



---

# JaJuMa

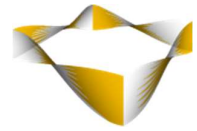
## Hyvä Inline CSS

07/2023



## Table of Contents

1. Introduction .....	- 3 -
2. System Requirements & Installation .....	- 4 -
2.1. Extension Installation via composer .....	- 4 -
2.2. Manual Extension Installation via FTP .....	- 4 -
2.3. Install Required Node Packages.....	- 5 -
3. Extension Scope.....	- 6 -
4. Configuration.....	- 7 -
2.4. Enable.....	- 8 -
2.5. Safe List.....	- 8 -
2.6. Classes Excluded From Hashing.....	- 9 -
2.7. Only Inline CSS On First Page Load.....	- 10 -
2.8. Compatibility Mode.....	- 11 -
2.9. Load styles.css Only After User Interaction .....	- 11 -
2.10. Npx Execution Path .....	- 12 -
2.11. Inline Style Cache Life Time .....	- 12 -
2.12. Clear Inline Style Cache .....	- 13 -
2.13. Enable Debug Mode.....	- 14 -
5. Support .....	- 15 -

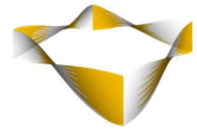


---

## 1. Introduction

[JaJuMa Hyvä Inline CSS](#) for Hyvä Themes & Magento 2 automatically generates per page and adding CSS inline. Allowing to run Magento stores without any CSS file.

Please see below for more details and configuration options provided by this extension.



---

## 2. System Requirements & Installation

!!! This extension requires **Hyvä Theme** !!! !!! Luma frontend is not supported !!!

For installing the extension, follow installation process as with any Magento Extension.

!!! Please note additional NodeJS installation steps described in section 2.3. !!!

### 2.1. Extension Installation via composer

For installing the extension via composer, follow installation process as with any Magento Extension from Magento Marketplace.

See: <https://devdocs.magento.com/extensions/install/>

### 2.2. Manual Extension Installation via FTP

For manual installation by FTP, please follow these steps:

#### Before Installing

1. We recommend you to duplicate your live store on a staging/test site and try installation on your staging/test site before deploying to your live store
2. Backup Magento files and the store database

Please Note: It's very important to backup all themes and extensions in Magento before installation, especially when you are working on a live server.

We strongly recommend you to do not skip this step.

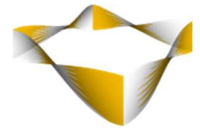
#### Upload the Extension

1. Log into your hosting space via a FTP client (e. g. FileZilla, WinSCP, cuteFtp)
2. Create Folder: <magentoroot>/app/code/Jajuma/HyvaInlineCSS
3. Unzip extension package and upload files into:  
<magentoroot>/app/code/Jajuma/HyvaInlineCSS
4. Enter and run the following commands at the command line:

---

```
php bin/magento setup:upgrade  
php bin/magento setup:static-content:deploy
```

---



---

### 2.3. Install Required Node Package

This extension requires the following NodeJS package to be installed:

- purgess

For installing this package, please go to your Tailwind CSS folder

---

```
cd /app/design/frontend/Acme/default/web/tailwind/
```

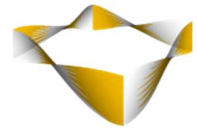
---

and run following command:

---

```
npm install purgess
```

---

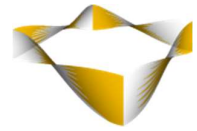


---

### 3. Extension Scope

[JaJuMa Hyvä Inline CSS](#) for Magento 2 provides following options for CSS optimization with Hyvä Themes:

- Generate and add CSS inline per page
- Safe List config for CSS classes in dynamically rendered content
- Compatibility Mode
- Option to inline CSS only on 1<sup>st</sup> page load
- Debug Tool

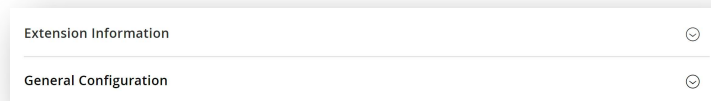


## 4. Configuration

In Magento Backend see

***JaJuMa -> Hyvä Inline CSS -> Configuration***

For JaJuMa Hyvä Inline CSS configuration:

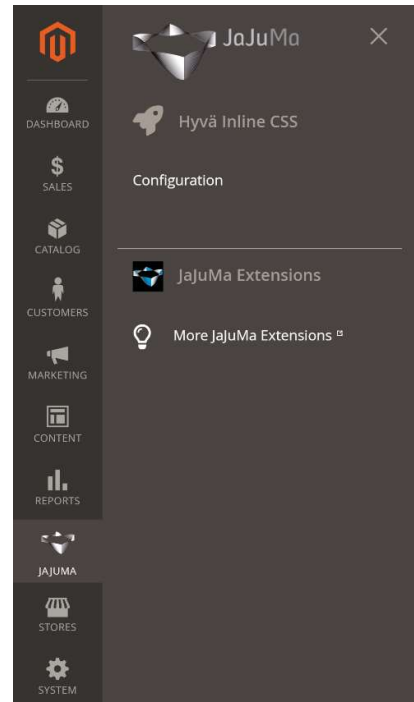


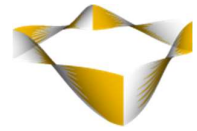
The configurations in

***JaJuMa -> Hyvä Inline CSS -> General Configuration***

provide options as described below.

---





## 2.4. Enable

To enable Hyvä Inline CSS, select from Drop Down:

- Yes → Hyvä Inline CSS is enabled
- No → Hyvä Inline CSS is disabled

### **NOTE:**

When enable this extension, please make sure node packages are installed correctly as described in section [2.3 - Install Required Node Package](#)

**Enable**  
[store view]

Yes

**If Yes:** CSS will be generated and added inline as configured below.  
**If No:** Module is disabled and not doing anything.

**Note:**  
Please make sure to install required node packages before enable this module.

```
cd /app/design/frontend/Acme/default/web/tailwind/  
npm install postcss postcss-cli purgecss
```

## 2.5. Safe List

The extension automatically collects all CSS classes that can be found in page source on page load.

In case some classes / styles are missing, e. g. CSS classes in dynamically rendered content, added by AJAX/GraphQL/REST API, you may configure these classes here as a list separated by space.

**Note:** Instead of configure the Safe List, you can also enable Compatibility Mode. See [2.8 - Compatibility Mode](#) for more details





### Safe List

[\[store view\]](#)

grid-cols-2 mx-6 mx-3

The extension automatically collects all CSS classes that can be found in page source on page load.

In case some classes / styles are missing, e. g. CSS classes in dynamically rendered content, added by AJAX/GraphQL/REST API, you may configure these classes here as a list separated by space.

**Note:**

Please make sure these classes are included in your regular styles.css file.

Instead of configure the Safe List, you may also enable Compatibility Mode (See below).

## 2.6. Classes Excluded From Hashing

The extension uses hash values to identify pages that are using the same collection of styles. To make this work efficiently, classes that make styles unique in all/many pages get excluded from hashing.

By default we exclude CSS classes matching the following patterns:

- Beginning with numeric characters
- More than 3 numeric characters
- Including uppercase characters
- Beginning with the terms *'category'* or *'product'*

In case you have custom classes that make styles on your pages unique, e. g. your SKU if not matching above patterns, classes with random postfix..., you can add them here separated by space.

### Classes Excluded From Hashing

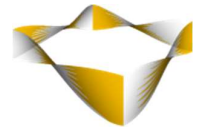
[\[store view\]](#)

/(\s[0-9]\w\*)|(\S\*[0-9]{4,})|(\S\*[A-Z]\S\*)|(\S\*(category|product)(\S\*)/

**Note:**

Please refer to the manual for more information on this config.  
On most sites it will not be needed to change the default setting.  
Default Regex to remove class from hashing:

/(\s[0-9]\w\*)|(\S\*[0-9]{4,})|(\S\*[A-Z]\S\*)|(\S\*(category|product)(\S\*)/



## 2.7. Only Inline CSS On First Page Load

With this option

- **Enabled:**  
CSS will be only added inline on a first page load.  
On subsequent page loads, when user has the regular styles.css file already in his browser cache, page will be loaded normally.
- **Disabled:**  
CSS will be added inline on all page loads.

### Notes:

- With this option enabled, Compatibility Mode is enabled automatically to make sure user will download the regular styles.css on first load.  
By default with low priority or only after user interaction – see option 2.9 - *Load styles.css Only After User Interaction*).
- With this option enabled, Magento will create separate FPC versions for each page with CSS added inline and without inline CSS.  
That means it will take more time to have your cache fully warmed.

Only Inline CSS On First Page Load [store view]

No

**If Yes:** CSS will be only added inline on a first page load. On subsequent page loads, when user has the regular styles.css file already in his browser cache, page will be loaded normally.  
Compatibility Mode is enabled automatically  
Magento will create separate FPC versions for each page with CSS added inline and without inline CSS

**If No:** CSS will be added inline on all page loads.



## 2.8. Compatibility Mode

With this option

- Enabled:  
The regular styles.css file will be added to each page for the browser to download with low priority (or only after user interaction – see next option below).
- Disabled:  
The regular styles.css file will not be added / loaded by browser.

You may enable this option to make 100% sure all styles on your site still work the same as without this module, including any dynamically rendered content.

But still benefit from CSS not being render-blocking anymore.

**Compatibility Mode**  
[store view]

No

**If Yes:** The regular styles.css file will be added to each page for the browser to download with low priority (or only after user interaction – see next option below).  
**If No:** The regular styles.css file will not be added / loaded by browser.

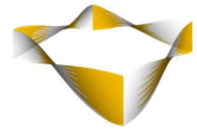
You may enable this option to make 100% sure all styles on your site still work the same as without this module, including any dynamically rendered content.  
But still benefit from CSS not being render-blocking anymore.

## 2.9. Load styles.css Only After User Interaction

This option applies if **Only Inline CSS On First Page Load** or **Compatibility Mode** are enabled.

With this option

- Enabled:  
The regular styles.css file will be loaded only after user interaction without any impact to your performance KPIs.
- Disabled:  
The regular styles.css file will be loaded with low priority.



<b>Load styles.css Only After User Interaction</b> <small>[store view]</small>	<input type="text" value="Yes"/>
This option applies if <b>Only Inline CSS On First Page Load</b> or <b>Compatibility Mode</b> are enabled.	
<b>If Yes:</b> The regular styles.css file will be loaded only after user interaction without any impact to your performance KPIs.	
<b>If No:</b> The regular styles.css file will be loaded with low priority.	
<b>Select "Yes" to avoid any impact on your Page Speed and Core Web Vitals KPIs.</b>	

## 2.10. Npx Execution Path

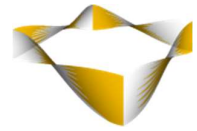
This option allows to define the path to your npx folder/executable.  
Please check with your hosting provider if unclear.

<b>Npx Execution Path</b> <small>[global]</small>	<input type="text" value="npx"/>
Path to npx folder/executable on your server. Default: npx	

## 2.11. Inline Style Cache Lifetime

Once inline styles are generated, they get saved in cache together with a hash value. The cache lifetime defines for how long generated styles are kept in cache. Generally it is recommended to keep the cache lifetime as long as possible and only flush the cache when needed.

<b>Inline Style Cache Lifetime</b> <small>[global]</small>	<input type="text" value="31536000"/>
Input cache lifetime (seconds). Default value: 1 year (31536000 seconds) You may configure a shorter cache lifetime. But normally inline style cache only need to be flushed after updating your Tailwind configs.	



## 2.12. Clear Inline Style Cache

This button will clear all the generated inline style css.

**Clear Inline Style Cache** [website] Clear Cache

This button will clear all the generated inline style css.

### Note:

You can also clear inline style cache from  
**System -> Cache Management -> Flush Inline CSS Cache**

Cache Management
Flush Cache Storage
Flush Magento Cache

<input type="checkbox"/>	Integrations API Configuration	Integrations API configuration file	INTEGRATION_API_CONFIG	ENABLED
<input type="checkbox"/>	Page Cache	Full page caching	FPC	ENABLED
<input type="checkbox"/>	Web Services Configuration	REST and SOAP configurations, generated WSDL file	WEBSERVICE	ENABLED
<input type="checkbox"/>	Translations	Translation files	TRANSLATE	ENABLED
<input type="checkbox"/>	Magewire Data	Magewire cache	MAGEWIRE	ENABLED
<input type="checkbox"/>	Checkout Deployments	Checkout deployment versions - Hyvä Themes	CHECKOUT_DEPLOYMENT	ENABLED

**Additional Cache Management**

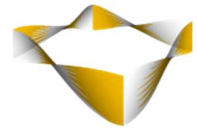
---

Flush Catalog Images Cache
Pregenerated product images files

Flush JavaScript/CSS Cache
Themes JavaScript and CSS files combined to one file

Flush Static Files Cache
Preprocessed view files and static files

Flush Inline CSS Cache
Generated inline css content



## 2.13. Enable Debug Mode

With this option

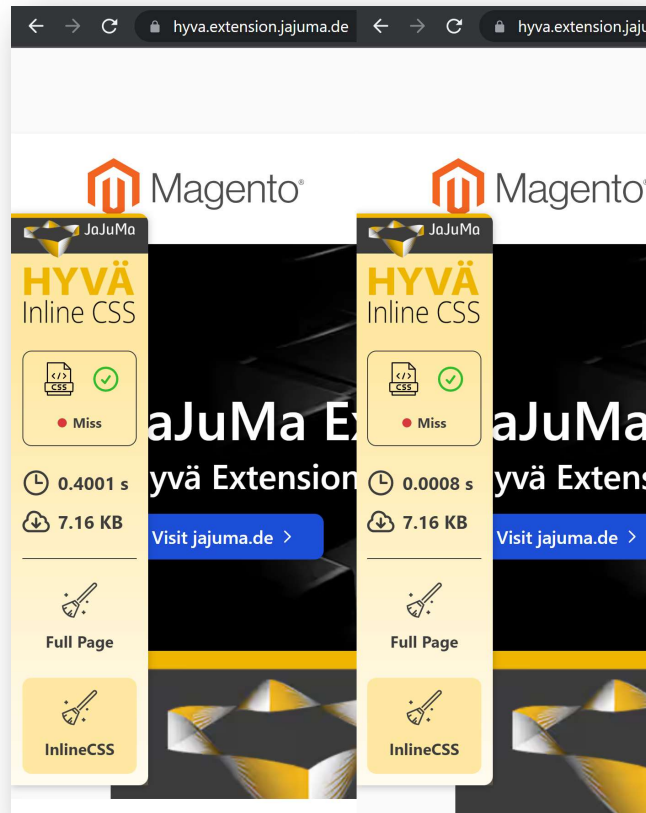
- Enabled:  
A debug popup will be showing in frontend with some info on inline style status, including time spent for processing and gzipped size.  
See example screenshot below.
- Disabled:  
No debug info will be output in frontend.

Enable Debug Mode [store view]

Yes

**If Yes:** A debug popup will be showing in frontend with some info on inline style status, including time spent for processing and gzipped size.

**If No:** No debug info will be output in frontend.





---

## 5. Support

Please feel free to contact JaJuMa support team via [support@jajuma.de](mailto:support@jajuma.de).

In case any additional information is required. We'd be more than happy to assist in setting up the extension.