

afterUploadVideo

2.2.0

API

```
var editorId = 'synapEditor';
var editorConfig = {};
var html = '';
var editor = new SynapEditor(editorId, editorConfig, html);

editor.setEventListener('afterUploadVideo', function (e) {
});
```

```
var editorId = 'synapEditor';
var editorConfig = {};
var html = '';

function SynapEditorAfterUploadVideo(e) {
}

new SynapEditor(editorId, editorConfig, html);
```

```
var editorId = 'synapEditor';
var editorConfig = {};
var html = '';
var eventListeners = {
  afterUploadVideo: function (e) {
  }
};

new SynapEditor(editorId, editorConfig, html, eventListeners);
```

e

e

```
// release 2.2.0
{
    editor: SynapEditor,
    eventType: 'afterUploadVideo',
    cancelable: false,
    returnValue: null,
    path: '/upload/path/filename.mp4', //
    elementId: 'se-123456-abcdefg-7890' // HTML Element Id (wrapper)
}
// release 2.2.1
{
    editor: SynapEditor,
    eventType: 'afterUploadVideo',
    cancelable: false,
    returnValue: null,
    fileType: 'video', // [2.2.1]
    path: '/upload/path/filename.mp4',
    elementId: 'se-123456-abcdefg-7890'
}
// release 2.10.0
{
    editor: SynapEditor,
    eventType: 'afterUploadVideo',
    cancelable: false,
    returnValue: null,
    fileType: 'video',
    path: '/upload/path/filename.mp4',
    responseData: {}, // [2.10.0] response
    elementId: 'se-123456-abcdefg-7890'
}
// release 2.15.0
{
    editor: SynapEditor,
    eventType: 'afterUploadVideo',
    cancelable: false,
    returnValue: null,
    fileType: 'video',
    path: '/upload/path/filename.mp4',
    responseData: {},
    elementId: 'se-123456-abcdefg-7890',
    error: undefined // [2.15.0]
}
}
```

2.15.0

2.15.0 afterUploadVideo .

afterUploadVideo error Error .

```
editor.setEventListener('afterUploadVideo', function (e) {
    if (e.error && e.error.status === 500) {
        e.editor.execCommand('showDialog', 'alert', ' ');
        var video = e.editor.getAPIModelById(e.elementId);
        video.remove();
    }
});
```