TechD·vision

Dokumentation Lazyload (v3.x)

Beschreibung

Das Modul Lazyload optimiert das Ladeverhalten von sämtlichen Bildern eines **Magento** Shops in den Bereichen ***Katalog, PDS und CMS*** während des Scrollens und in Viewports, aber auch zusätzlich in den Bereichen ***Warenkorb, Checkout und Kundenbereich*.**

Funktionsmerkmale

Funktion	Beschreibung
Lazy Loading für Katalog- und CMS-Inhalte	Lazy Loading für Bilder in den Bereichen Katalog/CMS/PDS und für Pagebuilder-Elemente.
Konfigurative Lazyload-Strategie	Möglichkeit zwischen nativem HTML-Lazyloading und Javascript-Lazyloading zu wählen.
Verbesserte Ladegeschwindigkeit	Verbesserung der Ladegeschwindigkeit von noch nicht sichtbaren Bildern durch Ersetzen mit einem 1x1 Pixel Dummy Image
Ausschluss von Bildern möglich	[OPTIONAL] Die Möglichkeit, z.B. Logos durch das Setzen der CSS Class **nolazyload** bei Img-Tags in Blocks auszublenden

Contributors

Sonja Baumgartner	s.baumgartner@techdivision.com	Developer
Martin Eisenführer	m.eisenfuehrer@techdivision.com	Maintenance
Ludwig Mair	l.mair@techdivision.com	Documentation
Isabella Patynko	i.patynko@techdivision.com	Documentation
Kenza Yamlahi	k.yamlahii@techdivision.com	Maintenance / Documentation
Patrick Mehringer	p.mehringer@techdivision.com	Maintenance / Documentation

Magento Version Compatibility

Magento Version	Latest Version
Magento >= 2.4.4 Commerce (CE/EE)	composer require techdivision/lazyload ^3.0.0
Magento 2.4.x Opensource (CE) / Commerce (EE)	composer require techdivision/lazyload ^2.1.0
Magento 2.3.x Opensource (CE) / Commerce (EE)	composer require techdivision/lazyload ^1.1.0

PHP Version

Compatible zu PHP Version PHP 7.4 / PHP 8.0 / PHP 8.1

Third Party Libraries

- lazysizes
- lazysizes - fix-ios-sizes
- Intersection Observer

Installation mit Composer

- Um im **TechDivision** Context ein Modul mittels Composer zu installieren, bitte per folgenden Befehl das Repo entsprechend einbinden

```
composer config repositories.repo.met.tdintern.de composer https://repo.met.tdintern.de/
```

Modul Installationsbefehle

Nach Einbindung des **MET-Composer-Repository** folgende Befehle zur Installation ausführen:

```
# add to composer require
composer require techdivision/lazyload ^3.1.0

# run magento setup to activate the module
bin/magento set:up
```

Aktivieren des Moduls

TIP

Das Modul **Country Popup** ist standardmäßig nach der Installation im Magento Backend verfügbar.

TechDivision >> Performance & Images >> Lazyload

Die Modulfunktionalität ist initial aktiviert!

Deinstallation

Folgende Punkte sind bei einer Deinstallation eines Moduls stets zu beachten:

- Gibt es Einträge in der Datenbank, die vor dem Deinstallieren bereinigt werden müssen?
- Sind evtl. Media Files (Images, Videos ect.) vorhanden, die vorab bereinigt werden müssen?
- Gibt es Konfigurationspfade in der Config (DB), die entfernt werden müssen?
- Caches entleeren
- Indexer neu starten wenn notwendig

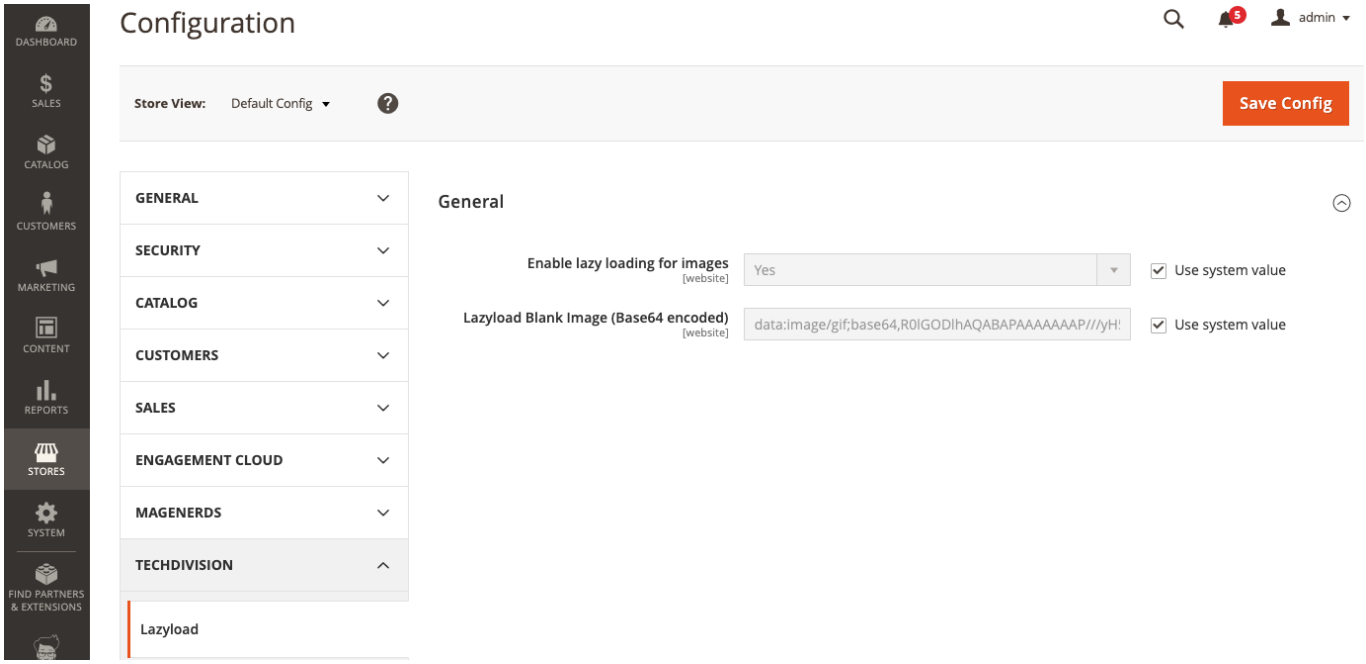
```
# uninstall Module
bin/magento module:uninstall techdivision_lazyload
```

Modul Konfiguration

Navigieren Sie zu **TechDivision >> Performance & Images >> Lazyload**

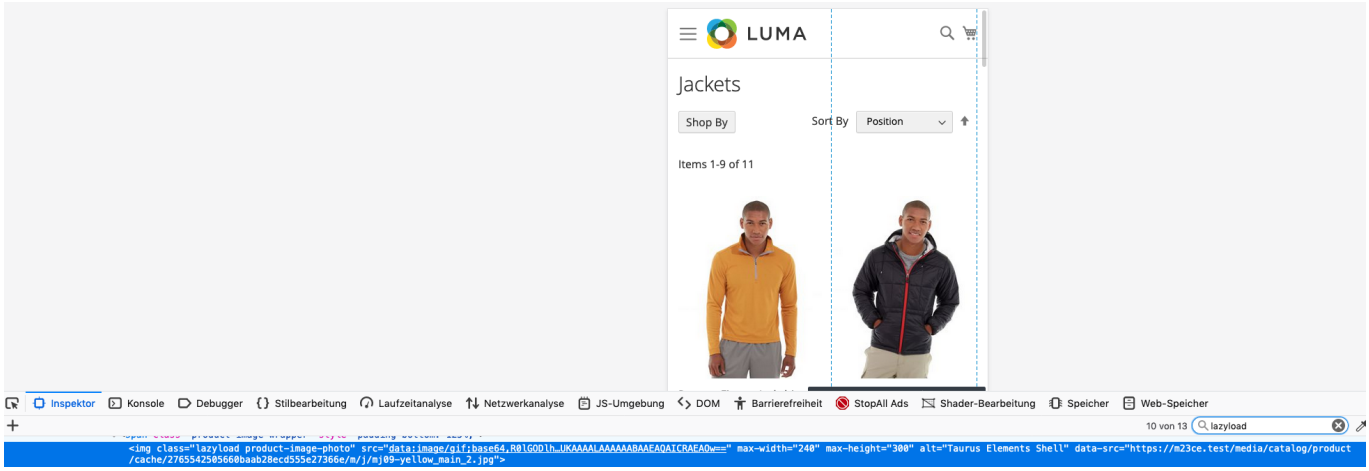
Allgemeine Modul-Optionen

Section	Option	Value	Default	Beschreibung
General	***Enable lazy loading for images***	Yes/No	Yes	Aktivieren des Moduls Lazyload
	Use HTML-Lazyload	Yes/No	No	Aktivieren/Deaktivieren der nativen Lazyload-Funktionalität **CAUTION** Diese Option wird nicht vollumfänglich von allen Browser unterstützt! Hier eine Übersicht für die Unterstützung.
	Lazyload Blank Image (Base64 encoded)		**CAUTION** • Wenn ***Enable lazy loading for images*** auf Yes gesetzt ist, bitte einmal mit btn:[Save Config] das Setting speichern • Nach dem Speichern sollte im Feld ***Lazyload Blank Image*** ein data:image erscheinen, als Platzhalter für fehlende Bilder • Dann bitte noch einmal mit btn:[Save Config] bestätigen



Die Technik des Lazyloadings via dem JS Plugin lazysizes ist wie folgt implementiert:

- Das Modul greift im Magento Prozess erst ganz am Ende des HTML Renderprozess ein um die HTML Img Tags entsprechend vorzubereiten
- Bilder, die sich beim Laden einer Shopseite (CMS/Kategorie, Bluefoot/Bluefoot Teaser und Page Designer), noch nicht im sichbaren Bereich befinden, werden im HTML img src Attribute mit einem 1x1 Pixel Gif ersetzt
- Das entsprechende HTML `<img data-src `Attribute wird mit dem orginalen Bildpfad erweitert und hinzugefügt
- Bilder, die sich beim Laden einer Shopseite (CMS/Kategorie, Bluefoot/Bluefoot Teaser und Page Designer), noch nicht im sichbaren Bereich befinden, werden mit der CSS Class class="lazyload" versehen
- Bilder, die beim Scrollen in den sichbaren Bereich gelangen, werden mit der CSS Class class="lazyloaded" versehen (die bestehende CSS Class lazyload wird durch die CSS Class lazyloaded ersetzt)
- Bilder (1x1 Pixel Gif), die beim Scrollen in den sichbaren Bereich gelangen, werden im HTML Img Tag mit dem im data-src bereitgestellten Bildpfad via JS ersetzt und so nachgeladen



Bedienungsanleitung Modul-Funktionen

Modul-Installation und -Konfiguration

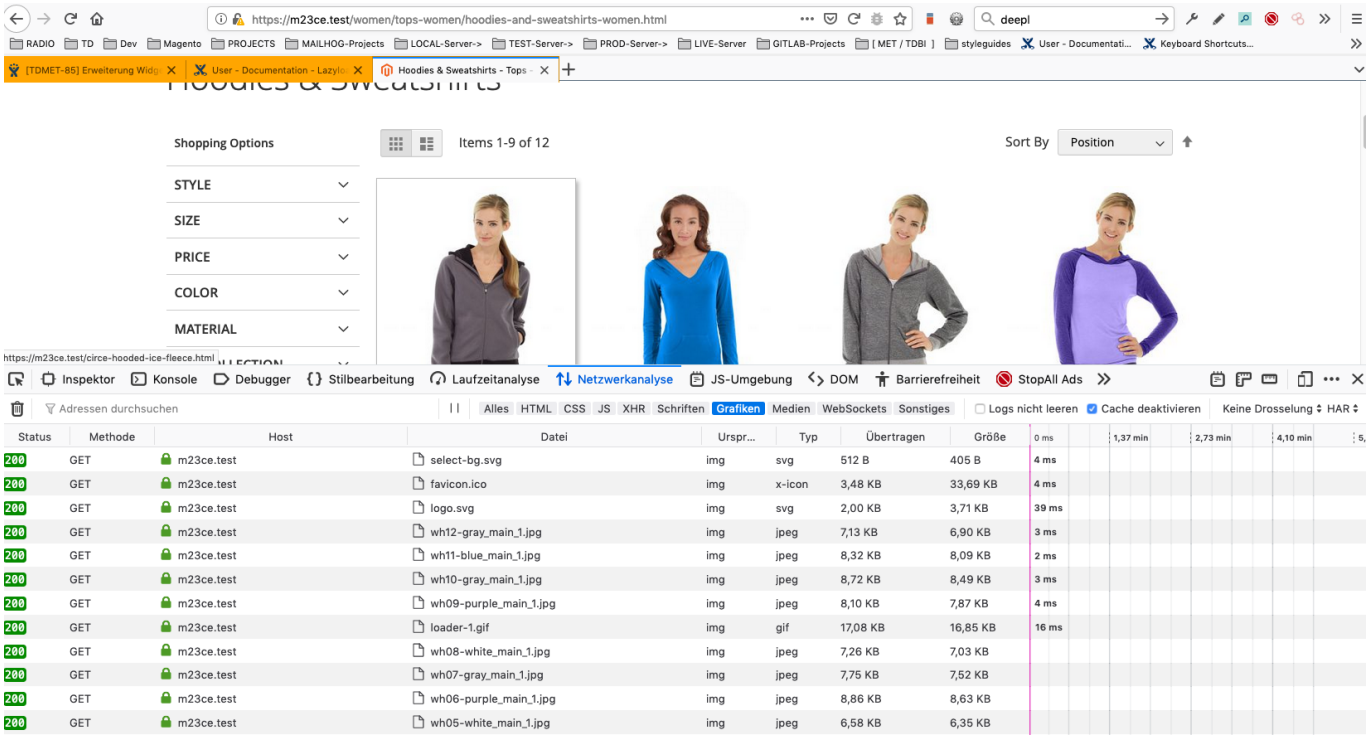
- Installation des Moduls
- Konfigurationseinstellungen des Moduls

Lazyloading und Testen im Browser

- Browser Testing: Firefox
- Browser Testing: Chrome
- Konfigurativer Ausschluss einzelner Bilder
- Lazyload Strategie

Browser Testing: Firefox

- Navigieren Sie zum Browser >> Firefox
- Öffnen Sie bitte Firebug >> Netzwerkanalyse >> Grafiken
- Sichtbaren Bereich neu laden und scrollen
- Dabei sollten dann schrittweise in der **Netzwerkanalyse** die nachgeladenen Bilder angezeigt werden im **Analyse Tool**

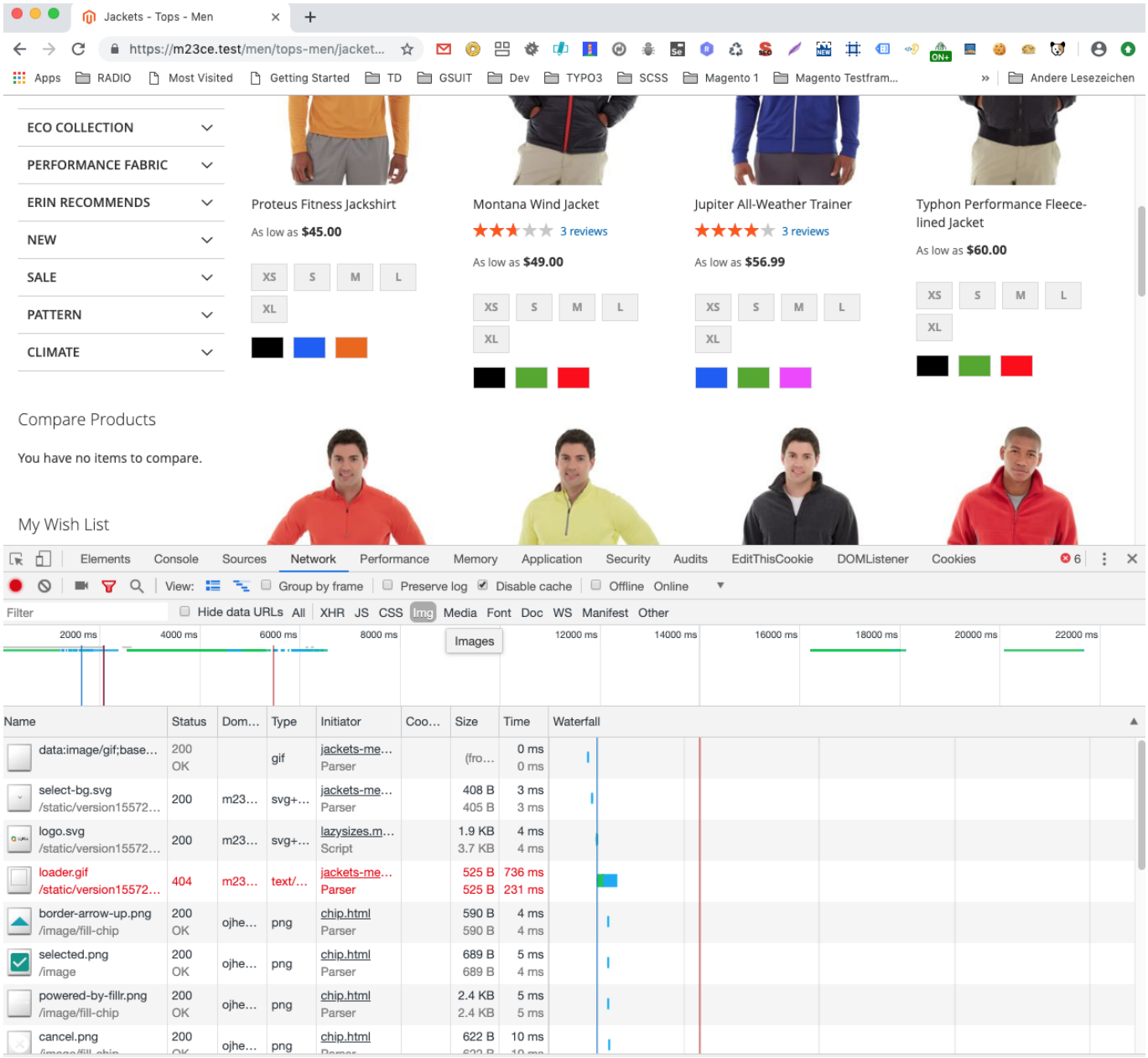


Status	Methode	Host	Datei	Urspr...	Typ	Übertragen	Größe
200	GET	m23ce.test	select-bg.svg	img	svg	512 B	405 B
200	GET	m23ce.test	favicon.ico	img	x-icon	3,48 KB	33,69 KB
200	GET	m23ce.test	logo.svg	img	svg	2,00 KB	3,71 KB
200	GET	m23ce.test	wh12-gray_main_1.jpg	img	jpeg	7,13 KB	6,90 KB
200	GET	m23ce.test	wh11-blue_main_1.jpg	img	jpeg	8,32 KB	8,09 KB
200	GET	m23ce.test	wh10-gray_main_1.jpg	img	jpeg	8,72 KB	8,49 KB
200	GET	m23ce.test	wh09-purple_main_1.jpg	img	jpeg	8,10 KB	7,87 KB
200	GET	m23ce.test	loader-1.gif	img	gif	17,08 KB	16,85 KB
200	GET	m23ce.test	wh08-white_main_1.jpg	img	jpeg	7,26 KB	7,03 KB
200	GET	m23ce.test	wh07-gray_main_1.jpg	img	jpeg	7,75 KB	7,52 KB
200	GET	m23ce.test	wh06-purple_main_1.jpg	img	jpeg	8,86 KB	8,63 KB
200	GET	m23ce.test	wh05-white_main_1.jpg	img	jpeg	6,58 KB	6,35 KB

Browser Testing: Chrome

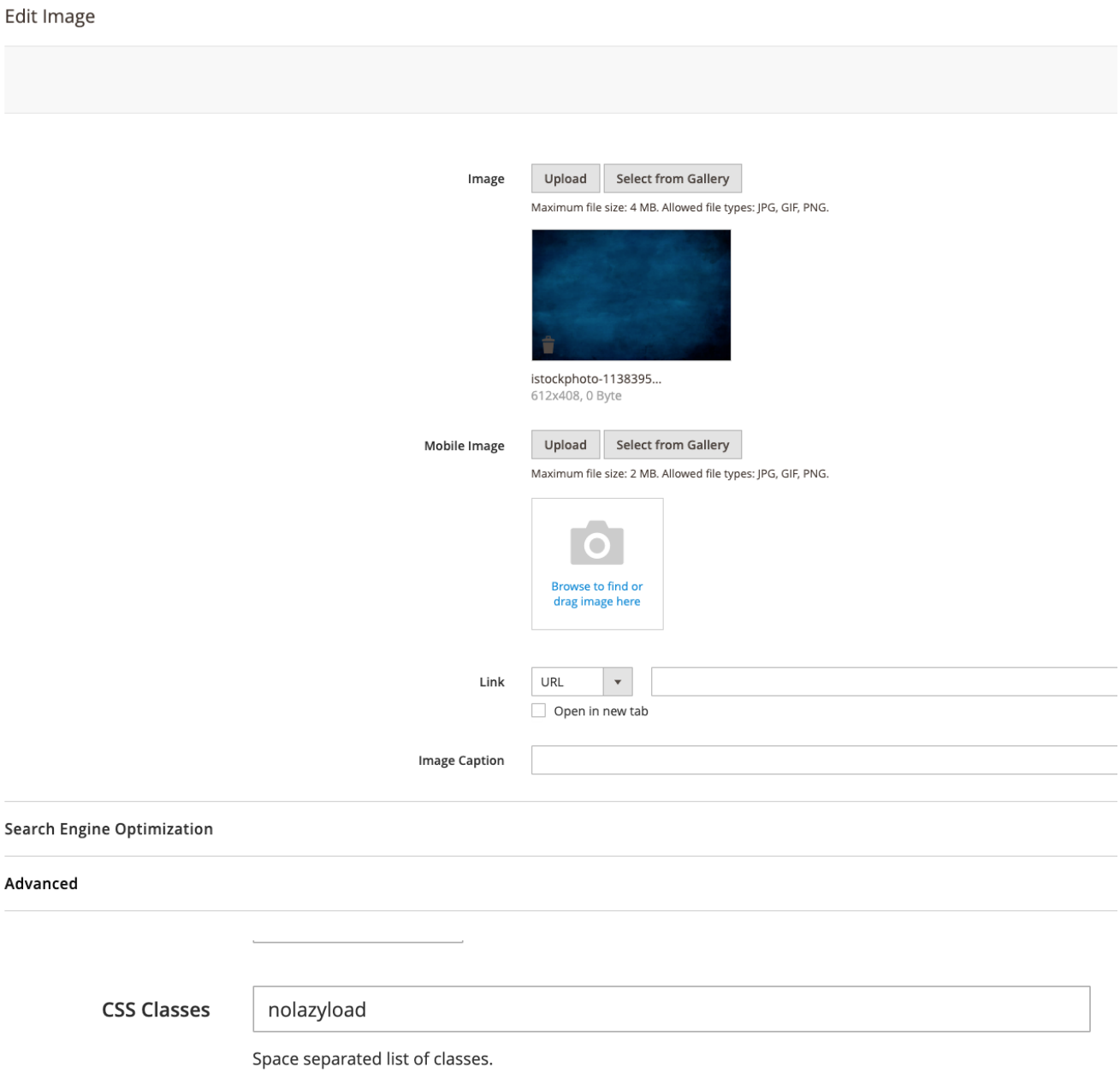
- Navigieren Sie zum Browser >> Chrome

- Öffnen Sie bitte **Developer Tool** und navigieren zu Developer Tool >> Network >> img
- Sichtbaren Bereich neu laden und scrollen
  - Dabei sollten dann schrittweise in der **Netzwerkanalyse** die nachgeladenen Bilder angezeigt werden im **Network Tool**



Konfigurativer Ausschluss einzelner Bilder

1. Navigieren Sie im Magento Admin Panel zur Seitenübersicht Content >> Pages
2. Wählen Sie eine die Seite aus und editieren Sie diese mit PageBuilder
3. Wählen Sie ein Bild auf der Seite aus und setzen Sie bei den Optionen unter Advanced als CSS-Klasse nolazyload



4. Das Element ist nun vom Lazyloading ausgeschlossen

Lazyload Strategie

1. Navigieren Sie im Magento Admin Panel zu den Moduleinstellungen TechDivision >> Performance & Images >> Lazyload
2. Wählen Sie die gewünschte Einstellung der Option Use HTML-Lazyload

> **NOTE** Seit Version 2.2.0/3.1.0 kann die Lazy-Load-Strategie konfigurativ eingestellt werden.

3. Das Lazyloading wird wie folgt umgesetzt und der HTML-Dom gerendert:

HTML Attribut	Javascript

<pre>

  <img class="product-image-photo"
  src="https://lazy/99.jpg"
  loading="lazy"
  width="240" height="300"
  alt="Simple Product 99">

</pre>
	<pre>
	
	  <img class="product-img lazyloaded"
	  src="https://lazy.test/99.jpg"
	  width="240" height="300"
	  alt="Simple Product 99"
	  data-src="https://lazy/99.jpg">
	
	</pre>

Mit der HTML-Strategie wird das HTML-Attribut loading gesetzt. Das Nachladen übernimmt dann der Browser selbst. Diese Option wird nicht vollumfänglich von allen Browser unterstützt! Hier eine Übersicht für die Unterstützung. Der Vorteil dieser Strategie ist eine Performance-Verbesserung, da weniger Javascript-Dateien geladen und ausgeführt werden.	Mit der Javascript-Strategie wird die data-src gesetzt. Nachdem der DOM geladen wurde, setzt das Javascript die src sobald das Bild im Sichtbereich des Browsers ist. Des Weiteren wird noch die CSS-Klasse lazyloaded gesetzt. Der Vorteil dieser Strategie ist die Kompatibilität zu allen Browsern.

Referenzen

Hilfreiche Links zu Tutorials, Manuals und allgemeinen Infos

Third Party Libraries

- lazysizes
- lazysizes - fix-ios-sizes
- Intersection Observer

Allgemeines

- Lazy loading
- Lazy loading via attribute for images & iframes (caniuse)

Glossar

Begriff	Beschreibung
Lazy Loading	Bezeichnet ein Entwurfsmuster, bei welchem Datenobjekte Werte oder andere abhängige Objekte bereitstellen, diese aber erst bei einer konkreten Anfrage aus der Datenquelle holen
View Port / Above the fold	Bezeichnet den, ohne Scrollen und Swipen sichtbaren Bereich in einem Browserfenster