolling="no" allowfullscreen />
        </div>
</body>
</html>
```

Api functions

Function: Constructor	

Params:

playerId	the iframe id

callbackFunc	A callback function. The api will call this function after a command is called to the player api.
	The callback function should have an event parameter.
	
	```function playerCallback(event) {```
	``` console.log(event.data);```
	```}```

Example:
```
const playerApi = new KMediaPlayerApi('player', playerCallback);
```

Description:
Create a new instance of the player api

Function:
removeEventListener

Params:

callbackFunc	A callback function. The api will call this function after a command is called to the player api.
	The callback function should have an event parameter.
	
	```function playerCallback(event) {```
	``` console.log(event.data);```
	```}```

Example:
```
playerApi.removeEventListener(playerCallback);
```

Description:
Unregister player api.
Use this function in your componentWillUnmount for example.

Function:
stop

Params:
No params

Example:
playerApi.stop()

Description:
Stop the video

Function:
play

Params:
No params

Example:
playerApi.play()

Description:
Play the video

Function:
play

Params:
No params

Example:
playerApi.play()

Description:
Play the video

Function:
setVolume

Params:

volume	A value from 0 to 1 for the amount of volume

Example:
playerApi.setVolume(0.3);

Description:
Set the player volume

Function:
getVolume

Params:
No params

Example:
playerApi.getVolume();

Callback result:

event.data.result	The current player volume

Description:
Set the player volume

Function:
getVolume

Params:
No params

Example:
playerApi.getVolume();

Callback result:

event.data.result	The current player volume

Description:
Set the player volume

Function:
getCurrentTime

Params:
No params

Example:
playerApi.getCurrentTime();

Callback result:

event.data.result	The current player time

Description:
Get the player current time

Function:
setCurrentTime

Params:

currentTime	The player time in seconds

Example:
playerApi.setCurrentTime(50);

Description:
Set the player current time

Function:
isFullScreen

Params:
No params

Example:
playerApi.isFullScreen();

Callback result:

event.data.result	Is player in full screen (true or false)

Description:
Get the player full screen status

Function:
isLoading

Params:
No params

Example:
playerApi.isLoading();

Callback result:

event.data.result	Is the player loading (true or false)

Description:
Check if the player is loading

Function:
isPlaying

Params:
No params

Example:
playerApi.isPlaying();

Callback result:

event.data.result	Is the video playing (true or false)

Description:
Check if the video is playing

Function:
isMuted

Params:
No params

Example:
playerApi.isMuted();

Callback result:

event.data.result	Is player volume mute (true or false)

Description:
Check if the player is mute

Function:
getInfo

Params:
No params

Example:
playerApi.getInfo();

Callback result:

event.data.result.result	The player media info

Example for result:

```
▼result:
  ▶ availableLanguages: (10) ["ua", "es", "no", "he", "de", "en", "ru", "fr", "it", "tr"]
  ▶ availableMediaTypes: (2) ["video", "audio"]
  ▶ byQuality: {nHD: "https://cdn.kabbalahmedia.info/yirxDJau.mp4", HD: "https://cdn.kabbalahmedia.info/R7vhmRre.mp4"}
    language: "en"
    mediaType: "video"
    preImageUrl: "/assets/api/thumbnail/yFAiAUX8"
    requestedLanguage: "en"
    requestedMediaType: "video"
    src: "https://cdn.kabbalahmedia.info/yirxDJau.mp4"
  ▶ unit: {id: "yFAiAUX8", content_type: "VIDEO_PROGRAM_CHAPTER", film_date: "2020-06-08", name: "New Life 1251 —  The Demanding Work Culture", duration: 1656, …}
  ▶ __proto__: Object
```

Description:
Get the player media info