

# EXT: CSS styled Filelist

Ключ расширения: **css\_filelinks**

Авторские права 2005-2006, juraj@sulek.sk, <juraj@sulek.sk>

Это документ публикуется под Open Content License  
доступной на <http://www.opencontent.org/opl.shtml>

Содержимое этого документа относится к TYPO3  
- GNU/GPL CMS/Framework доступны на [www.typo3.com](http://www.typo3.com)

## Содержание

<b>EXT: CSS styled Filelist.....</b>	<b>1</b>	Описание.....	3
<b>Введение.....</b>	<b>1</b>	Дополнительный класс.....	3
Предназначение.....	1	Классы.....	3
<b>Руководство пользователя.....</b>	<b>1</b>	Шаблон.....	4
Установка.....	1	Обработка ссылок.....	7
Быстрый старт.....	2	Загрузить из пути.....	8
<b>Настройка.....</b>	<b>3</b>	<b>Важно.....</b>	<b>8</b>
Список файлов.....	3	<b>Список To-Do.....</b>	<b>9</b>
		<b>Список изменений.....</b>	<b>9</b>

## Введение

### Предназначение

Расширение для добавления ссылок на файлы ("tt\_content.uploads") совместимости с CSS и настроек через typoscript.

#### Функции:

- Определение пользователем дополнительных классов, в зависимости от расширений файлов.
- Улучшена настройка через Typoscript
- Определение шаблона.
- Определяемые пользователем метки
- Подготовка для объединения с DAM (Filelinks DAM usage)

## Руководство пользователя

### Установка

При установке доступны следующие параметры:

**CONFIGURATION:**

(Notice: You may need to clear the cache after configuration of the extension. This is required if the extension adds these settings.)

**Enable features**

**Don't insert \_CSS\_DEFAULT\_STYLE** [dont\_default\_css]  
  
 Default: 0

**Allow the "Read from path" field** [allow\_read\_from\_path]  
  
 Default: 0

**The Filelinks field is DAM.** [default\_dam]  
 (The "CSS Multimedia" extension change the default Filelinks field to DAM. I don't know if this makes all versions from "CSS Multimedia" and it is impossible in some version of Typo3 to check if this field is DAM or not. So please select this option if you have installed "CSS Multimedia" and the Filelinks show no results on frontend you can try to turn this option on. For more informations check the manual.)  
  
 Default: 0

**Path to file icons** [pathtoicons]  
  
 Default:

### Don't insert \_CSS\_DEFAULT\_STYLE

Здесь можно отключить вставку формы тега <style> из CSS по умолчанию для этого расширения прямо в HTML. CSS по умолчанию можно также отключить через этот typoscript:

```
plugin.tx_cssfilelist._CSS_DEFAULT_STYLE >
```

### Allow the "Read from path"

Здесь можно включить или отключить поле "Загрузить из пути".

В это поле можно ввести путь к папе, из которой будут показаны все файлы. После ввода пути в это поле, все вставленное в поле "Файлы" будет проигнорировано. За дополнительной информацией обратитесь к разделу "Загрузить из пути".

### The Filelinks field is DAM

Если установлены расширения DAM и CSS Multimedia, а расширение CSS Filelinks при этом не работает, можно попытаться включить этот флажок. Это происходит из-за того, что CSS Multimedia автоматически изменяет поле Медиа на ссылку на поле DAM, при установленном расширении DAM, а CSS Filelinks по другому обрабатывает это поле. Но в некоторых версиях Typo3 невозможно проверить, является ли это поле ссылкой на DAM или нет (во внешнем интерфейсе выдается старый TCA, но во внутреннем интерфейсе TCA новый, с измененным полем Медиа). Если это поле включить без установки DAM или CSS Multimedia, этот параметр будет проигнорирован и поле Медиа будет обработано как обычно.

### Path to file icons

По умолчанию расширения добавляют в css путь к файлу значков как /typo3/gfx/fileicons/ или /t3lib/gfx/fileicons/. Если его нужно изменить, например на typo3/gfx/fileicons/ (без слеша в начале) можно сделать это здесь. Существует лишь одно требование — по этому пути должен находиться файл "default.gif".

### Быстрый старт:

Это расширение работает с пользовательской функцией:

```
userFunc=tx_cssfilelinks->renderFileLinks
```

Для этой функции требуется некоторый typoscript для корректной работы. В нем содержится неполный путь, вы сами должны его дополнить этим.

```
tt_content.uploads.20.
```

Для всего typoscripts в этом руководстве.

Например, Typoscript:

```
fileList.field=media
```

должен быть

```
tt_content.uploads.20.fileList.field=media
```

# Настройка

## Список файлов

```
fileList{
  field=media
  path=uploads/media/
  path.override.field=select_key
  path.override.listNum=first
  path.override.listNum.splitChar=|
  override.filelist.field=select_key
}
```

Этот typoscript где сохранены файлы и какой путь использовать. Это добавляется по умолчанию и не требует замены при использовании полей для медиа из обычной базы данных.

## Описание

```
description.field=imagecaption
description_ifElementEmpty=###FILENAME###
```

description и description\_ifElementEmpty имеют свойства stdWrap.

описание	
description	Здесь можно сохранять описание для файлов. Возможно использовать свойства stdWrap, например description.field, description.ifEmpty.field description.value(...) ... по умолчанию оно установлено на description.field=imagecaption
description_ifElementEmpty	Определите, что нужно использовать для описания, если поле description для элемента пусто. Например, при таких настройках:  description.field=imagecaption description_ifElementEmpty=###FILENAME###  И таких описаниях: "Описание 1 Описание 2  Описание 4"  третий элемент без описания, для его описания будет использоваться метка ###FILENAME###. (если установлено использование dam для списка файлов, установить то, какое поле использовать для описания фала при пустом поле описания, можно через такую настройку: dam.damDescription=field )

## Дополнительный класс

```
additionalClass{
  image=bmp,gif,ico,jpg,png,tiff
  video=wmv,avi,asf,mpg
  audio=mp3,wav,mid
}
```

Этот класс заменяет маркер **###ADDITIONALCLASS###** (смотрите раздел шаблонов). Можно определить дополнительный класс в зависимости от расширений файлов. Синтаксис довольно прост:

```
additionalClass.название_класса=расширения_файлов_через_запятую
```

### ПРИМЕР:

```
additionalClass.image=bmp,ico
additionalClass.stuff=bmp
```

filename	Additional class
image.bmp	image stuff
image.ico	image
document.doc	

## Классы

```
Classes{
  addFirst=1
  addLast=1
  addOdd=1
  addEven=1
  ext.prefixIfFirstNumber=
}
```

Значения first,last,even и odd будут заменять маркер **###FIRSTLASTODDEVEN###** (смотрите раздел шаблонов).

Классы	
classes.addFirst	Если класс "first" будет добавлено к первому элементу.
classes.addLast	Если класс "last" будет добавлено к последнему элементу.
classes.addOdd	Если класс "odd" будет добавлено к каждому нечетному элементу.
classes.addEven	Если класс "even" будет добавлено к каждому четному элементу.
classes.ext.prefixIfFirstNumber	<p>Если расширение файла начинается с цифры, например *.3df. Вставленные значения <b>###FILEEXT###</b> и <b>###CLASS###</b> не будут настоящими определениями класса, потому что класс не может начинаться с цифры. Здесь можно определить строку, добавляемую как префикс к расширению файла, начинающегося с цифры</p> <p>Например, при такой настройке:  <code>classes.ext.prefixIfFirstNumber=num</code>  и расширении файла  3ds  значение, вставленное в <b>###FILEEXT###</b> и <b>###CLASS###</b> будет <code>num3ds</code>  оставив это значение пустым, будет вставлено лишь расширение файла без префикса.</p>

## Шаблон

```

layout{
  global=<div class="l###LAYOUT###"><span class="filecount">###FILECOUNT###</span>###FILE###</div>
  file=<a href="###URL###" class="###CLASS###" ###TARGET###>###COUNTER###</a> ###TITLE###
  ###FILESIZE###</a>
  fileSize{
    layout=(###SIZE### ##SIZEFORMAT###)
    char=none
    format=auto
    desc=b|kb|mb
    round=2
    decimalPoint=.
  }
}

```

Шаблон	
layout.global	Шаблон для всего расширения
layout.file	Шаблон для одного файла
layout.fileSize.layout	Шаблон для метки <b>###FILESIZE###</b>
layout.fileSize.char	Должны ли символы метки <b>###SIZEFORMAT###</b> переводиться в верхний или нижний регистр. Возможные значения — "upper", "lower", "firstUpper", "firstLower", "none". Введя none, получите оригинальный <b>###SIZEFORMAT###</b> без преобразования регистра.
layout.fileSize.format	Which format should the size have. Possible values are b-bite, kb-kilobite, mb-megabite, auto.
layout.fileSize.round	Количество знаков после десятичной точки.
layout.fileSize.desc	<p>Можно ввести описание для b, kb и mb, по следующим правилам:  <code>layout.fileSize.desc=описание байта   описание килобайта   описание мегабайта</code>  <b>ПРИМЕР</b>  через <code>layout.fileSize.desc=kbite mbite mgbite</code>  получаем такие значения: 12.01 kbite или 10.00 mbite.</p>
layout.hideNotProcessedMarkers	Если установлено 1, то все не обработанные метки удаляются. Например, установленная метка <b>###test###</b> , ни разу не использованная не появится во внешнем интерфейсе, если этот параметр равен 1.
layout.fileSize.decimalPoint	Здесь можно установить знак, использующийся в качестве разделителя целой и дробной части размера файла. По умолчанию установлена ".", что выглядит так "12.34 Kb".

## Свойства stdWrap

layout.global, layout.file и layout.fileSize.layout поддерживают свойства stdWrap, поэтому можно использовать, например, такой код:

```

tt_content.uploads.20.layout.file >
tt_content.uploads.20.layout.file.cObject=COA
tt_content.uploads.20.layout.file.cObject{
  wrap=<div class="###CLASS###">|</div>
  10 = TEXT
  10.wrap=<span>|</span>
  10.value=<a href="###URL###">###TITLE###</a>
  20 < .10
  20.value=###FILESIZE###
  30 < .20
  30.value=###DESCRIPTION###
}

```

## Метки

Только для общего шаблона	
###FILE###	Местоположение файлов
Только для layout.file	
###URL###	Вставьте url файла.
###CLASS###	Эта метка включает ###FILEEXT### ##ADDITIONALCLASS### ##FIRSTLASTODDEVEN###.
###TARGET###	Вставляется свойство linkProc.target
###FILEEXT###	Вставляется расширение файла. Эта метка может использоваться в классе.
###ADDITIONALCLASS###	Вставляется дополнительный класс (смотрите раздел Дополнительный класс).
###FIRSTLASTODDEVEN###	Вставляются значения классов first, last, even и odd. (смотрите раздел классы)
###COUNTER###	Вставляется простой счетчик, для первого файла номер=1
###ICON###	Вставляется изображение со значком или эскизом, в зависимости от параметра linkProc.iconCObject . Установка linkProc.iconCObject.makeThumbs=1 заставляет расширение создавать эскизы изображений вместо значков.
###TITLE###	Вставляется название файла (установив расширение dam usage для списка файлов, возможно определить поле, используемое в качестве названия), если выбрано linkproc.removePrependedNumbers=1, то вместо, например "filename_01.jpg", получится "filename.jpg"  Для удаления расширения файла из названия можно использовать такой typoscript: "title.trimExt=1".
###FILENAME###	Вставляется название файла. Игнорируется свойство linkproc.removePrependedNumbers и использование dam.
###DESCRIPTION###	Вставляется описание из свойств во внутреннем интерфейсе. Используя dam, возможно определить поле tx_dam, из которого будет взято описание. За дополнительной информацией обратитесь к руководству по расширению "Filelink DAM usage".
###FILESIZE###	Вставляется элемент layout.fileSize, если позволено свойство <b>show file sizes</b> . Этот маркер сделан для поддержки функциональности <b>show file sizes</b> . Если это не нужно, можно установить: ...file=... ###TITLE### (###SIZE### ##SIZEFORMAT###)... вместо ...file=... ###TITLE### ##FILESIZE###... это будет работать также .  <div style="border: 1px solid #ccc; padding: 5px; width: fit-content; margin: 10px auto;"> <p><b>Show filesize:</b> Check this, if you wish the file sizes to be displayed. <input checked="" type="checkbox"/></p> </div>
###SIZE###	Вставляется размер файла.
###SIZEFORMAT###	Вставляется формат размерности файла (смотрите layout.fileSize.format).
Для обоих	
###FILECOUNT###	Вставляется счетчик файлов.
###LAYOUT####	Вставляется значение из поля Шаблон настроек ссылок для файлов из внутреннего интерфейса.  <div style="border: 1px solid #ccc; padding: 5px; width: fit-content; margin: 10px auto;"> <p><b>Layout:</b> ? Select a layout. Layout 1 ▾</p> </div>

## Определяемые пользователем метки

```

layout{
    userMarker{
        global{
            MARKER1=TEXT
            MARKER1.value=test
        }
        file{
            ICON2 = USER
            ICON2.userFunc=tx_cssfilelinks->Icons
        }
        dam{
            FILECREATE=date_cr,date_mod
            AUTOR=autor
        }
    }
}

```

Существует три типа определяемых пользователем меток:

- общие
- для файла
- dam

Первые два используют следующий СИНТАКСИС:

```
layout.userMarker.global.USRMARKER=USER  
layout.userMarker.global.USRMARKER.userFunc=someClass->someFunction
```

затем эти маркеры можно использовать таким образом:

```
layout.global=<div class="###LAYOUT###">###USRMARKER###
```

Маркеры для файла имеют тот же синтаксис.

#### DAM userMarker:

Последние (dam) имеют такой СИНТАКСИС:

```
layout.userMarker.dam.USRMARKER = названия полей таблицы tx_dam через запятую
```

Для их использования, нужно установить расширение Filelinks DAM usage (dam\_filelinks). При добавлении нескольких полей будет выбрано последнее не пустое поле.

Внимание: маркеры чувствительны к регистру.

Это расширение добавляет данные в userMarker.global и userMarker.file typoscript, перед обработкой данного typoscript.

Для общих маркеров	
markerData.fileCount	Счетчик файлов
markerData.fileLayout	Поле шаблон из настроек ссылок файлов

Для маркеров файлов	
markerData.fileUrl	Url файла
markerData.fileName	Название файла
markerData.fileTitle	Заголовок файла
markerData.fileSize	Размер файла
markerData.fileExt	Расширение файла
markerData.dam	Если файл получен из базы данных dam, через расширение "Dam filelinks usage", markerData.dam будет содержать ID этого файла, в другом случае онобудет нулем
markerData.fileCount	Счетчик файлов
markerData.fileLayout	Поле шаблон из настроек ссылок файлов

используя такой код:

```
layout.userMarker{  
    file.ICON = USER  
    file.ICON.userFunc=myClass->Icons  
}
```

и изменив его таким образом:

```
layout.userMarker{  
    file.ICON = USER  
    file.ICON.userFunc=myClass->Icons  
    file.ICON.markerData{  
        fileUrl=...  
        fileName=...  
        fileTitle=...  
        fileSize=...  
        fileExt=...  
        dam=...  
        fileCount=...  
        fileLayout=...  
    }  
}
```

будет возможно использовать markerData в собственных сценариях.

## ПРИМЕР:

Для `userMarker.file` (`userMarker.global` имеют такой же синтаксис).

Если нужно создать собственную функцию для создания значков, то можно сделать это так:

```
layout{
  file=<a href="###URL###" class="###CLASS###">###ICON2### ###TITLE### ###FILESIZE###</a>
  userMarker{
    file.ICON2 = USER
    file.ICON2.userFunc=myClass->Icons
  }
}
```

а функция может быть:

```
function Icons($content,$conf){
  return '';
}
```

Для `userMarker.dam`:

Для использования этих маркеров, нужно установить расширение "Filelist DAM usage (dam\_filelist)". Возможно показать, например, Автора из этого файла через такой код:

```
layout{
  file=<a href="###URL###" class="###CLASS###">###AUTOR### ###TITLE### ###FILESIZE###</a>
  userMarker.dam.AUTOR=autor
}
```

где `autor` — это поле в таблице `tx_dam`.

## Обработка ссылок

```
linkProc {
  target = _blank
  jumpurl = {$styles.content.uploads.jumpurl}
  jumpurl.secure = {$styles.content.uploads.jumpurl_secure}

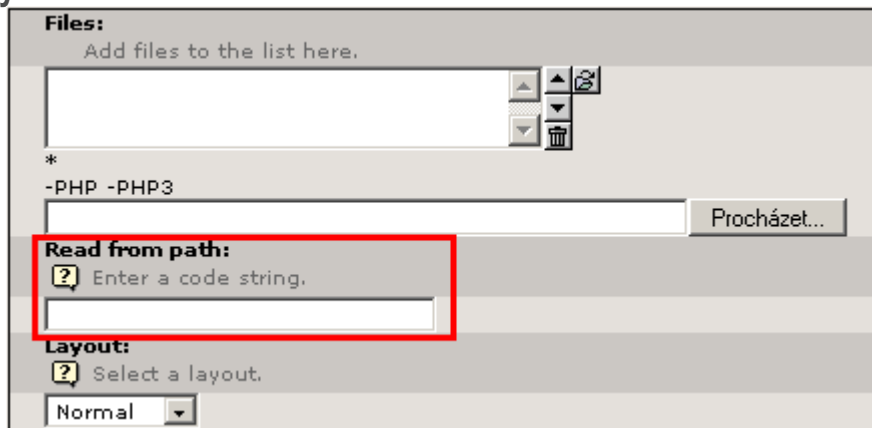
  removePrependedNumbers = 1

  alternativeIconPath=fileadmin/icons/

  iconCObject = IMAGE
  iconCObject.makeThumbs=0
  iconCObject.file.import.data = register : ICON_REL_PATH
  iconCObject.file.width = 150
}
```

Обработка ссылок	
<code>target</code>	Назначение ссылок. Если оставить пустым, назначения не будет установлено.
<code>jumpurl</code>	Установив 1, ссылки будут не такими: "upload/file.jpg", а чем-то вроде этого " <a href="http://www.site.com/index.php?id=1&amp;type=0&amp;jumpurl=uploads%2Ffile.jpg">http://www.site.com/index.php?id=1&amp;type=0&amp;jumpurl=uploads%2Ffile.jpg</a> " и возможно защитить папку "upload", например через <code>_htaccess</code> .
<code>jumpurl.secure</code>	Установив 1, ссылки станут похожими на " <a href="http://www.site.com/index.php?id=1&amp;type=0&amp;jumpurl=uploads%2Ffile.jpg&amp;juSecure=1&amp;locationData=9%3Att_content%3A44&amp;juHash=2469f79328">http://www.site.com/index.php?id=1&amp;type=0&amp;jumpurl=uploads%2Ffile.jpg&amp;juSecure=1&amp;locationData=9%3Att_content%3A44&amp;juHash=2469f79328</a> ", тем самым путь станет еще более запутанным, чем с установкой <code>jumpurl</code>
<code>alternativeIconPath</code>	Обычно, путь поиска значков для этого расширения — <code>t3lib/gfx/fileicons/</code> . Здесь можно определить альтернативный путь поиска значков. Если соответствующих значков не найдено, проверяется <code>default.gif</code> , если и этот файл отсутствует, поиск осуществляется по старому пути ( <code>t3lib/gfx/fileicons/</code> ).
<code>removePrependedNumbers</code>	Установив на 1, получите название для файла <code>file_01.jpg</code> , не <code>file_01.jpg</code> , а <code>file.jpg</code> .
<code>iconCObject</code>	С этим <code>cObject</code> возможно установить, как должен выглядеть маркер <code>###ICON###</code> .
<code>iconCObject.makeThumbs</code>	Установив на 1, получим создание эскизов изображений вместо использования значков. Этот параметр имеет свойства <code>stdWrap</code> .
<code>iconCObject.file.width</code>	Ширина значков

## Загрузить из пути



The screenshot shows a file manager interface with the following elements:

- Files:** A section with the text "Add files to the list here." and a list area.
- \* -PHP -PHP3**: A filter or search term.
- Procházet...**: A button to refresh or search.
- Read from path:** A section with a question mark icon and the text "Enter a code string." This section is highlighted with a red box.
- Layout:** A section with a question mark icon and the text "Select a layout." Below it is a dropdown menu showing "Normal".

В это поле можно ввести путь к каталогу, все файлы из которого нужно показать, это избавляет от необходимости вносить каждый файл отдельно.

Значение может быть разделено "|" на параметры:

- 1: Путь
- 2: Допустимые расширения файлов через запятую (без пробелов); если пусто, используются все файлы.
- 3: Сортировка: name, size, ext, date
- 4: Обратный порядок: установите "r" для обратного порядка сортировки.

ПРИМЕР:

командой

```
"fileadmin/myOrder/|jpg,gif,png|name|r"
```

можно получить все файлы jpg, gif и png и директории fileadmin/myOrder отсортированные в обратном порядке по имени.

## Важно

Используя "Filelinks DAM usage(dam\_filelinks)" с css\_filelinks версией 0.2.0 и выше, версия dam\_filelinks должна быть 0.3.0 и выше. Это нужно по причине того, что начиная с css\_filelinks версии 0.2.0, DAM usage внесена в код расширения и после версии 0.2.0 все наполнение DAM было удалено, поэтому dam\_filelinks должен использовать обработчик. Так, с версией css\_filelinks < 0.2.0 нужно использовать dam\_filelinks < 0.3.0, а с css\_filelinks >= 0.2.0 нужно использовать dam\_filelinks >= 0.3.0.



# Список То-До

Пока ничего.

## Список изменений

### Изменения в `css_filelinks` 0.2.16

- New parameter добавлено to hook "getFileUrl"(due to the new `dam_filelinks` functionality `linkProc.jumpurl.damSecure`).

### Изменения в `css_filelinks` 0.2.15

- `layout.decimalPoint` добавлено

### Изменения в `css_filelinks` 0.2.14

- Bug repaired by `getIcon` hookfunction

### Изменения в `css_filelinks` 0.2.13

- Bug repaired by(`linkProc.iconCObject.makeThumbs=1`)(thx. to Robert Lockerbie for patch)

### Изменения в `css_filelinks` 0.2.12

- `title.trimExt` добавлено.

### Изменения в `css_filelinks` 0.2.11

- "Path to file icons" добавлено

### Изменения в `css_filelinks` 0.2.10

- bug repaired (thx. to Peter Bachmann for hint)

### Изменения в `css_filelinks` 0.2.9

- bug #3796 repaired (thx. to Jan-Claas Dirks)

### Изменения в `css_filelinks` 0.2.8

- `classes.ext.prefixIfFirstNumber` добавлено
- bug repaired by XCLASS definition

### Изменения в `css_filelinks` 0.2.7

- `iconCObject.makeThumbs` has now `stdWrap` properties

### Изменения в `css_filelinks` 0.2.6

- Changes by "Загрузить из пути" typoscript and php to allow all "stdWrap.filelist" parameters

### Изменения в `css_filelinks` 0.2.5

- Because the CSS Multimedia change the media field to DAM reference field if DAM is installed the CSS Filelinks did not work correct, this problem is now solved

### Изменения в `css_filelinks` 0.2.4

- "Загрузить из пути" field добавлено

### Изменения в `css_filelinks` 0.2.3

- Небольшие изменения в руководстве

### Изменения в `css_filelinks` 0.2.2

- css updated: try to find icons on 2 places: `t3lib/gfx/` and `typo3/gfx/`

### Изменения в `css_filelinks` 0.2.1

- bug repaired thx. to Pieter for hint

### Изменения в `css_filelinks` 0.2.0

- all dam stuff is removed from `css_filelinks` and `dam_filelinks` use now hooks

### Изменения в `css_filelinks` 0.1.8

- bug repaired

### Изменения в `css_filelinks` 0.1.7

- try to find the default icon configuration on 2 places: `t3lib/gfx/` and `typo3/gfx/`

### Изменения в `css_filelinks` 0.1.6

- bug repaired by jumpurl.secure (thx. to Herbert Roider)
- alternativelconPath добавлено (thx. to Alex Widschwendter for patch)
- manual changed

#### **Изменения в css\_filelinks 0.1.5**

- manual changed

#### **Изменения в css\_filelinks 0.1.4**

- hooks to every function добавлено for better extending
- ###FILENAME### placeholder добавлено
- description\_ifElementEmpty property добавлено
- manual changed

#### **Изменения в css\_filelinks 0.1.3**

- stdWrap properties for: layout.global, layout.file, layout.filesize.layout

#### **Изменения в css\_filelinks 0.1.2**

- bug repaired (thx. to Daniel Costas)

#### **Изменения в css\_filelinks 0.1.1**

- properties linkProc.target, linkProc.jumpurl, linkProc.jumpurl.secure and linkProc.removePrependedNumbers добавлено
- добавлено marker ###TARGET###
- добавлено marker ###ICON###
- изменения в руководстве пользователя

#### **Изменения в css\_filelinks 0.1.0**

- изменения в руководстве пользователя
- if file don't exist on disk it will not appear on frontend
- icon добавлено

#### **Изменения в css\_filelinks 0.0.11**

- page TSconfig добавлено
- css changes
- default layout changes
- изменения в руководстве пользователя

#### **Изменения в css\_filelinks 0.0.10**

- description.field property добавлено
- изменения в руководстве пользователя

#### **Изменения в css\_filelinks 0.0.9**

- Better Dam integration
- изменения в руководстве пользователя

#### **Изменения в css\_filelinks 0.0.8**

- PHP changes
- better PHP comments

#### **Изменения в css\_filelinks 0.0.7**

- New default CSS
- New default file and global layout
- Edit icon for frontend editing добавлено
- добавлено property layout.hideNotProcessedMarkers
- добавлено layout.userMarkers.dam

#### **Изменения в css\_filelinks 0.0.6**

- добавлено markerData.fileCount and markerData.fileLayout to layout.userMarker.file
- добавлено ###FILEEXT###, ###ADDITIONALCLASS###, ###FIRSTLASTODDEVEN###

- добавлено `classes.addFirst=1`, `classes.addLast=1`, `classes.addOdd=1`, `classes.addEven=1`

- изменения в руководстве пользователя

#### **Изменения в `css_filelinks 0.0.5`**

- изменения в руководстве пользователя

- `userdefined markers`

#### **Изменения в `css_filelinks 0.0.4`**

- изменения в руководстве пользователя

- `bug repair`

#### **Изменения в `css_filelinks 0.0.3`**

- изменения в руководстве пользователя

#### **Изменения в `css_filelinks 0.0.2`**

- добавлено `layout.Filesize.format=auto`

#### **`css_filelinks 0.0.1`**

- первый выпуск