

Tutorials & FAQs

In diesem Bereich der Dokumentation möchten wir Dir einige Tutorials an die Hand geben, mit denen die Betreuung Deines Shopware Shops noch einfacher wird. Du findest hier beispielsweise Informationen zum Debuggen von Fehlermeldungen in Shopware, hilfreiche Tricks für die Analyse in der Datenbank aber auch alles Wissenswerte zur DSGVO.

Zudem beschäftigen wir uns mit den FAQ aus der Shopware Community, um die häufigsten Fragen zu beleuchten und Tipps zu geben.

Änderungen am Template vornehmen

Ziel

Ziel dieses Tutorials soll es sein anhand von kleinen Beispielen die Anpassung des Templates zu beschreiben und das Vorgehen aufzuzeigen. In den ersten Beispielen werden die Schritte der Durchführung sehr genau beschrieben, in den weiteren verzichten wir darauf, da die Vorgehensweise stets dem gleichen Muster folgt.

Allgemeine Vorgehensweise

Als Vorgehensweise legen wir überall die gleichen Schritte zu Grunde:

- Definition der Anpassung
- Suchen der Templatedatei, die für den gesuchten Bereich zuständig ist
- Mit der Anpassung im eigenen Template ableiten
- Testen

Sowohl das Bare-Theme als auch das Responsive Theme sind Standard Templates, die nie geändert werden sollten, da Änderungen bei einem Update überschrieben werden. Daher muss immer korrekt in Deinen eigenen Theme abgeleitet werden.

Falls noch nicht vorhanden, erstelle Dir im Backend im [Theme Manager](#) Dein eigenes Theme, welches vom Responsive Theme ableitet. Shopware legt Dir dann im /themes Ordner die Ordnerstruktur für Dein Theme an, befüllt dies aber nicht mit Dateien, da diese nach der Vererbungslogik aus den Standard-Templates geholt werden. (mehr zur Vererbung findest Du [hier](#))

Größentabelle auf der Detailseite

Ziel soll sein, eine Verlinkung zur Größentabelle auf der Detailseite anzuzeigen, diese soll sich direkt unterhalb der Artikelnummer und den Freitextfeldern 1 und 2 befinden, sofern der Artikel mit dem entsprechenden Freitextfeld versehen ist.

Artikelattribut erstellen / pflegen

Da die Verlinkung nur bei sinnigen Artikel angezeigt werden soll, macht es Sinn, dies über [Freitextfelder](#) zu regeln. Erstelle also ein neues Freitextfeld für die Artikel, im Beispiel nutzen wir hier "attr4":

The screenshot shows two sections of a configuration interface:

- Datenbank Einstellungen:** A text input field for 'Spaltenname' containing 'attr4', a dropdown menu for 'Spaltentyp' set to 'Checkbox - INT(1)', and a red button labeled 'Datenbank Spalte zurücksetzen'.
- Anzeige Einstellungen:** A text input field for 'Label' containing 'Größentabelle anzeigen', a text input field for 'Support Text' containing 'Soll die Größentabelle auf der Detailseite angezeigt werden?', an empty text input field for 'Hilfe Text', a dropdown menu for 'Position' set to '0', a checked checkbox for 'Im Backend anzeigen', and an unchecked checkbox for 'Übersetzbar'.

Das Freitextfeld ist nun im Backend verfügbar und kann bearbeitet werden.

Templateanpassung

Wie finde ich die Stelle, die ich anpassen muss?

Um herauszufinden, wo sich diese Änderung abspielen muss, öffne das Theme-Verzeichnis des Bare Themes. Da sich die hier gewünschte Änderung auf der Detailseite abspielt, navigiere in den Ordner `/themes/Frontend/Bare/frontend/detail`. Hier findest Du alle Template-Dateien, die für die Detailseite verantwortlich sind.

Da unsere Änderung unterhalb der Artikelnummer stehen soll, brauchen wir also den Code, der die Artikelnummer bereitstellt. Du kannst hierfür z.B. mittels Entwicklungsumgebung oder auch einem Editor wie "Notepad++" die Dateien nach Stichwörtern durchsuchen. Suche daher nun in allen Dateien des Pfades nach "ordernummer", da die "ordernummer" in mehreren Dateien Verwendung findet, musst Du also durch die Ergebnisse navigieren und schauen, welche Datei die gewünschte Aufgabe erfüllt. In diesem Fall ist das die **index.tpl**, da hier die Blöcke zu Artikelnummer und den Attributfeldern direkt untereinander stehen. Der Block für die Artikelnummer heißt hier: **frontend_detail_data_ordernummer**.

Wir wissen also nun, dass die *index.tpl* die Datei ist, von der wir ableiten müssen und der Block `frontend_detail_data_ordernummer` heißt.

Änderung korrekt ableiten

Erstelle nun in **/themes/Frontend/DEINTHEME/frontend/detail** eine Datei namens **index.tpl**.

Zu aller erst wird definiert, dass diese Datei vom Original ableitet, die erste Zeile lautet also:

```
{extends file="parent:frontend/detail/index.tpl"}
```

Nun erweitern wir den Block. Da die Größentabelle unter der Artikelnummer stehen soll, muss der vorherige Inhalt über ein **{\$smarty.block.parent}** geladen werden ([Erklärung](#)). Der erweiterte Block sieht dann so aus:

```
{* Append Article SKU with link to size chart *}
{block name='frontend_detail_data_ordernumber'}
  {$smarty.block.parent}
  {if $$Article.attr4 == true}
    <li class="base-info--entry entry--sku">
      {* Size chart - Label *}
      {block name='frontend_detail_data_sizechart_label'}
        <strong class="entry--label">
          {s name="SizeChartLabel" namespace="frontend/detail/data"}{/s}
        </strong>
      {/block}
      {* Size chart - Link *}
      {block name='frontend_detail_data_sizechart_link'}
        <strong class="entry--content">
          {s name="SizeChartLink" namespace="frontend/detail/data"}{/s}
        </strong>
      {/block}
    </li>
  {/if}
{/block}
```

Sobald das Theme im Shop neu kompiliert wird, wird der neue Code nun kompiliert und somit aktiviert.

Größe:

In den Warenkorb

 Vergleichen  Merken  Bewerten

Artikel-Nr.: SW10151

Größenberatung? [Größentabelle](#)

Textbausteine füllen

Die Textbausteine "SizeChartLabel" und "SizeChartLink" sind aktuell noch leer und müssen noch über die [Textbaustein-Verwaltung](#) gefüllt werden. Öffne dazu die Textbausteinverwaltung und suche die Textbausteine anhand des Namens oder alternativ unter dem vergebenen Namespace "*frontend/detail/data*". Als Beispiel füllen wir das Label mit "Größenberatung?" und den Link mit folgendem

HTML-Code:

```
<a href="/groessentabelle" title="Größentabelle" target="_blank">Größentabelle</a>
```

Die Verlinkung zu /groessentabelle setzt an dieser Stelle natürlich voraus, dass eine solche Seite existiert, diese kann beispielsweise über die [Shopseiten](#) erstellt werden.

Weitere Beschreibung unterhalb des Listings

In machen Fällen kann es sinnvoll sein, eine weitere Beschreibung unterhalb des Kategorielistings anzuzeigen. Ziel dieses Tutorials ist es also, eine Textbox ähnlich der Kategoriebeschreibung auch unterhalb des Listings anzuzeigen, sofern die Kategorie über diesen Text verfügt.

Artikelattribut erstellen / pflegen

Da die Beschreibungsbox nur dann angezeigt werden soll, wenn der Text auch vorhanden ist, macht es Sinn, dies über [Freitextfelder](#) zu regeln.

Erstelle also ein neues Freitextfeld vom Typ "Größerer Text" oder "HTML" für die Kategorien, im Beispiel nutzen wir hier "attribute1":

Das Freitextfeld ist nun im Backend verfügbar und kann bearbeitet werden.

Templateanpassung

Wie finde ich die Stelle, die ich anpassen muss?

Um herauszufinden, wo sich diese Änderung abspielen muss, öffne das Theme-Verzeichnis des Bare Themes. Da sich die hier gewünschte Änderung im Listing abspielt, navigiere in den Ordner /themes/Frontend/Bare/frontend/listing, hier findest Du alle Template-Dateien, die für das Listing verantwortlich sind.

Da wir in diesem Fall nicht wissen, wo wir ansetzen müssen, schauen wir zunächst in die index.tpl, die für jede Komponente des Themes das "Grundgerüst" darstellt und finden dort das inkludierte Listing und darunter die Tagcloud.

Wir wissen also nun, dass die *index.tpl* die Datei ist, von der wir ableiten müssen und der **Block frontend_listing_index_listing** heißt.

Variablen herausfinden

Um an das Attribut zu kommen, muss die Variable korrekt sein. Du kannst Dir zum Finden der Variable das Standard Theme zu Nutze machen:

Schauen zuerst, mit welchen Variablen das Theme an der Stelle arbeitet. Im Listing ist das zum Beispiel **\$\$sCategoryContent**. Du kannst Dir in Deiner abgeleiteten Datei nun mittels

```
<pre>{$sCategoryContent | print_r}</pre>
```

alle Variablen aus dem Array "sCategoryContent" ausgeben lassen und Dir hier einfach Deine gesuchte Variable suchen.

Änderung korrekt ableiten

Erstelle nun in /themes/Frontend/DEINTHEME/frontend/listing eine Datei namens index.tpl.

Zu aller erst wird definiert, dass diese Datei vom Original ableitet, die erste Zeile lautet also:

```
{extends file="parent:frontend/listing/index.tpl" }
```

Der Block "frontend_listing_index_listing" wird dann wieder abgeleitet und wir fügen das **{\$smarty.block.parent}** am Anfang hinzu, sodass wir den Inhalt danach ins Template geben:

```
{* Listing *}
{* Listing um Beschreibung erweitern *}
{block name="frontend_listing_index_listing"}
{$smarty.block.parent}
  {if $$sCategoryContent.attribute.attributel}
  {$sCategoryContent.attribute.attributel}
  {/if}
{/block}
```

Da hier nur das Attribut selbst ausgegeben wird, sieht das natürlich noch nicht gut aus:



>Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet citta kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet. Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet citta kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet. Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet citta kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet.

Also gehen wir hin und holen uns das Styling der Kategoriebeschreibung, damit die untere Box gleich der oberen aussieht. Wie im Abschnitt "Wie finde ich die Stelle, die ich anpassen muss?" beschrieben, suchen wir also nun die Kategoriebeschreibung, die sich in der text.tpl befindet und adaptieren das Schema auf unsere Beschreibung:

```
{block name="frontend_listing_index_listing"}
{$smarty.block.parent}
  <div class="hero-unit category--teaser panel has--border is--rounded">
```

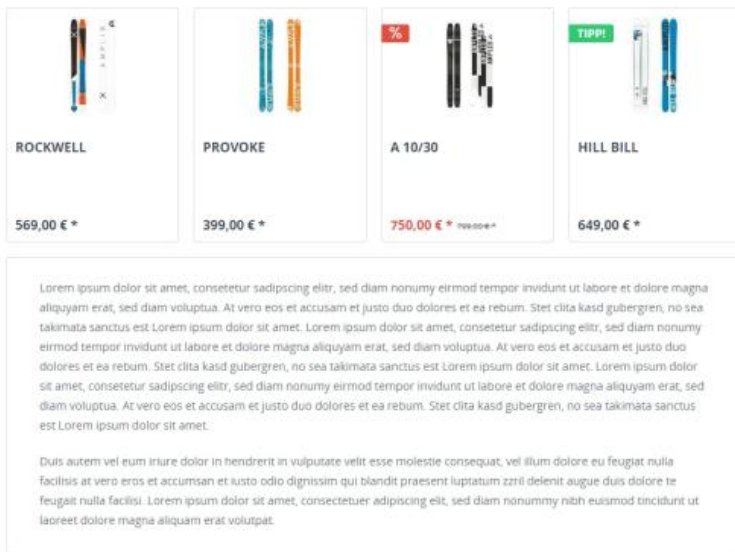
```

<div class="hero--text panel--body is--wide">
  {if $$sCategoryContent.attribute.attribute1}
    <div class="teaser--text-long">
      {$sCategoryContent.attribute.attribute1}
    </div>
    <div class="teaser--text-short is--hidden">
      {$sCategoryContent.attribute.attribute1|strip_tags|truncate:200}
      <a href="#" title="{s namespace="frontend/listing/listing" name="Listing
        {s namespace="frontend/listing/listing" name="ListingActionsOpenOffCan
      </a>
    </div>
    <div class="teaser--text-offcanvas is--hidden">
      <a href="#" title="{s namespace="frontend/listing/listing" name="Listing
        <i class="icon--arrow-left"></i> {s namespace="frontend/listing/listin
      </a>
      <div class="offcanvas--content">
        <div class="content--title">{$sCategoryContent.cmsheadline}</div>
        {$sCategoryContent.attribute.attribute1}
      </div>
    </div>
  {/if}
</div>
</div>
{/block}

```

Das Ergebnis sieht dann wie folgt aus:


Desktop:



Mobile mit ausgeklappter OffCanvas Beschreibungsbox:

< Fenster schließen

519,00 € *



ROCKWELL

569,00 € *



A 10/30

750,00 € * ~~999,00 €~~

Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet. Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet. Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet. Duis autem vel eum iriure dolor in hendrerit in vulputate velit esse molestie consequat, vel illum

Gewisse Artikel immer als Lieferbar deklarieren

Ziel dieses Tutorials soll es sein, gewisse Artikel, zum Beispiel Dropshipping-Artikel als immer lieferbar zu deklarieren, egal wie hoch der Lagerbestand von Shopware ist.

Artikelattribut erstellen / pflegen

Da die Lieferzeit auch hier wieder für Artikel separat aktivierbar sein soll, macht es Sinn, dies über [Freitextfelder](#) zu regeln.

Erstelle also ein neues Freitextfeld vom Typ "Checkbox" für die Artikel, im Beispiel nutzen wir hier "attr5":

Datenbank Einstellungen

Spaltenname:

Spaltentyp:

Datenbank Spalte zurücksetzen

Anzeige Einstellungen

Label:

Support Text:

Hilfe Text:

Position:

Im Backend anzeigen:

Übersetzbar:

Das Freitextfeld ist nun im Backend verfügbar und kann bearbeitet werden.

Templateanpassung

Wie finde ich die Stelle, die ich anpassen muss?

Die Lieferbarkeit der Artikel wird in Shopware an vielen Stellen angezeigt, daher suchen wir in unserer Entwicklungsumgebung / Notepad++ innerhalb des Bare Themes wieder nach Stichwörtern, um deren Verwendung zu ermitteln. Dabei stellen wir fest, dass "delivery" zumeist in der `delivery_informations.tpl` verwendet wird, die auch an den relevanten Stellen im Frontend inkludiert wird.

Wir wissen also, welche Datei für die Lieferzeitenanzeige verantwortlich ist, da diese nur einen Block hat, ist auch dieser klar.

Änderung korrekt ableiten

Erstelle nun in `/themes/Frontend/DEINTHEME/frontend/plugins` eine Datei namens `delivery_informations.tpl`.

Zu aller erst wird definiert, dass diese Datei vom Original ableitet, die erste Zeile lautet also:

```
{extends file="parent:frontend/plugins/delivery_informations.tpl"}
```

Nun Ändern wir den Block so ab, dass wir auf unser Attribut abfragen und dann den entsprechenden Lieferstatus ausgeben. Der "Sofort Lieferbar" Status lässt sich im Template anhand des Namens ableiten: "DetailDataInfolnstock" - Alternativ kannst Du aber auch nach dem Textbaustein suchen und dessen Namen im Template suchen, das geht genauso. Nun bauen wir die Änderung ein:

```
{block name='frontend_widgets_delivery_infos'}
  {if $sArticle.attr5 == true || $sBasketItem.additional_details.attr5 == true}
    <div class="product--delivery">
      <link href="http://schema.org/InStock" itemprop="availability">
```



```

        <p class="delivery--information">
<span class="delivery--text delivery--text-available">
  <i class="delivery--status-icon delivery--status-available"></i>
      {s name="DetailDataInfoInstock"}}{/s}
</span>
</p>
</div>
{else}
  {smarty.block.parent}
{/if}
{/block}

```

Und somit wird der Artikel im Frontend auf "Sofort Lieferbar" deklariert, auch wenn der Bestand 0 ist.

Etwas aus dem Template entfernen

Falls Du Inhalt aus dem Frontend bewusst ausblenden willst, funktioniert das am einfachsten, indem Du den gewünschten Block leer überschreibst, dadurch wird der Inhalt im Frontend ausgeblendet. Dazu ein kurzes Beispiel:

```

{extends file="parent:PARENT"}
{block name='NAME'}
{/block}

```

Beachte, dass hier die gleichen Voraussetzungen gelten, wie in den anderen Beispielen auch, also immer nur in Deinem Theme ableiten und niemals das originale Theme ändern!

Anpassung der robots.txt

Da Shopware die robots.txt ebenfalls über das Theme ausliefert, kann diese auch bequem angepasst werden, der Pfad zur Template-Datei, die für die Erzeugung der robots.txt verantwortlich ist, ist folgender: /frontend/robots_txt/index.tpl

Hier kannst Du in Deinem Theme wieder entsprechend ableiten, um die robots.txt nach Deinen Wünschen anzupassen.

Abweichende Lieferadresse auf der Rechnung ausgeben

Bei den Dokumenten handelt es sich ebenfalls um Templatedateien, die Standard-Dokumententemplates findest Du unter Bare/documents, die Dateien haben die folgende Bedeutung:

Datei	Verantwortlich für
index.tpl	Rechnung
index_gs.tpl	Gutschrift

index_ls.tpl	Lieferschein
index_sr.tpl	Stornorechnung

Beispielhaft ergänzen wir nun den Block der Lieferart um die Angabe der Lieferadresse, falls diese abweichend zur Rechnungsadresse ist.

```
{block name="document_index_info_dispatch"}
  {if $Order._dispatch.name}
    <div style="font-size:11px;color:#333;">
      {s name="DocumentIndexSelectedDispatch"}{/s}
      {$Order._dispatch.name}
    </div>
  {/if}
  {assign var="shippingaddress" value="shipping"}
  {assign var="shippingArr" value=$User.$shippingaddress}
  {assign var="billingArr" value=$User.$address}

  {$billingString = $billingArr['salutation']|cat:$billingArr['title']|cat:$billingArr["first"]}
  {$shippingString = $shippingArr['salutation']|cat:$shippingArr['title']|cat:$shippingArr["first"]}

  {if $shippingString != $billingString}
    {s name="DocumentIndexSelectedShippingaddress"}{/s}

    <div style="margin-left: 20px;">
      {if $User.$shippingaddress}
        {$User.$shippingaddress.salutation|salutation}
        {if {config name="displayprofiletitle"}}
          {$User.$shippingaddress.title}<br/>
        {/if}
        {$User.$shippingaddress.firstname} {$User.$shippingaddress.lastname}

        {$User.$shippingaddress.street}

        {if {config name=showAdditionAddressLine1}}
          {$User.$shippingaddress.additional_address_line1}

        {/if}
        {if {config name=showAdditionAddressLine2}}
          {$User.$shippingaddress.additional_address_line2}

        {/if}
        {if {config name=showZipBeforeCity}}
          {$User.$shippingaddress.zipcode} {$User.$shippingaddress.city}
        {else}
          {$User.$shippingaddress.city} {$User.$shippingaddress.zipcode}
        {/if}
      {/if}
    </div>
  {/if}
{/block}
```

Weitere Länderflaggen für den Sprachwechsel hinzufügen

Im Standard sind für den Sprachwechsel die Flaggen für Deutschland, Großbritannien, die Niederlande, Frankreich, Spanien und Italien hinterlegt. Diese sind in der Datei /themes/Frontend/Responsive/frontend/_public/src/less/_components/flags.less hinterlegt. Um weitere Flaggen für die Auswahl bereitzustellen, ist eine Theme-Anpassung notwendig.

Anpassungen

flags.less

Die Datei flags.less enthält die Informationen zu den Lokalisierungen und Flaggen, die verwendet werden sollen.

Jede Lokalisierung erhält einen eigenen Eintrag, der mit dem ISO-Code beginnt. Diesen Code kannst Du in den Grundeinstellungen unter **Shopeinstellungen > Lokalisierungen** entnehmen.

Generell können Flaggen per CSS selbst erstellt werden.

Dies bietet sich vor allem für Flaggen an, die z.B. aus horizontalen/vertikalen Streifen bestehen.

Als Anhaltspunkt können auch die im Standard hinterlegten Flaggen herangezogen werden.

Auch komplexere Flaggen können so erstellt werden. Da hier allerdings mit mehreren Layern zu arbeiten ist, erfordert dies jedoch ggf. erweiterte CSS-Kenntnisse.

Als einfachere Alternative ist es möglich, eine fertige Flagge als Base64 einzubinden. Hierzu wird ein Bild in eine Zeichenfolge umformatiert und diese Zeichenfolge in der Datei hinterlegt.

Weitere Details zu Base64 sowie einen Konverter findest Du unter <https://www.base64-image.de/>

In Deinem Theme speicherst Du die angepasste Datei unter:

/themes/Frontend/DEINTHEME/frontend/_public/src/less/_components/flags.less

Beispiel

Flagge als CSS erstellen

Exemplarisch ist hier die Erweiterung für Ungarn aufgeführt

```
&.hu_HU { background-position: 0 0; background-image: linear-gradient(to bottom, #DD0000 33%,
```

Der Eintrag in der flags.less beginnt mit dem ISO-Code der Lokalisierung, in diesem Fall hu_HU. Danach folgen die Informationen zur Erstellung der Flagge. Der Anfang

```
background-position: 0 0; background-image:
```

ist für alle per CSS erstellten Flaggen identisch. Der Teil

```
linear-gradient(to bottom, #DD0000 33%, white 33%, white 66%, #336333 66%);
```

beinhaltet die Informationen zur Erstellung der Flagge. In diesem Fall wird die Flagge **von oben nach unten** ("linear-gradient(to bottom)") erstellt. Der erste Teil nimmt **33%** der Fläche ein und erhält die Farbe mit dem HEX-Wert **DD0000**. Der zweite Farbbereich beginnt bei 33% und endet bei 66% mit der Farbe Weiß. Der letzte Bereich beginnt bei 66% und erhält den Farbwert 336333).

Die vollständige Datei sieht dann wie folgt aus:

```
/*
Language Flags
=====
Displays a country flag the size of 14px x 11px used for language selections purposes.
inspired by: https://github.com/dhanishgajjar/css-flags
en_GB: we need to use svg for en_GB, cause it's smaller than the CSS-Version
*/
```

```
.language--flag {
  .unitize-height(11);
  .unitize-width(14);
  display: inline-block;
  text-indent: 100%;
  white-space: nowrap;
  overflow: hidden;

  &.de_DE { background-position: 0 0; background-image: linear-gradient(to bottom, black 33%
  &.en_GB { background-position: 0 0; background-size: 100%; background-repeat: no-repeat; b
  &.nl_NL { background-position: 0 0; background-image: linear-gradient(to bottom, #DD0000 3
  &.fr_FR { background-position: 0 0; background-image: linear-gradient(to right, #21468B 33
  &.es_ES { background-position: 0 0; background-image: linear-gradient(#DD0000 25%, #FFCE00
  &.it_IT { background-position: 0 0; background-image: linear-gradient(to right, #009246 33
  &.hu_HU { background-position: 0 0; background-image: linear-gradient(to bottom, #DD0000 3
}
```

Flagge als Base64 einbinden

Exemplarisch ist hier die Erweiterung für Schweden aufgeführt.

Suche Dir zunächst eine Bilddatei der Flagge, die Du verwenden möchtest und wandele diese in den Base64-Code um.

```
&.sv_SE { background-position: 0 0; background-size: 100%; background-repeat: no-repeat; backg
```

Der Eintrag in der **flags.less** beginnt mit dem ISO-Kürzel der Lokalisierung. Anschließend folgt ein für die Einbindung von Base64-Code allgemeingültiger Teil

```
background-position: 0 0; background-size: 100%; background-repeat: no-repeat; background-imag
```

Der individuelle Code wird dann daran angehängen.

Die vollständige Datei sieht dann wie folgt aus:

```
/*
Language Flags
=====
Displays a country flag the size of 14px x 11px used for language selections purposes.
inspired by: https://github.com/dhanishgajjar/css-flags
en_GB: we need to use svg for en_GB, cause it's smaller than the CSS-Version
*/
.language--flag {
  .unitize-height(11);
  .unitize-width(14);
  display: inline-block;
  text-indent: 100%;
  white-space: nowrap;
  overflow: hidden;

  &.de_DE { background-position: 0 0; background-image: linear-gradient(to bottom, black 33%
  &.en_GB { background-position: 0 0; background-size: 100%; background-repeat: no-repeat; b
  &.nl_NL { background-position: 0 0; background-image: linear-gradient(to bottom, #DD0000 3
  &.fr_FR { background-position: 0 0; background-image: linear-gradient(to right, #21468B 33
  &.es_ES { background-position: 0 0; background-image: linear-gradient(#DD0000 25%, #FFCE00
  &.it_IT { background-position: 0 0; background-image: linear-gradient(to right, #009246 33
  &.sv_SE { background-position: 0 0; background-size: 100%; background-repeat: no-repeat; b
}
```

Zahlungsstatus von Bestellungen im Frontend anzeigen

Am einfachsten kannst Du den Zahlstatus im Frontend unter dem Bestellstatus bereitstellen. Erweitere hierzu die "order_item.tpl" in Deinem eigenen Theme unter: /themes/Frontend/EIGENERTHEMENAME/frontend/account/. Dort kannst Du dann den folgenden Block mit dem Zahlstautstatus erweitern:

```
{extends file="parent:frontend/account/order_item.tpl"}
{block name="frontend_account_order_item_status_value"
  {$smarty.block.parent}
  <div class="column--value">
    <span class="order--status-icon status--{$offerPosition.cleared}"></span>
    {if $offerPosition.cleared==9}
      {s name="partially_invoiced"}Teilweise in Rechnung gestellt{/s}
    {elseif $offerPosition.cleared==10}
      {s name="completely_invoiced"}Komplett in Rechnung gestellt{/s}
    {elseif $offerPosition.cleared==11}
      {s name="partially_paid"}Teilweise bezahlt{/s}
    {elseif $offerPosition.cleared==12}
      {s name="completely_paid"}Komplett bezahlt{/s}
    {elseif $offerPosition.cleared==13}
      {s name="1st_reminder"}1. Mahnung{/s}
    {elseif $offerPosition.cleared==14}
      {s name="2nd_reminder"}2. Mahnung{/s}
    {elseif $offerPosition.cleared==15}
      {s name="3rd_reminder"}3. Mahnung{/s}
    {elseif $offerPosition.cleared==16}
      {s name="encashment"}Inkasso{/s}
    {elseif $offerPosition.cleared==17}
      {s name="open"}Offen{/s}
    {elseif $offerPosition.cleared==18}
      {s name="reserved"}Reserviert{/s}
    {elseif $offerPosition.cleared==19}
      {s name="delayed"}Verzoegert{/s}
    {elseif $offerPosition.cleared==20}
      {s name="re_crediting"}Wiedergutschrift{/s}
    {elseif $offerPosition.cleared==21}
      {s name="review_necessary"}Überprüfung notwendig{/s}
    {/if}
  </div>
{/block}
```

Icon korrekt zuweisen

Damit nun noch das entsprechende Icon neben dem Status angezeigt wird musst Du noch ein paar Anpassungen am Style vornehmen. Wenn Du in LESS erfahren bist, kannst Du wie gewohnt deine Anpassungen an den LESS Dateien vornehmen. Solltest Du nur diese Anpassungen vornehmen möchten, so kannst Du folgenden Code in eine all.less in dem Verzeichnis /themes/Frontend/EIGENERTHEMENAME/frontend/_public/src/less hinzufügen:

```
.order--status-icon {
  &.status--9, //partially_invoiced
  &.status--10, //completely_invoiced
  &.status--11, //partially_paid
  &.status--13, //1st_reminder
  &.status--14, //2nd_reminder
  &.status--15, //3rd Reminder
  &.status--17, //open
  &.status--18, //reserved
  &.status--19{ //delayed
    background: @highlight-info;
  }

  &.status--12, //completely_paid
  &.status--20{ //re_crediting
    background: @highlight-success;
  }

  &.status--16, //encashment
  &.status--21{ //review_necessary
```

```

}
}
background: @highlight-error;
}
}

```

Wenn Du nach der Anpassung den Cache leerst und Dein Theme neu kompilierst wird der Zahlstatus wie folgt im Frontend angezeigt:

Bestellungen

Hier finden Sie Ihre Bestellungen nach Datum sortiert.

Datum	Bestellnummer	Versandart	Bestellstatus	Aktionen
07.06.2017 09:35:00	20010	Standard Versand	<ul style="list-style-type: none"> ■ Bestellung wurde noch nicht bearbeitet ■ Offen 	Anzeigen
07.06.2017 09:34:00	20009	Standard Versand	<ul style="list-style-type: none"> ■ Bestellung wurde noch nicht bearbeitet ■ Teilweise bezahlt 	Anzeigen
07.06.2017 09:34:00	20008	Standard Versand	<ul style="list-style-type: none"> ■ Bestellung wurde noch nicht bearbeitet ■ Inkasso 	Anzeigen
07.06.2017 09:30:00	20007	Standard Versand	<ul style="list-style-type: none"> ■ Bestellung wurde noch nicht bearbeitet ■ Komplette bezahlt 	Anzeigen

Shopware Logo und Schriftzug aus dem Footer entfernen

Logo entfernen

Über eine Template-Anpassung kannst Du das Shopware Logo aus dem Footer entfernen.



* Alle Preise inkl. gesetzl. Mehrwertsteuer zzgl. Versandkosten und ggf. Nachnahmegebühren, wenn nicht anders beschriftet.

Realized mit Shopware



Beispiel

Zunächst musst Du Dich mit Deinem FTP-Server verbinden. Im Shopware-Verzeichnis hast Du unter "**themes > Frontend > DEIN THEME > frontend > index**" die Möglichkeit, eine Datei einzubinden, welche das Logo entfernt.

Erstelle hierzu im genannten Pfad die Datei "footer.tpl" mit folgendem Inhalt:

```

{extends file="parent:frontend/index/footer.tpl"}
{block name="frontend_index_shopware_footer_logo"}
{/block}

```

Bitte beachte hierbei, dass es sich um ein Beispiel handelt und ggf. weitere Änderungen am Code vorgenommen werden müssen. Außerdem solltest Du darauf achten ein eigenes Theme zu erstellen, welches vom originalen Theme abgeleitet wird, damit Deine Änderungen update-sicher sind.

Nachdem Du die Datei erstellt, das Theme neu kompiliert und den Cache geleert hast, sollte das Logo entfernt sein.

Schriftzug entfernen

Falls Du den Schriftzug "Realisiert mit Shopware" ändern bzw. entfernen möchtest, kannst Du dies in der **Textbaustein-Verwaltung** unter **Einstellungen > Textbausteine** im Backend umsetzen. Hier kannst Du einfach nach dem Textbaustein "**IndexCopyright**" suchen und den Wert entfernen.

AGPLv3

Bedeutung für Entwickler und Kunden

Die Affero General Public License, Version 3, (AGPLv3) ist ein Derivat der GPLv3 und gestattet jedermann die lizenzgebührenfreie Nutzung der Software. Shopware 4.0 CE ist damit Freie Software bzw. Open Source Software und darf kopiert, weiterverbreitet und auf die eigenen Bedürfnisse angepasst werden. Die AGPLv3 enthält wie die GPL eine Copyleft Klausel, d.h. Bearbeitungen von Shopware CE müssen ebenfalls wieder unter der AGPLv3 lizenziert werden. Entwickler erhalten Zugang zum Sourcecode und dürfen Shopware CE beliebig modifizieren und in modifizierter Form unter Beachtung der Lizenzbedingungen weiterverbreiten. Kunden dürfen Shopware lizenzgebührenfrei einsetzen, müssen aber die Lizenzpflichten der AGPLv3 erfüllen. Der Zugang zum Sourcecode stellt sicher, dass Shopware CE nachhaltig eingesetzt werden kann und erforderliche Anpassungen auf der Basis der AGPLv3 beauftragt werden können.

Template Änderungen

Das Standard-Template von Shopware CE ist anders als der Rest von Shopware CE unter der New BSD lizenziert. Dies erlaubt es bei Eigenentwicklungen von Frontend Templates, trotz des verwendeten Block Systems und damit geerbter Teile des Standard-Templates eine Lizenz nach den eigenen Wünschen oder entsprechend den Wünschen des Kunden zu wählen. Es ist wichtig zu beachten, dass dies für Backend-Templates von Shopware 4.0 CE nicht gilt. Diese sind bei Shopware 4.0 CE unter der AGPLv3 lizenziert, und daher müssen alle angepassten Versionen des Standard-Backendtemplates ebenfalls wieder unter der AGPLv3 lizenziert werden, da es sich dann um ein "work based on the program" im Sinne der AGPLv3 handelt.

Plugins

Wir möchten Entwicklern von Plugins vielfältige Geschäftsmodelle ermöglichen und keine Lizenzvorgaben für selbst geschriebene Plugins machen. Daher erlauben wir als Ausnahme von den Lizenzbedingungen der AGPLv3 eine Lizenzierung von selbst geschriebenen Plugins unter anderen Lizenzbedingungen als der AGPLv3, auch wenn ein Plugin von der Klasse "Shopware_Components_Plugin_Bootstrap" abgeleitet ist. Eine Freigabe als Open Source Software ist daher nicht zwingend. Der Lizenztext der AGPLv3 enthält aus diesem Grund den nachfolgenden Zusatz: Additional permission under AGPLv3, Sec. 7 If you modify this Program, or any covered work, by combining it with Plugins entirely written by you, the licensors of this Program grant you additional permission to convey those Plugins under terms of your choice, provided that those terms do not place additional restrictions on the Program.

Core Anpassungen

Alle Modifikationen und Erweiterungen des Shopware Sourcecodes, die nicht das Frontend Template betreffen oder im Rahmen des Plugin Systems erfolgen, sind Modifikationen in Sinne der AGPLv3 und müssen daher unter der AGPLv3 lizenziert werden. Dies gilt auch für Override-Klassen.

Änderungen an Shopware CE müssen dann unter der AGPLv3 lizenziert werden, wenn die geänderte Software an Dritte weitergeben wird oder wenn ein Shop damit betrieben wird. Jegliche Weitergabe von Kopien veränderter Shopware-Versionen wird in der AGPLv3 als "convey" bezeichnet und führt dazu, dass der Sourcecode der geänderten Versionen unter den Bedingungen der Ziffern 5 und 6 der AGPLv3 zur Verfügung gestellt werden muss. Auch der bloße Betrieb eines Shops führt dazu, dass Nutzer die Software verwenden. Dies wird in Ziffer 13 AGPLv3 als "interacting with it remotely through a computer network" bezeichnet und führt ebenfalls zur Pflicht der Freigabe von Änderungen (siehe FAQ 6).

Änderungen freigeben

Die AGPLv3 enthält in den Ziffern 5 und 6 genaue Vorgaben, welche Lizenzpflichten bestehen, wenn Shopware CE geändert wurde: - Änderungen müssen gekennzeichnet und mit einem Datum versehen werden. Am besten werden diese Angaben im Header des Sourcecodes gemacht (z.B. `/** * 31.07.2012: install method changed, © 2012 Max Muster * This program is free software: you can redistribute it and/or modify * it under the terms of the GNU Affero General Public License as * published by the Free Software Foundation, either version 3 of the * License, or (at your option) any later version. */` - Der maschinenlesbare und vollständige Sourcecode der geänderten Shopware CE muss zugänglich gemacht werden. Die AGPLv3 gestattet hier mehrere Varianten. Am einfachsten ist es, den Sourcecode direkt mitzuliefern oder, wenn Shopware 4 nur online genutzt oder weiterverbreitet wird, auf derselben Website kostenlos zum Download anzubieten. Dabei darf der Sourcecode auch auf einem anderen Server gehostet werden, wenn der Downloadlink in unmittelbarer Nähe zum Download des Objektcodes angebracht wird und sichergestellt ist, dass der Sourcecode auch tatsächlich downloadbar ist. Die Details können dem Lizenztext der AGPLv3 entnommen werden.

Unterschiede zwischen der GPL und der AGPLv3

Die GNU General Public License, Version 3, (GPLv3) enthält ein schwächeres Copyleft als die AGPLv3. Die Pflicht, Modifikationen der Software wieder als Freie Software zu lizenzieren, besteht bei der GPLv3 nur beim herkömmlichen Vertrieb der Software, aber nicht bei SaaS- oder Cloud-Diensten, bei denen die Software ausschließlich auf einem Server abläuft. Wir möchten sicherstellen, dass auch in diesen Fällen der Sourcecode von Änderungen an Shopware 4 CE zur Verfügung gestellt werden muss und haben uns daher für die AGPLv3 entschieden. Wer modifizierte Versionen von Shopware 4 CE betreibt, muss gem. Ziffer 13 AGPLv3 auch den modifizierten Sourcecode unter der AGPLv3 zur Verfügung stellen. Davon abgesehen unterscheiden sich GPLv3 und AGPLv3 nicht.

Markenrecht

Nein. "Shopware" ist eine eingetragene Marke der Shopware AG. Daher dürfen auch keine mit "Shopware" verwechselungsfähigen Bezeichnungen (wie z.B. "MyShopware") verwendet werden. Unsere Kunden sollen einfach erkennen können, welche Shopware-Versionen von uns stammen. Dies gilt nicht nur für den Namen unter dem eine modifizierte Version vertrieben wird, sondern auch für alle Bezeichnungen innerhalb der Software. Die AGPLv3 erlaubt jedoch, geänderte Versionen von Shopware 4 CE unter einer anderen

Bezeichnung weiterzugeben.

Support für Open Source Plugins / Templates

Nein, wer seine selbst geschriebenen Plugins oder Templates unter einer Open Source-Lizenz anbietet, muss keinen Support dafür anbieten. Wenn Plugins oder Templates jedoch verkauft werden, muss im Regelfall auch für die Qualität nach den Regelungen des Kaufrechts Gewährleistung gegeben werden.. Davon unabhängig darf immer auch zusätzlicher (entgeltlicher) Support angeboten werden.

Kompatibilität der AGPLv3

Die AGPLv3 ist u.a. mit folgenden Lizenzen kompatibel: GPLv3, GPLv2 or any later version, LGPLv3, LGPLv2.1 or any later version, BSD license (2-clause and 3-clause), MIT license, Apache License 2.0. Kompatibel bedeutet, dass Code, der unter diesen Lizenzen genutzt werden darf, zusammen mit Shopware 4 CE einheitlich unter der AGPLv3 lizenziert werden kann.

Ingeration in bestehende Software-Lizenzen

Eine Implementierung ist nur möglich, wenn die andere Open Source Lizenz kompatibel mit der AGPLv3 ist. Dann ist eine Nutzung der modifizierten Shopware insgesamt unter der AGPLv3 zulässig. Dabei ist aber darauf zu achten, dass zusätzliche Lizenzpflichten, die nicht standardmäßig in der AGPLv3 enthalten, aber nach Ziffer 7 AGPLv3 zulässig sind, gesondert aufgeführt werden. Dafür fügt man dem Lizenztext folgenden Hinweis bei: "Additional Terms:.....". Zudem muss der Text der Lizenz des implementierten Codes gesondert beigefügt werden sowie die Information, aus welchem Programm dieser Code stammt

Backend-Passwort zurücksetzen

Vorwort

Um das Backend Passwort zurückzusetzen, benötigst Du Zugriff auf Deine Shopware-Datenbank oder Zugriff via SSH auf Deinen Shop.

Passwort ändern

Über die Datenbank

Um das Passwort auf "demo" zu ändern, öffne phpMyAdmin und öffne die Tabelle **s_core_auth** in deiner Datenbank. Hier bearbeitest Du Deinen gewünschten Datensatz und setzt folgende Daten:

```
password: $2y$10$av4vFeRXw8d/f5bEaV.pzu5J4nEsEiaV/mAbxskbtYhf.HfFitMzW  
encoder: bcrypt
```

Nach dieser Änderung kannst Du Dich mit dem neuen Passwort anmelden.

Über SSH/CLI

Wenn Du mit SSH auf Deinen Shop zugreifen kannst, dann kannst Du Dir mit folgenden CLI-Befehl einen neuen Admin Benutzer erstellen:

```
php bin/console sw:admin:create
```

Hier findest Du noch ein paar weitere Informationen zu den Shopware CLI Befehlen: [Shopware CLI: Nützliche Befehle & Tricks](#)

```
php ../bin/console sw:admin:create --email=$EMAIL --username=$USERNAME --name=$NAME --locale=$
```

Benutzerrechte für Plugins ohne ACL-Unterstützung

Einleitung

Mitunter kann es vorkommen, dass ein im Store erworbenes Plugin die Benutzerverwaltung von Shopware nicht unterstützt und keine ACL-Ressource bereitstellt. In bestimmten Fällen kann es dennoch gewünscht sein, dass nicht alle Benutzer Zugriff auf den jeweiligen Menüpunkt haben. In diesem Tutorial erklären wir, wie Du über eine kleine manuelle Konfiguration im Backend den Menüpunkt für einzelne Benutzer ausblenden kannst.

Auslesen des Controllers

Zunächst benötigst Du den Aufruf, der im Backend stattfindet, wenn der Menüpunkt von einem Benutzer angeklickt wird. Diesen kannst Du am Besten mit Firebug (Firefox Add-On) oder dem Chrome-Debugger (F12) auslesen. In unserem Beispiel nehmen wir die Anpassung für das Plugin "Business Essentials" vor. Bei **Business Essentials** handelt es sich um den GET-Parameter **/backend/BusinessEssentials?...**

Firebug

Im Firebug kannst Du die URL entweder im Reiter **Konsole** oder auch im Reiter **Netzwerk** ausgelesen werden. Am besten leerst Du die Ansicht einmal komplett, bevor Du auf den gewünschten Menüpunkt klickst. Direkt der erste GET-Parameter sollte hier die konkrete URL für den Menüpunkt sein.



Chrome-Debugger

Im Chrome-Debugger verhält es sich ähnlich. Auch hier kann die Information über den Reiter **Netzwerk** ausgelesen werden.



Anlegen der Benutzerrechte

Anhand der URL kannst Du nun den genauen Controller-Namen bestimmen. Im Fall von Business Essentials ist das **BusinessEssentials**. Die Parameter, die im GET-Aufruf nach "?" kommen, kannst Du ignorieren. Der Aufruf kann aber bei den verschiedenen Plugins auch stark vom Plugin-Namen abweichen. Mit diesem Wissen kannst Du nun über "*Einstellungen > Benutzerverwaltung*" eine neue Regel & Berechtigung anlegen. Wechsel dazu zunächst in den Menüpunkt "*Regeln & Berechtigungen*" und wähle dort eine Rolle aus. Im Anschluss kannst Du über "*Ressource hinzufügen*" eine neue Berechtigung für Business Essentials anlegen.

Bei der Namensgebung ist es wichtig, dass der Name der Ressource exakt dem lower-case (alle Buchstaben klein) des Controller-Aufrufs entspricht. In unserem Beispiel ist dies also **businessessentials**. Lege also nun eine Ressource mit dem Namen businessessentials an und füge im Anschluss das Privileg **read (1)** hinzu. Alle Rollen, die nun den Menüpunkt sehen sollen, müssen auch das Privileg **read** zugewiesen bekommen. Für die Rollen, die den Menüpunkt nicht sehen sollen, muss die gesamte Ressource **businessessentials** deaktiviert werden. Alle weiteren Funktionen der Benutzerverwaltung müssen individuell im Plugin umgesetzt werden und sind ohne Programmieraufwand im Plugin nicht umsetzbar.

Ausblenden des Hauptmenüs

Es ist ebenfalls möglich die Hauptmenü-Einträge auszublenden. Hierzu musst du den Controller-Namen aus der `s_core_menu` für den Menü-Eintrag auslesen und in lower-case als Resource wie oben beschrieben anlegen.

Berechnung des Warenkorbbes

Einleitung

Das Herzstück eines Shops ist der Warenkorb. Eine nachvollziehbare und korrekte Berechnung der Summen, Positionen und Mehrwertsteuern ist essentiell um das Vertrauen der Kunden in den Shop nicht zu stören. Aus diesem Grund soll dieser Artikel einmal über die Berechnungsgrundlage des Warenkorbbes aufklären und Dir Möglichkeiten an die Hand geben, mit denen Du die Berechnung selbst überprüfen kannst. Neben zahlreichen Grundfunktionen von Shopware wie Preisgruppen, Ab- und Aufschläge, Versandkosten, Gutscheine und Promotionen, gibt es natürlich auch unzählige Plugins die sich in die Berechnung des Warenkorbbes einhängen. Somit ist es wichtig, dass Du Dir auch ein Bild davon machst, welche Erweiterungen die Berechnung eventuell beeinflussen und welche Auswirkungen dies haben kann.

Grundlagen

In Deutschland gibt es zwei valide Methoden zur Berechnung der Mehrwertsteuer. Das **vertikale** und das **horizontale** Verfahren. Shopware verwendet das **vertikale** Verfahren, hier wird auf Positionsebene die enthaltene Mehrwertsteuer errechnet, gerundet und abschließend zusammenaddiert. Im Gegensatz dazu wird beim **horizontalen** Verfahren zunächst die Summe der enthaltenen Mehrwertsteuer gebildet und anschließend gerundet. Die Verwendung des **vertikalen** Verfahrens ermöglicht das Ausweisen der enthaltenen Mehrwertsteuer pro Position auf der /checkout/confirm-Seite. Dies ist beim **horizontalen** Verfahren nicht möglich.

Shopware zeigt zudem an allen Stellen des Frontends zwei Nachkommastellen. Damit deine Kunden die Berechnung stets nachvollziehen können, wird auch immer mit dem auf 2-Stellen gerundeten Wert gerechnet.

Berechnungsbeispiele

Nettopreise

Vertikal (von Shopware verwendet)

Anzahl	MwSt. Satz	Netto	Brutto	zzgl. MwSt.		Netto Gesamt	zzgl. MwSt.	Brutto Gesamt
5	7	1,49	1,59	0,10		5*1,49	5*0,10	7,45+0,50
						7,45	0,50	7,95

Horizontal

Anzahl	MwSt. Satz	Netto	Brutto		Netto Gesamt	zzgl. MwSt.	Brutto Gesamt
5	7	1,49	-		5*1,49	7,45*0,07	7,45+0,52
					7,45	0,52	7,97

Bruttopreise

Vertikal (von Shopware verwendet)

Anzahl	MwSt. Satz	Netto	Brutto	enth. MwSt.		Netto Gesamt	enth. MwSt.	Brutto Gesamt
5	7	1,49	1,59	0,10		7,95-0,50	5*0,10	5*1,59
						7,45	0,50	7,95

Horizontal

Anzahl	MwSt. Satz	Netto	Brutto		Netto Gesamt	zzgl. MwSt.	Brutto Gesamt
5	7	-	1,59		7,95/1,07	7,43-7,95	5*1,59
					7,43	0,52	7,95

Erklärung

Die Berechnungsunterschiede entstehen hier auf Basis der verwendeten Berechnungsmethode. In Onlineshops ist die **vertikale** Berechnung durchaus die gängigere, da es vielen Shopbetreibern wichtig ist, die enthaltene Mehrwertsteuer auf Positionsebene ausweisen zu können. Dies wäre bei der **horizontalen** Berechnung nicht möglich, da die gerundeten Positionswerte niemals den Gesamtwert ergeben können.

Debugging

Shopware hat keine zentrale Stelle zur Berechnung des Warenkorbes. Die Berechnungen im Front- und Backend sind grundsätzlich voneinander getrennt. Daher solltest Du immer den kompletten Weg in deinem Shop nachvollziehen, wenn Du hier einen Fehler vermutest. Am besten gehst Du wie folgt vor, wenn sich ein Kunde bei Dir meldet:

- **Warenkorb im Frontend nachstellen**
- **Screenshot der Bestellabschlussseite anfertigen (/checkout/finish)**
- **Summen in der Bestellübersicht mit dem Screenshot vergleichen**
- **Summen in der PDF-Rechnung mit dem Screenshot vergleichen**
- **Errechneten Warenkorb mit [Excel-Tabelle](#) überprüfen**

Sollte sich hier ein Problem bestätigen, solltest Du zunächst prüfen, ob eines der eingesetzten Plugins in die Berechnung des Warenkorbes eingreift. Hierzu kannst Du den Sicherheitsmodus nutzen oder einen

Testshop aufsetzen, der keine Plugins einsetzt.

Defektes Plugin entfernen

Unter Umständen kann es dazu kommen, dass ein Plugin nicht mehr richtig funktioniert und sich ggf. auch nicht mehr korrekt im Plugin Manager deaktivieren oder deinstallieren lässt. In diesen Fällen ist es Möglich, das Plugin und die dazugehörigen Plugin Daten manuell zu entfernen.

Beachte, dass Du hierbei Daten aus der Datenbank und Dateien von deinem Server entfernst. Stelle also sicher, dass Du eine Backup der Datenbank und Deiner Daten angelget hast, um im Notfall alles zurücksichern zu können.

Funktionen deaktivieren

Zunächst solltest Du die Funktionen des betroffenen Plugins deaktivieren. Handelt es sich beispielsweise um ein Zahlungsplugin oder Versandplugin, deaktiviere die jeweilige Zahlungsart oder die Versandmethode, damit sie nicht mehr von Kunden genutzt werden kann. Generell sollten die Funktionen, die das defekte Plugin bereit gestellt hatte nicht mehr genutzt werden, damit es beim Entfernen des Plugins nicht zu Problemen kommt.

Datenbankeintrag entfernen

Wenn Du das Plugin nur deaktivieren möchtest, reicht es, in der Datenbank Tabelle **s_core_plugins** den Eintrag in der Spalte **active** des jeweiligen Plugins auf 0 zu setzen. Dies kann schon ausreichen, wenn ein defektes Plugin nicht mehr genutzt werden soll und nur deaktiviert werden soll. Möchtest Du es komplett entfernen, weil es nach wie vor Probleme macht oder weil es sogar komplett neu installiert werden soll, kannst Du den Datensatz des Plugins in der Datenbank Tabelle **s_core_plugins** entfernen.

Plugindaten entfernen

Unter Umständen hat das Plugin auch ein Pluginverzeichnis angelegt, welches noch physikalisch von Deinem Server entfernt werden sollte. Dies findest Du in der Regel im Shopware Hauptverzeichnis unter **engine/Shopware/Plugins/**. Dieses Verzeichnis hat die drei Unterordner **Community**, **Default** und **Local**, welche jeweils die drei Unterordner **Backend**, **Core** und **Frontend** haben. Das Verzeichnis des Plugins ist in der Regel in einem dieser Unterordner zu finden. Sollte es hier nicht zu finden sein, solltest Du dich am besten noch mal beim Hersteller des Plugins erkundigen, ob und wo die physikalischen Daten abgelegt werden.

Datenschutz-Grundverordnung (DSGVO)

Am 25. Mai 2018 wird die europäische Datenschutz-Grundverordnung (DSGVO) wirksam und löst nationale Regelungen ab. Gültigkeit hat die DSGVO für alle Unternehmen und Institutionen, die in der EU tätig sind und mit personenbezogenen Daten, wie Namen, Adressen, Bankdaten und Geburtstagen arbeiten. Die Vereinheitlichung der Verarbeitung von personenbezogenen Daten zielt vor allem auf bessere Transparenz im Umgang mit Verbraucherdaten und einem erweiterten Schutz dieser Daten ab. Dieser Wiki-Artikel soll Dir eine kurze Zusammenfassung über die Verarbeitung von personenbezogenen Daten in Shopware geben, damit Du die benötigten Dokumentationen (bspw. das Verzeichnis von Verarbeitungstätigkeiten oder kurz VVT) oder die Datenschutzerklärung für eine Shopware-Instanz besser erstellen kannst. Da die Umsetzung der DSGVO von Unternehmen zu Unternehmen unterschiedlichste Ausmaße annimmt, solltest Du bei der Umsetzung in Deinem Shop stets Rücksprache mit Deinem Rechtsbeistand halten, um hier alle Anforderungen zu erfüllen. Einen allgemeinen Überblick über die Inhalte der DSGVO könnt ihr euch in unserem Whitepaper verschaffen.

Welche personenbezogenen Daten werden in Shopware verarbeitet und gespeichert?

In diesem Punkt möchten wir Dir aufzeigen, welche personenbezogenen Daten direkt in Shopware verarbeitet werden. Als eCommerce-System ist es natürlich zwingend notwendig, Daten des Kunden zu verarbeiten um bspw. Lieferadressen aufzunehmen. Hier solltest Du beachten, dass natürlich nicht nur die Software selbst an der Verarbeitung beteiligt ist, sondern auch die dahinterliegende Hardware. Da das jeweilige Setup Deines Hosters stark individuell ist, solltest Du ggf. auch mit diesem klären, inwieweit dort die Kommunikation stattfindet (bspw. bei einem getrennten Datenbankserver, wo die Applikation über ein Netzwerk mit der Datenbank kommuniziert). Hier soll es rein um die softwareseitige Verarbeitung dieser Daten in Shopware gehen.

Dieser Punkt ist bewusst erstmal sehr allgemein gehalten. Shopware selbst speichert unterschiedlichste Daten, die entweder einen direkten Bezug (= personenbezogen) oder keinen direkten Bezug (= anonymisiert) zum Benutzer haben. Anonymisierte Daten sind keine personenbezogenen Daten und fallen nicht unter die DSGVO, sodass hierfür keine besonderen Maßnahmen zu treffen sind.

Pseudonymisierte Daten, die man z.B. über eine Kennziffer einer Person zuordnen kann (z.B. über die Bankverbindung, eine Kundennummer oder eine Bestellnummer) sind wieder personenbezogene Daten, auch wenn nicht sofort erkennbar ist, wer die Person ist.

Die personenbezogenen Daten werden immer dann benötigt, wenn Shopware eine Eingabe des Kunden entgegennimmt oder der Shopbetreiber mit diesen Daten arbeiten muss. Dies kann zum einen die Registrierung oder Bewertungsfunktion im Frontend sein, aber auch die Verarbeitung von Bestellungen über das Backend. Natürlich können auch über die RestAPI mit Autorisierung personenbezogene Daten abgeholt werden. Zudem werden natürlich auch anonymisierte Daten erfasst um bspw. Artikelempfehlungen ("Kunden kauften auch", "Kunden haben sich ebenfalls angesehen", ...) und Statistiken im Backend ausspielen zu können. Da man bei den unzähligen Funktionen von Shopware schnell den Überblick verliert, möchten wir Dir hier auflisten, welche Daten in Shopware gespeichert werden.

Diese Liste kann unvollständig sein. Plugins sind in der Lage die Datenhaltung zu erweitern und weitgehende Bereiche in Shopware zu erweitern. Ob Du weitere personenbezogene Daten verarbeitest, musst Du individuell in Deinem Shop evaluieren!

Die DSGVO definiert in Art. 4 Nr. 1 DSGVO personenbezogene Daten wie folgt: „alle Informationen, die sich auf eine identifizierte oder identifizierbare natürliche Person beziehen; als identifizierbar wird eine natürliche Person angesehen, die direkt oder indirekt, insbesondere mittels Zuordnung zu einer Kennung, wie einem Namen, zu einer Kennnummer, zu Standortdaten, zu einer Online-Kennung oder zu einem oder mehreren besonderen Merkmalen, die Ausdruck der physischen, physiologischen, genetischen, psychischen, wirtschaftlichen, kulturellen oder sozialen Identität dieser natürlichen Person, identifiziert werden kann.“

Kundendaten

Um eine Bestellung und weitere Handlungen im Shop durchführen zu können, benötigt der Kunde ein Kundenkonto. Dies enthält unter anderem die Adressdaten, aber auch weitere personenbezogene Informationen, je nachdem was Du in Deinen Einstellungen konfiguriert hast. Eine Abfrage von Geburtsdatum ist ebenso denkbar, wie die Angabe des Firmennamens. Hier solltest Du konkret die Registrierung Deines Shops prüfen, damit Du einen Überblick erhältst, welche Daten Du vom Kunden sammelst. In der Datenbank von Shopware beginnen alle Tabellen die Kundendaten enthalten mit dem Kürzel "s_user*". Aber auch im Backend kannst Du diese Daten natürlich einsehen, indem Du unter Kunden > Kunden, den jeweiligen Kunden öffnest.

Bestellungen

Das Herzstück eines jeden Onlineshops ist natürlich der Checkout. Um eine Bestellung in Shopware auslösen zu können, benötigt der Kunde natürlich zunächst ein Kundenkonto. All diese Informationen werden auch zusammen in der Bestellung gespeichert. Dazu kommen die Produkte die der Kunde in Deinem Shop erwirbt und auch die IP-Adresse des Kunden. Darüber hinaus wird auch der Referrer - also die Quelle, über die der Kunde auf Deinen Shop gelangt ist, mitgespeichert. Alle relevanten Informationen kannst Du über das Backend unter Kunden > Bestellungen einsehen, sowie in der Datenbank in den Tabellen mit dem Kürzel "s_order". Weitere Informationen zur Verarbeitung von IP-Adressen findest Du hier: [Werden IP-Adressen in Shopware gespeichert?](#)

Bankdaten (SEPA)

Die Shopware-eigene SEPA-Funktion wird heutzutage kaum noch genutzt. Die meisten Shopbetreiber benutzen eine externe Lösung, die diese Daten nicht in Shopware speichert, sondern nur den Zahlungserfolg an Shopware überträgt (bspw. PayPal Plus). Solltest Du die shopwareeigene SEPA-Funktion nutzen, ist dies natürlich auch eine Verarbeitung von personenbezogenen Daten. Unsere SEPA-Funktion findest Du hier erklärt: [SEPA-Lastschrift](#) Die vom Kunden eingegebenen SEPA-Daten werden in der Tabelle s_core_payment_data aufbewahrt.

IP-Adressen

Wie oben bereits erwähnt, werden die IP-Adressen zu einzelnen Bestellungen gespeichert. Darüber hinaus findest Du eine Auflistung von Bereichen, in denen eine IP-Adresse gespeichert wird, in diesem Dokument: [Werden IP-Adressen in Shopware gespeichert?](#)

Newsletter

Wie viele andere Systeme auch, bietet Shopware eine Funktion zur Newsletter-Registrierung. Hier findet sich bspw. das Formular im Footer des Shops oder ein erweitertes Formular über den Aufruf /newsletter.

Welche Daten hier erhoben werden, ist stark abhängig von der Grundeinstellung "Erweiterte Felder in der Newsletter-Registrierung abfragen". Je nach Definition werden hier weitere Informationen wie der Vor- und Nachname des Kunden erhoben. Eine Dokumentation zum Newsletter findest Du hier: [Newsletter](#)

Alle Informationen, die der Kunde im Frontend hinterlegt, findest Du in der Administrationsoberfläche im Bereich "Marketing > Newsletter > Verwaltung > Empfänger". In der Datenbank werden die Tabellen `s_campaigns_mailaddresses` und `s_campaigns_maildata` zur Speicherung der Daten genutzt. Darüber hinaus kannst Du auch definieren, ob bei einer Registrierung zum Newsletter das Double-Optin Verfahren genutzt werden soll, hierzu benutzt Du die Grundeinstellung "Double-Opt-In für Newsletter-Anmeldungen". Die Optin-Daten werden in der Tabelle `s_core_optin` gespeichert.

Formulare

Die Formulare in Shopware versenden eine Mail, mit den von Kunden eingegebenen Daten, an die im Formular hinterlegte Adresse. Welche Daten dort abgefragt werden, liegt im Ermessen des Shopbetreibers und kann stark individualisiert werden/variieren. Welche Daten Du dort also wie verarbeitest, solltest Du selbstständig unter Inhalte > Formulare im Backend prüfen. Die Dokumentation zur Formularfunktion in Shopware findest Du hier: [Formulare](#)

Bewertungen

Auf der Artikeldetailseite bietet Shopware die Möglichkeit eine Produktbewertung abzugeben. Diese muss per Double-Optin bestätigt werden, wenn die Grundeinstellung "Double-Opt-In für Bewertungen" aktiviert ist. Für das Optin-Verfahren ist die Tabelle `s_core_optin` zuständig. Die bestätigten Bewertungen können dann im Backend unter Artikel > Bewertungen eingesehen, gelöscht oder freigegeben werden. Hier findest Du eine Dokumentation zur Bewertungsfunktion in Shopware: [Artikelbewertungen](#)

Backend

Die Administrationsoberfläche ist eine kleine Besonderheit in diesem Bereich, da man dort natürlich ebenfalls Kundendatensätze einsehen und anlegen kann, aber auch Administrationsbenutzer, die ebenfalls mit einer Mailadresse verknüpft sind. Das Backend verarbeitet also nicht nur die Daten der Kunden in Deinem Shop, sondern auch ein Stück weit die Daten Deiner Mitarbeiter. Die Benutzerverwaltung des Backends kannst Du unter Einstellungen > Benutzerverwaltung konfigurieren. Es gibt ebenfalls die Möglichkeit einzelne Bereiche des Backends zu beschränken oder den Zugriff nur lesend zu erlauben. Eine Dokumentation zu Rechten und Rollen im Backend findest Du hier: [Benutzerverwaltung](#)

RestAPI

Wenn Du weitere Systeme, wie bspw. ein ERP angeschlossen hast, kommuniziert dieses in der Regel über die RestAPI mit dem Shop. Die Benutzer der Rest-API werden wie oben beschrieben über das Backend angelegt. Hier findest Du eine Auflistung von Daten, die Du über die Rest-API erhalten kannst: [REST](#)

Welche Daten werden anonymisiert gespeichert?

Shopware nutzt anonymisierte Inhalte, um Deinen Kunden im Shop einzelne Cross-Selling Vorschläge anzubieten. Dies betrifft bspw. die Funktionen "Kunden kauften auch" und "Kunden haben sich ebenfalls angesehen". Dort wird das Verhalten der Kunden in Deinem Shop anonymisiert ausgewertet und für die

Anzeige von Empfehlungen genutzt. Diese Funktionen kannst Du im Backend über das Performance-Modul verwalten. Weitere Dokumentation zu diesem Bereich findest Du hier: [Crossselling](#)

Statistiken

Im Backend kannst Du zudem Statistiken zum Kaufverhalten, Bestellungen und vielem mehr einsehen. Eine Dokumentation zu den Statistiken findest Du auch in unserer Dokumentation: [Auswertungen](#)

Das Erfassen von Statistiken kannst Du auch über das Performance-Modul verwalten, zudem findet sich in den Grundeinstellungen auch die Möglichkeit einzelne IP-Adressen von der Erfassung auszuschließen. Diese Daten werden aus allen Daten die in der Shopware-Datenbank erfasst werden, festgehalten.

Welche Informationen werden verschlüsselt übertragen?

Die verschlüsselte Übertragung von Daten erfolgt im Regelfall über das HTTPS-Protokoll. Um dies in Deinem Shop nutzen zu können, benötigst Du zunächst ein gültiges SSL-Zertifikat, welches auf Deinem Server eingerichtet werden muss. Im Anschluss kannst Du anhand dieser Anleitung SSL in Deinem Shop aktivieren: [SSL-Zertifikate im Shop einrichten](#) Hier muss generell zwischen der verschlüsselten Übertragung von Daten und der verschlüsselten Speicherung unterschieden werden. Wir empfehlen grundsätzlich alle Bereiche des Shops per HTTPS zu betreiben, damit alle personenbezogenen Daten per HTTPS verschlüsselt übertragen werden. Die Speicherung dieser Daten erfolgt dann in den jeweiligen Datenbank-Tabellen. Die Daten an sich sind nur dann verschlüsselt, wenn dies erforderlich ist, z.B. das Benutzerkennwort. Die Verschlüsselung von Passwörtern kann in den Grundeinstellungen definiert werden.

Welche Informationen werden im Browser des Nutzers gespeichert?

Ob das Setzen von Cookies, die technisch für den Betrieb des Shops nicht erforderlich sind, ein Opt-in notwendig ist oder nicht, ist zwischen Juristen strittig. Wir empfehlen, mindestens eine der etablierten Lösungen für die datenschutzrechtliche Information über Cookies umzusetzen. Mehr Informationen dazu findest Du unter www.cookiechoices.org; bitte nimm hierzu mit deinem Rechtsbeistand Rücksprache.

Shopware selbst bietet bereits im Standard die Möglichkeit den Cookie-Hinweis zu aktivieren.

Cookies (Session, SLT, CSRF)

Shopware speichert Cookies im Browser des Besuchers um die Grundfunktionen des Shops zu gewährleisten. Anhand der Cookies wird bspw. der Warenkorbinhalt, der Login-Zustand und auch der CSRF-Schutz ermöglicht. Ohne Cookies im Browser zugelassen zu haben, kann Shopware nicht benutzt werden. **WICHTIG:** Shopware speichert stets nur IDs im Browser des Kunden, die Zuweisung zu den jeweiligen Informationen erfolgen im Bereich der Applikation.

Session

Anhand des Session-Cookies entscheidet Shopware, ob der jeweilige User einen aktiven Warenkorb besitzt und ob der User eingeloggt ist. Er dient also als Identifikation zwischen Browser und Server. Es werden

keine weiteren Informationen bis auf die Session-ID im Browsers gespeichert. Der Umgang mit Sessions wird serverseitig über PHP gesteuert und ist unabhängig von Shopware zu sehen.

CSRF

Darüber hinaus erzeugt Shopware einen individuellen CSRF-Cookie bei Besuch des Shops, damit der Kunde die einzelnen Bereiche des Shops bedienen kann. Hier findest Du Informationen zum CSRF-Schutz: [CSRF-Protection](#)

SLT

Eines der neuen Features in 5.3 ist der SLT-Cookie der es dem Shop ermöglicht den Kunden bei Rückkehr zum Shop wiederzuerkennen, auch wenn die Session bereits ausgelaufen ist. Alle Informationen zum SLT-Cookie findest Du hier: [Shopware Login Token](#) Der SLT-Cookie kann in den Grundeinstellungen darüber hinaus auch deaktiviert werden.

Merkzettel

Legt ein Kunde ein Produkt auf den Merkzettel, wird hierzu ein Cookie mit dem Namen "sUniqueID" angelegt, um den Inhalt des Merkzettels zu speichern. Die gespeicherten Produkte werden in der Tabelle s_order_notes abgelegt.

Zuletzt angesehene Artikel

Im Local Storage des Browsers werden zudem die Informationen zu den "zuletzt angesehenen Artikeln" gespeichert. Hier findest Du die Dokumentation zu dieser Funktion: [Zuletzt angesehen](#)

Übersicht der Cookies

In dieser Tabelle findest Du für die Nutzung von Shopware relevante Cookies:

Name des Cookies	Funktion	Speicherdauer
__csrf_token-1	Wird für die Validierung von Kundenangaben benötigt.	Ist an die Browsersitzung gebunden.
session-1	Identifiziert die aktuelle Sitzung, den Benutzer und sein Warenkorb.	Ist an die Browsersitzung gebunden.
cookiePreferences	In einem serialisierten String werden die Informationen gespeichert, welche Cookies der Kunde wünscht.	Die Informationen werden für ein halbes Jahr gespeichert.

slt	Ermöglicht es, den Kunden bei Rückkehr zum Shop wiederzuerkennen, auch wenn die Session bereits ausgelaufen ist.	Wird für ein Jahr gespeichert oder durch manuelles Abmelden des Kunden gelöscht.
sUniqueID	Ist für die Zuordnung der Merkliste zuständig und wird beim Nutzen der Merkliste genutzt.	Wird für ein Jahr gespeichert.
x-ua-device	Dient zur Ermittlung des verwendeten Endgerätes bspw. für die korrekte Anzeige des Shops.	Ist an die Browsersitzung gebunden.
allowCookie	Speichert die Cookie-Einstellungen des Shopkunden.	Läuft nicht ab.
timezone	Die Zeitzone wird abgespeichert bspw. für Zeitangaben in E-Mails.	
PHPSESSID	PHP Session, die zur Zuordnung der aktuellen Sitzung dienen (Warenkorb, etc.).	Ist an die Browsersitzung gebunden.
csrf	Sämtliche CSRF-Cookies dienen dazu die Formulareingaben zu validieren, um XSS Attacken verhindern zu können.	Ist an die Browsersitzung gebunden.
sw-cache-hash	Technischer Cookie für die Cache-Funktionalität.	
cookie-preference	Für die Cookie-Präferenzen des Kunden, falls diese im Offcanvas konfiguriert werden.	Die Informationen werden für ein halbes Jahr gespeichert.
sw-states	Technischer Cookie für die Cache-Funktionalität.	

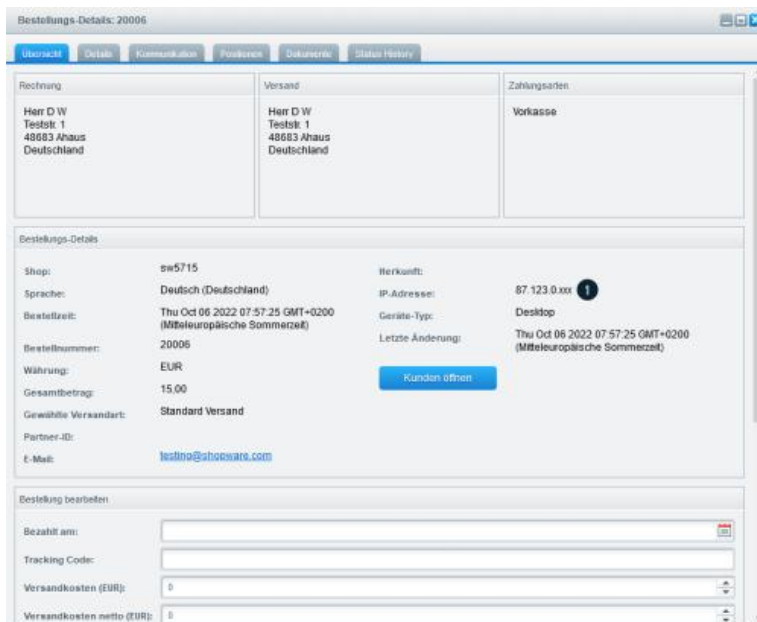
Welche IP-Adressen werden gespeichert?

Im Shopware Standard werden IP-Adressen hinterlegt. Dies dient zur eindeutigen Identifikation des Benutzers im Shop. Es werden im Standard von Shopware an vier Stellen IP-Adressen gespeichert.

s_order

In der Bestellübersicht findest Du die IP-Adresse der Bestellung. Dazu wechselst du im Backend auf **Kunden > Bestellungen** und rufst dort eine der Bestellungen auf.

Die IP-Adresse wird auf der **Bestellungs-Detail-Seite (1)** angezeigt. Die letzten 3 Stellen (das letzte Oktett) werden hierbei ausgeblendet.



Die IP-Adresse findest Du ebenfalls unverschlüsselt in der Datenbanktabelle `s_order` in der Spalte `remote_addr` wieder.

s_core_log

In der Tabelle `s_core_log` werden alle Aktivitäten von Backend-Benutzern hinterlegt. Anhand dieser Tabelle kann z.B. nachvollzogen werden wann Änderungen vorgenommen worden sind. Diese kannst Du dann ggf. einer Person anhand der IP-Adresse zuordnen.

s_statistics_pool

Die hier gesammelten Daten werden in den Statistiken eingesetzt, um die Besucher pro Tag zu ermitteln.

s_statistics_currentusers

In der Tabelle `s_statistics_currentusers` werden temporäre Daten für das Widget "Kunden online" gespeichert und alle 3 Minuten wieder geleert.

Wie kann ich die Datenschutzerklärung in meinen Shop einbinden?

Auch nach der DSGVO ist für Websites eine Datenschutzerklärung erforderlich. So ist nun über die zu Grunde gelegten Rechtsgrundlagen ebenso zu informieren, wie über die Erforderlichkeit für einen Vertragsabschluss oder ob eine andere Verpflichtung zur Bereitstellung der Daten besteht. Es ist zudem notwendig, dass Du in der Datenschutzerklärung darüber informierst, wie Du mit personenbezogenen Daten umgehst. Die Datenschutzerklärung kann dabei bspw. eine Shopseite sein, die Du in den relevanten Bereichen Deines Shops verlinkst. Wichtig hierbei ist, dass Du diese Informationen „bei Erhebung“ der personenbezogenen Daten gibst. Daher sollte die Datenschutzerklärung regelmäßig mit nur einem Klick erreichbar sein. Typischerweise erfolgt dies im Footer und auch im Checkout-Bereich. Hier findest Du die Dokumentation zu den Shopseiten in Shopware: [Shopseiten](#)

Die einzelnen Texte im Checkout oder überall da, wo Du personenbezogene Daten verarbeitest, kannst Du in der Regel über die Textbausteinverwaltung anpassen. Die Textbausteinverwaltung findest Du hier: [Textbausteine](#)

Zudem bieten auch die Formulare die Möglichkeit zur Einbindung einer zusätzlichen Checkbox: [Formulare - Beispiel Checkbox](#)

Wir weisen Dich darauf hin, dass die Sanktionen bei Nichteinhaltung dieser Informationspflichten gravierend sind. Fehlt eine Datenschutzerklärung, ist die Verarbeitung in der Regel rechtswidrig. Des Weiteren können Bußgelder verhängt werden, Schadensersatz oder ein Unterlassungsanspruch können geltend gemacht werden, sollte die Informationspflichten fehlerhaft erfüllt werden. Zudem können Verbraucherschutzverbände sowie Datenschutzvereine gegen einen Website-Betreiber ohne Datenschutzerklärung vorgehen. Letztlich können auch Abmahnungen durch Mitbewerber drohen.

Wie kann ich die Datenschutzerklärung per Checkbox in der Registrierung verlinken?

Du kannst einfach über die Grundeinstellungen > Storefront > Anmeldung/Registrierung die Option "Datenschutz-Bedingungen müssen über Checkbox akzeptiert werden" aktivieren. Im Registrierungsprozess findest Du dann eine Checkbox, die zusätzlich auf die Datenschutz-Bestimmungen verlinkt. Über den Textbaustein "RegisterLabelDataCheckbox" kannst Du den Text anpassen und auf die jeweilige Shopseite verlinken. Zwingend erforderlich ist eine separate Checkbox jedoch nicht. Es genügt auch ein Hinweis auf die Datenschutzinformationen im Bestellprozess (z.B. „Die Hinweise zum Datenschutz habe ich zur Kenntnis genommen.“), wenn die Datenschutzhinweise dort verlinkt sind.

Kann ich einen Datenschutzhinweis an allen relevanten Stellen einstellen?

Im Zuge der Einführung der DSGVO kann Shopware ab der Version 5.4.3 nun an allen relevanten Stellen einen Datenschutzhinweis anzeigen. Die zugehörige Einstellung findest Du in den **Grundeinstellungen** unter **Weitere Einstellungen > Datenschutz**.

Welche personenbezogenen Daten werden von Shopware an Dritte übertragen?

Im Standard von Shopware werden keine Informationen an Dritte übertragen. Durch Erweiterungen kann dies natürlich zusätzlich erfolgen. Setzt Du beispielsweise Paypal in Deinem Shop ein, werden Daten aus dem Shop (die Lieferadresse, die Bestellsumme und der Warenkorb) an Paypal übertragen. Hier gibt es natürlich auch zahlreiche weitere Dienstleister die Daten aus Shopware weiterverarbeiten, falls Du eine solche Erweiterung in Deinem Shop einsetzt. Prominente Beispiele wären hier Zahlungsanbieter, ERP-

Systeme und auch Newsletter-Dienstleister. Welche Daten hier an Dritte übertragen werden, erfragst Du am besten beim Hersteller der jeweiligen Erweiterung.

Wie kann ich personenbezogene Daten strukturiert ausgeben lassen?

Import/Export

Die DSGVO sieht vor, dass ein Kunde auf Anfrage die ihn betreffenden Daten strukturiert vom Shopbetreiber zur Verfügung gestellt bekommt. Shopware bietet hier die Import/Export-Funktion, mit der Du alle relevanten Daten eines Kunden exportieren kannst. Hierzu bietet das Modul bei den relevanten Profilen (Bestellungen, Kunden, Newsletter-Empfänger) die Möglichkeit einen Customer-Stream anzulegen. Du kannst hier bspw. vorab einen Customer-Stream mit dem betroffenen Kunden anlegen und einen Export erzeugen. Dabei solltest Du beachten, dass je nach Registrierung des Kunden (Schnellbesteller oder Kundenkonto) ggf. auch mehr als ein Kundenkonto bestehen kann. Eine Dokumentation über die Export-Möglichkeiten (CSV und XML) findest Du hier: [Import / Export](#)

Datenbank

Auch die Datenbank bietet die Möglichkeit, über ein SQL-Query die Informationen zu einem Kunden ausgeben zu lassen.

Wie kann ich alle personenbezogenen Daten aus meinem Shop löschen?

Alle in diesem Dokument definierten Daten können in der Regel komfortabel über das Backend gelöscht werden. Wenn ein Kunde nun also von Dir die Löschung aller personenbezogenen Daten wünscht, dann kannst Du diese Daten über das Bestell- und Kundenmodul im Backend löschen. Es werden automatisch auch die jeweiligen Verknüpfungen entfernt. Zudem kannst Du über das Newsletter-Modul den Empfänger auch aus der Empfängerliste entfernen. Shopware bietet also ohne Probleme die Möglichkeit, auf Kundenanfrage alle personenbezogenen Daten über das Backend zu entfernen. Ob und wann Du verpflichtet bist, Daten von Kunden zu löschen, ist eine Rechtsfrage, die Du mit Deinem Rechtsbeistand klären solltest. Pauschale Aussagen sind hierzu leider nicht möglich.

FAQ

Inwieweit wird Shopware auf die neue DSGVO vorbereitet?

Seit geraumer Zeit sind wir bei Shopware damit beschäftigt, in Zusammenarbeit mit den bekannten Zertifizierungsstellen sicherzustellen, dass das System den Anforderungen der im Mai wirksam werdenden DSGVO genügt. Dabei hat sich herausgestellt, dass Shopware nach unserem jetzigen Kenntnisstand den Shopbetreibern bereits heute die notwendigen Funktionen bereitstellt, die sie brauchen, um die nötigen Einstellungen vorzunehmen, die die Regeln der DSGVO erfordern. So stellt Shopware etwa bereits in der regulären Endbenutzer-Dokumentation alle erforderlichen Werkzeuge bereit, z.B. auch um personenbezogene Daten wieder aus dem System zu entfernen, was eine Kernforderung der neuen Datenschutz-Grundverordnung ist. Ein besonderes Plugin/Update ist bzgl. der DSGVO nicht geplant.

Werden entsprechende Funktionen automatisch mit SW Updates eingespielt?

Sollte sich die Notwendigkeit der Anpassung unserer Software ergeben, werden wir natürlich eine entsprechende Anpassung per Update bereitstellen.

Die Tutorials sind ab Shopware 5.4.3 hinfällig, da die Funktion ab diesem Zeitpunkt im Core von Shopware vorhanden ist. Hierzu findest du in den Grundeinstellungen den Punkt "Datenschutz".

Wie kann ich in der Newsletter-Registrierung auf die Datenschutz-Bestimmungen verweisen?

An allen Stellen die personenbezogene Daten verarbeiten, solltest Du einen Hinweis auf die Datenschutzbestimmungen platzieren. Hier findest Du ein kleines Tutorial, wie Du einen solchen Hinweis in Deinem Theme einbauen kannst. Als Basiswissen solltest Du dir das Tutorial zur Anpassung von Themes ansehen: [Tutorial - Änderungen am Template vornehmen](#)

Umsetzung

Für die Umsetzung der Änderungen musst Du in deinem Theme zwei kleine Änderungen durchführen:

Zunächst ergänzt Du in der Datei: **/themes/Frontend/DeinTheme/frontend/index/footer-navigation.tpl** folgenden Code:

```
{extends file="parent:frontend/index/footer-navigation.tpl"}
{block name="frontend_index_footer_column_newsletter_form"}
  {$smarty.block.parent}
  {s name="IndexFooterNewsletterPrivacy" die
  <a title="Datenschutzbestimmungen" href="{url controller=custom sCustom=7}">Datenschutzbestimmungen
{/block}
```

Im Anschluss ergänzt du noch in der Datei:

/themes/Frontend/DeinTheme/frontend/newsletter/index.tpl den nachfolgenden Code:

```
{extends file="parent:frontend/newsletter/index.tpl"}
{block name="frontend_newsletter_form_submit"}
  {s name="IndexFooterNewsletterPrivacy" die
  <a title="Datenschutzbestimmungen" href="{url controller=custom sCustom=7}">Datenschutzbestimmungen
  {$smarty.block.parent}
{/block}
```

Nachdem Du die Änderungen durchgeführt hast, musst Du einmal dein Theme kompilieren, damit die Änderungen im Frontend angezeigt werden.

Sofern die Datei in Deinem Theme noch nicht enthalten ist, musst Du diese im angegebenen Pfad selbst anlegen.

Mit dieser Anpassung wird ein neuer Textbaustein **IndexFooterNewsletterPrivacy** angelegt, der auch über das Backend bearbeitet werden kann. Die Verlinkung im Textbaustein verweist auf die Standard-Shopseite "Datenschutz", falls Du eine eigene Shopseite nutzt, muss der Link entsprechend angepasst werden.

Frontend nach der Anpassung



Der hinzugefügte Hinweis **(1)** unterhalb der Newsletter-Registrierung im Footer.

Vorname

Nachname

Straße und Nr.

PLZ Ort

* hierbei handelt es sich um ein Pflichtfeld

Die [Datenschutzbestimmungen](#) habe ich zur Kenntnis genommen **(1)**

Der Hinweis **(1)** am Ende des Newsletter-Formulars.

Wie kann ich in der Registrierung auf die Datenschutz-Bestimmungen verweisen?

Hier findest Du ein kleines Tutorial, wie Du einen solchen Hinweis in deinem Theme einbauen kannst. Als Basiswissen solltest Du dir das Tutorial zur Anpassung von Themes ansehen: [Tutorial - Änderungen am Template vornehmen](#)

Umsetzung

Für die Umsetzung der Änderungen musst Du in deinem Theme eine Änderungen durchführen:

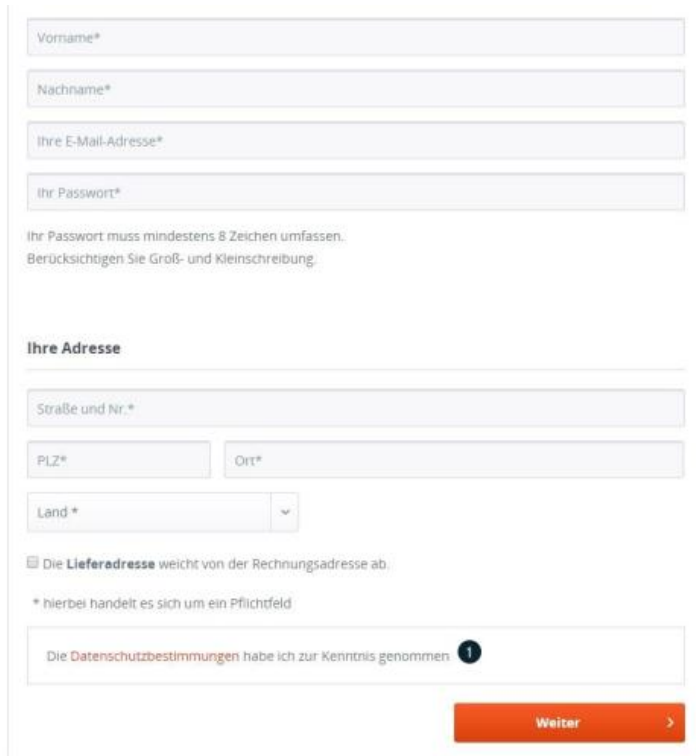
In der Datei: **/themes/Frontend/DeinTheme/frontend/register/index.tpl** ergänzt Du hierzu einfach den nachfolgenden Code. Ist die Datei in Deinem Theme noch nicht vorhanden, musst Du sie manuell anlegen:

```
{extends file="parent:frontend/register/index.tpl"}
{block name='frontend_register_index_form_submit'}
<div class="register--privacy">
  {s name="RegisterPrivacyDie"
  <a title="Datenschutzbestimmungen" href="{url controller=custom sCustom=7}">Datenschutzbestimmungen
  </div>
  {$smarty.block.parent}
{/block}
```

Alternativ kannst Du in den Grundeinstellungen auch die Checkbox für die Datenschutz-Bestimmungen unter **Einstellungen > Grundeinstellungen > Weitere Einstellungen > Datenschutz > Datenschutz-Bedingungen müssen über Checkbox akzeptiert werden:** aktivieren.

Mit dieser Anpassung wird ein neuer Textbaustein **RegisterPrivacy** angelegt, der auch über das Backend bearbeitet werden kann. Die Verlinkung im Textbaustein verweist auf die Standard-Shopseite "Datenschutz", falls Du eine eigene Shopseite nutzt, muss der Link entsprechend angepasst werden.

Frontend nach der Anpassung



The screenshot shows a registration form with the following fields: Vorname*, Nachname*, Ihre E-Mail-Adresse*, and Ihr Passwort*. Below the password field, there is a note: "Ihr Passwort muss mindestens 8 Zeichen umfassen. Berücksichtigen Sie Groß- und Kleinschreibung." The "Ihre Adresse" section includes fields for Straße und Nr.*, PLZ*, Ort*, and Land*. A checkbox is checked, with the text "Die Lieferadresse weicht von der Rechnungsadresse ab." Below this, a note states "* hierbei handelt es sich um ein Pflichtfeld". A new field at the bottom contains the text "Die Datenschutzbestimmungen habe ich zur Kenntnis genommen" followed by a small circular icon with the number 1. An orange "Weiter" button with a right-pointing arrow is at the bottom right.

Der nach den Änderungen neu hinzugefügte Hinweis **(1)**.

Wie kann ich in den Formularen auf die Datenschutz-Bestimmungen verweisen?

Hier findest Du ein kleines Tutorial, wie Du einen solchen Hinweis in Deinem Theme einbauen kannst. Als Basiswissen solltest Du dir das Tutorial zur Anpassung von Themes ansehen: [Tutorial - Änderungen am Template vornehmen](#)

Umsetzung

Für die Umsetzung der Änderungen musst Du in deinem Theme eine Änderungen durchführen:

In der Datei: **/themes/Frontend/DeinThemefrontend/forms/form-elements.tpl** ergänzt Du hierzu einfach den nachfolgenden Code. Ist die Datei in Deinem Theme noch nicht vorhanden, musst Du sie manuell anlegen:

```

{extends file="parent:frontend/forms/form-elements.tpl"}
{block name='frontend_forms_form_elements_form_submit'}
    {s name="SupportPrivacy"} Die
    <a title="Datenschutzbestimmungen" href="{url controller=custom sCustom=7}">Datenschutzbestimmungen
{/$smarty.block.parent}
{/block}

```

Nachdem Du die Änderungen durchgeführt hast, musst Du einmal dein Theme kompilieren, damit die Änderungen im Frontend angezeigt werden.

Mit dieser Anpassung wird ein neuer Textbaustein **SupportPrivacy** angelegt, der auch über das Backend bearbeitet werden kann. Die Verlinkung im Textbaustein verweist auf die Standard-Shopseite "Datenschutz", falls Du eine eigene Shopseite nutzt, muss der Link entsprechend angepasst werden.

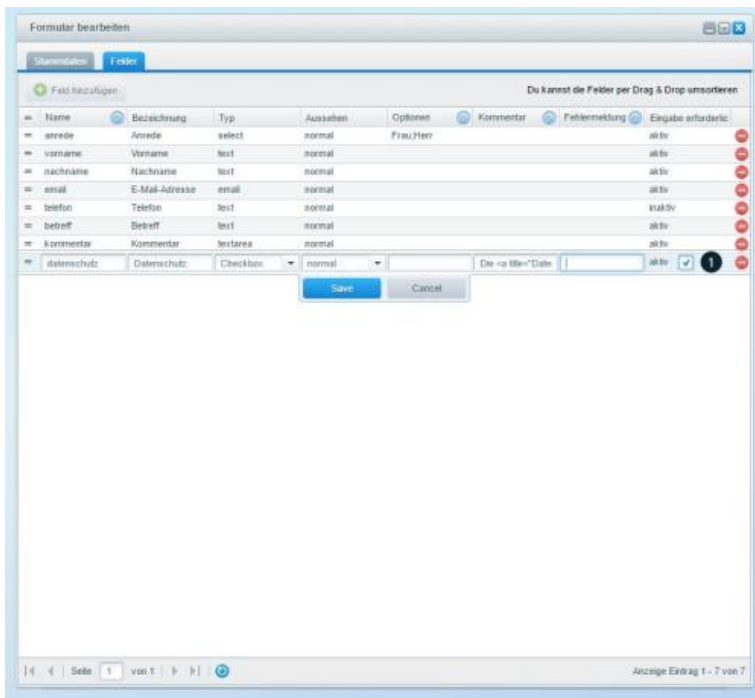
Frontend nach der Umsetzung

Der nach den Änderungen neu hinzugefügte Hinweis **(1)**.

Checkbox statt Hinweistext einblenden

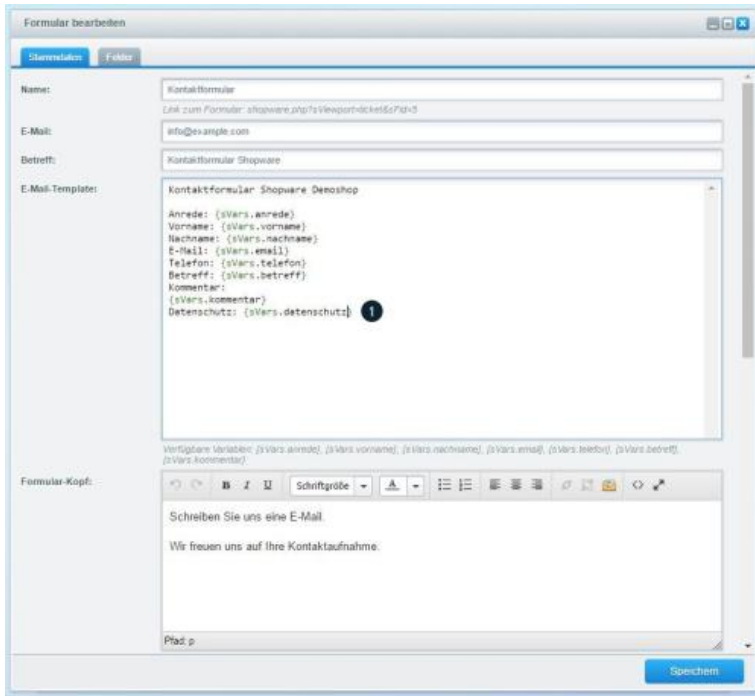
Alternativ kannst du eine Checkbox im Formular einbauen, welche angehakt werden muss, um das Formular abzusenden, dazu gehst Du wie folgt vor:

Erstelle das **Feld (1)** im Formular:



Wenn du möchtest, dass die Checkbox initial deaktiviert ist, dann trage bei "Optionen" eine "1" ein.

Zur Bestätigung kannst Du Dir den Wert der Eingabe auch in der eMail **ausgeben lassen (1)**:



Im Frontend sieht das **Ergebnis (1)** dann wie folgt aus:

Kontaktformular

Schreiben Sie uns eine E-Mail.

Wir freuen uns auf Ihre Kontaktaufnahme.

Kontaktformular

Anrede*

Vorname*

Nachname*

E-Mail-Adresse*

Telefon

Betreff*

Kommentar*

Datenschutz*
Die [Datenschutzbestimmungen](#) habe ich zur Kenntnis genommen.

The fields marked with * are required.

>

Wie kann ich in den Artikelbewertungen auf die Datenschutz-Bestimmungen verweisen?

An allen Stellen die personenbezogene Daten verarbeiten, solltest Du einen Hinweis auf die Datenschutzbestimmungen platzieren. Hier findest Du ein kleines Tutorial, wie Du einen solchen Hinweis in Deinem Theme einbauen kannst. Als Basiswissen solltest Du dir das Tutorial zur Anpassung von Themes ansehen: [Tutorial - Änderungen am Template vornehmen](#)

Textbaustein

Für die Umsetzung der Änderungen musst Du in deinem Theme eine kleine Änderungen durchführen:

Zunächst ergänzt Du in der Datei: **/themes/Frontend/DeinTheme/frontend/detail/comment/form.tpl** folgenden Code:

```
{extends file="parent:frontend/detail/comment/form.tpl"}
```

```
{block name='frontend_detail_comment_input_actions'}
```

```
  <p>{s name="CommentPrivacy"}Die
```

```
  <a title="Datenschutzbestimmungen" href="{url controller=custom sCustom=7}">Datenschutzbestimmungen
```

```
  {smarty.block.parent}
```

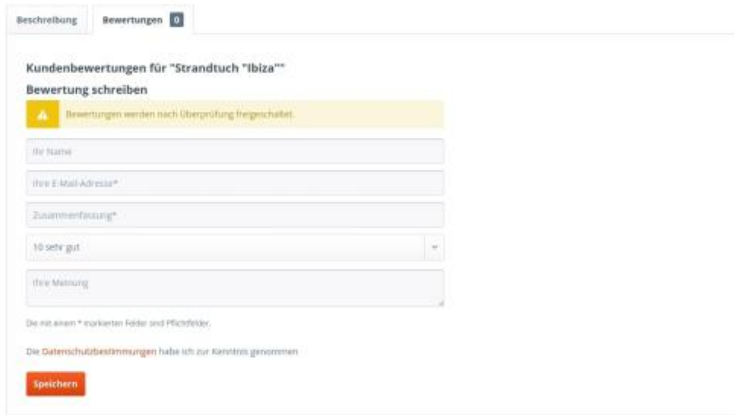
```
{/block}
```

Nachdem Du die Änderungen durchgeführt hast, musst Du einmal dein Theme kompilieren, damit die Änderungen im Frontend angezeigt werden.

Sofern die Datei in Deinem Theme noch nicht enthalten ist, musst Du diese im angegebenen Pfad selbst anlegen.

Mit dieser Anpassung wird ein neuer Textbaustein **CommentPrivacy** angelegt, der auch über das Backend bearbeitet werden kann. Die Verlinkung im Textbaustein verweist auf die Standard-Shopseite "Datenschutz", falls Du eine eigene Shopseite nutzt, muss der Link entsprechend angepasst werden.

Frontend nach der Anpassung



The screenshot shows a web interface for writing a customer review. At the top, there are tabs for 'Beschreibung' and 'Bewertungen'. Below the tabs, the title is 'Kundenbewertungen für "Strandtuch "ibiza""'. The main heading is 'Bewertung schreiben'. A yellow banner with a warning icon states 'Bewertungen werden nach Überprüfung freigeschaltet.'. The form includes input fields for 'Ihr Name', 'Ihre E-Mail-Adresse*', 'Zusammenfassung*', a dropdown menu for 'Ist sehr gut', and a text area for 'Ihre Meinung'. At the bottom, there is a 'Speichern' button and a note: 'Die mit einem * markierten Felder sind Pflichtfelder. Die Datenschutzbestimmungen habe ich zur Kenntnis genommen.'

Checkbox

Alternativ kannst du auch eine Checkbox einbauen. Hier ein Beispiel für die Checkbox:

Zunächst ergänzt Du in der Datei: **/themes/Frontend/DeinTheme/frontend/detail/comment/form.tpl** folgenden Code:


```
{extends file="parent:frontend/detail/comment/form.tpl"}
{block name='frontend_detail_comment_input_actions'}
  <p>
    <input name="comment-checkbox" type="checkbox" id="commentcheckbox" required="required" />
    <label for="commentcheckbox" class="chklable">{s name="CommentPrivacy"}Die
    <a title="Datenschutzbestimmungen" href="{url controller=custom sCustom=7}">Datenschutzbestim
  </p>
  { $smarty.block.parent }
{/block}
```

Frontend nach der Anpassung

Beschreibung | **Bewertungen** 0

Kundenbewertungen für "Strandtuch "Ibiza""

Bewertung schreiben

 Bewertungen werden nach Überprüfung freigeschaltet.

Ihr Name

Zusammenfassung*

10 sehr gut

Ihre Meinung

Die mit einem * markierten Felder sind Pflichtfelder.

Die [Datenschutzbestimmungen](#) habe ich zur Kenntnis genommen

Speichern

Tipps und Tricks

Unter gewissen Umständen, kann es notwendig sein einzelne personenbezogene Daten erst garnicht zu erheben. Möchtest du bspw. die E-Mail-Adresse des Kunden nicht speichern, reicht ein einfaches deaktivieren des Double-Opt-In Verfahrens. Dies kannst du in den Grundeinstellungen unter **Storefront > E-Mail-Einstellungen > Double-Opt-In für Artikelbewertungen** deaktivieren.

Wie kann ich in den Blog-Kommentaren auf die Datenschutz-Bestimmungen verweisen?

Hier findest Du ein kleines Tutorial, wie Du einen solchen Hinweis in Deinem Theme einbauen kannst. Als Basiswissen solltest Du dir das Tutorial zur Anpassung von Themes ansehen: [Tutorial - Änderungen am Template vornehmen](#)

Textbaustein

Für die Umsetzung der Änderungen musst Du in deinem Theme eine kleine Änderungen durchführen:

Zunächst ergänzt Du in der Datei: **/themes/Frontend/DeinTheme/frontend/blog/comment/form.tpl** folgenden Code:

```
{extends file="parent:frontend/blog/comment/form.tpl"}
{block name='frontend_blog_comments_input_submit'}
  <p>{s name="BlogPrivacy"}Die
  <a title="Datenschutzbestimmungen" href="{url controller=custom sCustom=7}">Datenschutzbestimmungen
  {&smarty.block.parent}
{/block}
```

Nachdem Du die Änderungen durchgeführt hast, musst Du einmal dein Theme kompilieren, damit die Änderungen im Frontend angezeigt werden.

Sofern die Datei in Deinem Theme noch nicht enthalten ist, musst Du diese im angegebenen Pfad selbst anlegen.

Mit dieser Anpassung wird ein neuer Textbaustein **BlogPrivacy** angelegt, der auch über das Backend bearbeitet werden kann. Die Verlinkung im Textbaustein verweist auf die Standard-Shopseite "Datenschutz", falls Du eine eigene Shopseite nutzt, muss der Link entsprechend angepasst werden.

Frontend nach der Anpassung



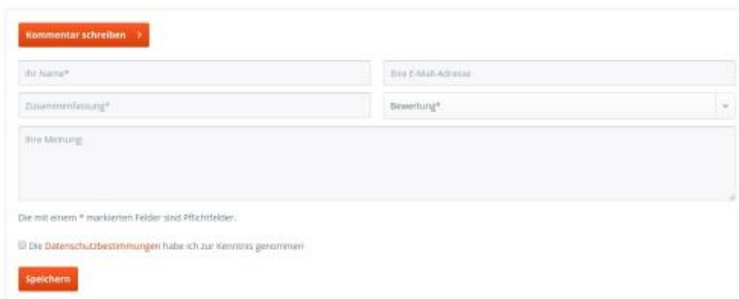
Checkbox

Alternativ kannst du auch eine Checkbox einbauen. Hier ein Beispiel für die Checkbox:

Zunächst ergänzt Du in der Datei: **/themes/Frontend/DeinTheme/frontend/blog/comment/form.tpl** folgenden Code:

```
{extends file="parent:frontend/blog/comment/form.tpl"}
{block name='frontend_blog_comments_input_submit'}
<p>
    <input name="blog-checkbox" type="checkbox" id="blogcheckbox" required="required" aria-re
    <label for="blogcheckbox" class="chklablel">{s name="CommentPrivacy"}Die
    <a title="Datenschutzbestimmungen" href="{url controller=custom sCustom=7}">Datenschutzbestim
    </p>
    {&smarty.block.parent}
{/block}
```

Frontend nach der Anpassung



Tipps und Tricks

Unter gewissen Umständen, kann es notwendig sein einzelne personenbezogene Daten erst garnicht zu erheben. Möchtest du bspw. die E-Mail-Adresse des Kunden bei Blog-Kommentaren nicht speichern, reicht ein einfaches deaktivieren des Double-Opt-In Verfahrens. Dies kannst du in den Grundeinstellungen unter **Storefront > E-Mail-Einstellungen > Double-Opt-In für Artikelbewertungen** deaktivieren.

Wie kann ich in der E-Mail-Benachrichtigung bei nicht verfügbaren Artikeln auf die Datenschutz-Bestimmungen verweisen?

An allen Stellen die personenbezogene Daten verarbeiten, solltest Du einen Hinweis auf die Datenschutzbestimmungen platzieren. Hier findest Du ein kleines Tutorial, wie Du einen solchen Hinweis in Deinem Theme einbauen kannst. Als Basiswissen solltest Du dir das Tutorial zur Anpassung von Themes ansehen: [Tutorial - Änderungen am Template vornehmen](#)

Textbaustein

Für die Umsetzung der Änderungen musst Du in deinem Theme eine kleine Änderungen durchführen:

Zunächst ergänzt Du in der Datei:

/themes/Frontend/DeinTheme/frontend/plugins/notification/index.tpl folgenden Code:

```
{extends file="parent:frontend/plugins/notification/index.tpl"}
{block name='frontend_detail_index_notification_button'}

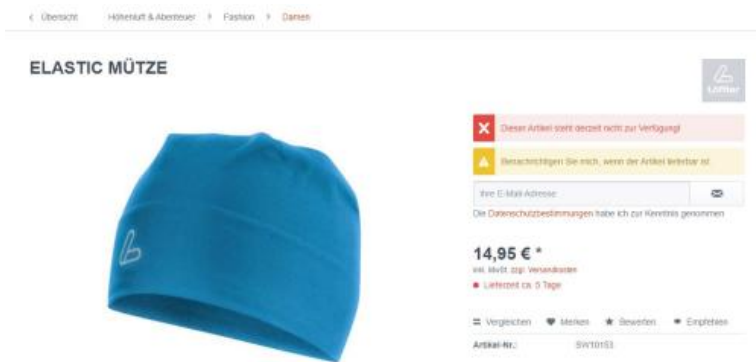
{$smarty.block.parent}
  <p>{s name="NotificationPrivacy"}Die
  <a title="Datenschutzbestimmungen" href="{url controller=custom sCustom=7}">Datenschutzbestimmungen
{/block}
```

Nachdem Du die Änderungen durchgeführt hast, musst Du einmal dein Theme kompilieren, damit die Änderungen im Frontend angezeigt werden.

Sofern die Datei in Deinem Theme noch nicht enthalten ist, musst Du diese im angegebenen Pfad selbst anlegen.

Mit dieser Anpassung wird ein neuer Textbaustein **NotificationPrivacy** angelegt, der auch über das Backend bearbeitet werden kann. Die Verlinkung im Textbaustein verweist auf die Standard-Shopseite "Datenschutz", falls Du eine eigene Shopseite nutzt, muss der Link entsprechend angepasst werden.

Frontend nach der Anpassung



Checkbox

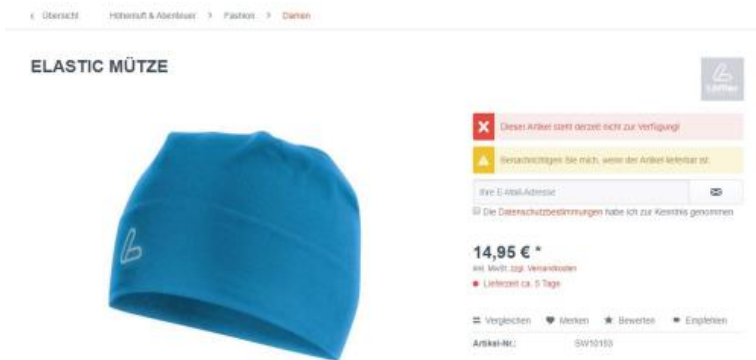
Alternativ kannst du auch eine Checkbox einbauen. Hier ein Beispiel für die Checkbox:

Zunächst ergänzt Du in der Datei:

/themes/Frontend/DeinTheme/frontend/plugins/notification/index.tpl folgenden Code:

```
{extends file="parent:frontend/plugins/notification/index.tpl"}
{block name='frontend_detail_index_notification_button'}
{$smarty.block.parent}
<p>
  <input name="notification-checkbox" type="checkbox" id="notificationcheckbox" required="r
  <label for="notificationcheckbox" class="chklable">{s name="NotificationPrivacy"} Die
  <a title="Datenschutzbestimmungen" href="{url controller=custom sCustom=7}">Datenschutzbestim
</p>
{/block}
```

Frontend nach der Anpassung



Tipps und Tricks

Unter gewissen Umständen, kann es notwendig sein einzelne personenbezogene Daten erst garnicht zu erheben. Möchtest du bspw. die E-Mail-Adresse des Kunden nicht speichern, reicht ein einfaches deaktivieren des Double-Opt-In Verfahrens. Diese Optionen kannst du in den Grundeinstellungen unter **Storefront > E-Mail-Einstellungen** einstellen.

Fehlermeldungen in Shopware debuggen

Auf dieser Seite findest Du einige hilfreiche Informationen, welche dir bei dem Verstehen von Fehlermeldungen und der Ermittlung der Ursache helfen werden.

Wie finde ich die Fehlermeldung?

Es kann an einigen Stellen durch vielfältige Gründe zu Fehlern kommen, die jedoch im Logfile nicht zu finden sind. Aber wie findest Du jetzt heraus, woran es jetzt scheitert?

Wenn ein schwerwiegender Fehler aufgetreten ist, merke Dir zuerst den Zeitpunkt, wann der Fehler auftrat! Danach schaue, mit welchen Schritten Du den Fehler reproduzieren kannst und überlege, ob der Fehler aus Shopware selbst oder einem Plugin kommen könnte. Danach begibst Du dich auf die Suche nach der Meldung.

Frontend Error Reporter

Der Frontend Error Reporter ist eine Möglichkeit, Dir schwerwiegende Fehler direkt im Frontend anzuzeigen, statt die "Es ist ein Fehler aufgetreten"-Seite anzuzeigen. Da das Template die Meldungen direkt ausgibt und so auch Deinen Kunden angezeigt wird, empfiehlt sich der Einsatz nur, wenn gerade wenig Traffic auf dem Shop ist oder Du in einer Entwicklungsumgebung arbeitest. Wie Du dieses erweiterte Error Debugging aktivierst, erklären wir Dir unter [Erweitertes Error-Debugging](#).

Log Dateien

Sollte irgendwo im Frontend ein schwerwiegender Fehler auftreten, wird dieser ins Logfile auf dem Server geschrieben, diese Datei findest Du unter `/var/log/` auf deinem Server, ein Beispiel wäre hier zum Beispiel **core_production-2016-02-01.log**. Je nachdem, in welcher Umgebung deine Installation betrieben wird (Live, Testsystem) oder von wo der Fehler kommt (Core, Plugin) kann die Datei auch anders heißen.

Wichtig hierbei ist vor allem das Datum. Es wird dabei für jeden Tag eine neue Datei angelegt. Wenn dein Fehler wahrscheinlich aus einem Plugin kommt, dürfte er in einer "plugin_production-DATUM.log" stehen, wenn er eher aus Shopware selbst kommt, steht er in einer "core_production-DATUM.log". Öffne die Datei und suche nach dem Zeitpunkt, zu dem der Fehler auftauchte (ca. minutengenau), hier solltest Du nun eine Fehlermeldung finden, die Dir den Fehler beschreibt. Wie Du diese Meldungen deutest, erklären wir noch.

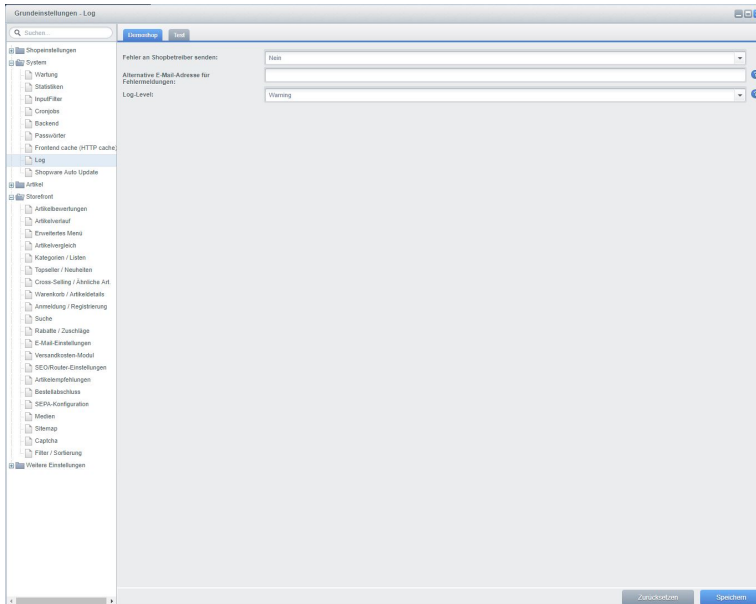
System-Log im Backend

Hierzu findest Du unter Einstellungen > Logfile den neuen Reiter namens "System-Log" (1). Mit einem Klick auf das Lupensymbol (2) erhältst Du detaillierte Informationen zu den einzelnen Meldungen. Im Bereich "Datei" (3) kannst Du zwischen den einzelnen Log-Dateien wechseln und auch gezielt Log-Dateien herunterladen, wenn Du diese beispielsweise weiterleiten möchtest.

Datum	Kanal	Level	Meldung	Code
08.02.2017 08:55:54	core	ERROR	Shopware/Components/CSRFTokenValidationException: The provided X-CSRF-Token for path 7...	
08.02.2017 08:55:52	core	ERROR	Shopware/Components/CSRFTokenValidationException: The provided X-CSRF-Token for path 7...	
08.02.2017 08:55:49	core	ERROR	Shopware/Components/CSRFTokenValidationException: The provided X-CSRF-Token for path 7...	
08.02.2017 08:55:34	core	ERROR	Shopware/Components/CSRFTokenValidationException: The provided X-CSRF-Token for path 7...	
08.02.2017 08:55:32	core	ERROR	Shopware/Components/CSRFTokenValidationException: The provided X-CSRF-Token for path 7...	
08.02.2017 08:55:29	core	ERROR	Shopware/Components/CSRFTokenValidationException: The provided X-CSRF-Token for path 7...	
08.02.2017 08:55:14	core	ERROR	Shopware/Components/CSRFTokenValidationException: The provided X-CSRF-Token for path 7...	
08.02.2017 08:55:12	core	ERROR	Shopware/Components/CSRFTokenValidationException: The provided X-CSRF-Token for path 7...	
08.02.2017 08:55:11	core	ERROR	Shopware/Components/CSRFTokenValidationException: The provided X-CSRF-Token for path 7...	
08.02.2017 08:55:09	core	ERROR	Shopware/Components/CSRFTokenValidationException: The provided X-CSRF-Token for path 7...	
08.02.2017 08:54:54	core	ERROR	Shopware/Components/CSRFTokenValidationException: The provided X-CSRF-Token for path 7...	
08.02.2017 08:54:52	core	ERROR	Shopware/Components/CSRFTokenValidationException: The provided X-CSRF-Token for path 7...	
08.02.2017 08:54:49	core	ERROR	Shopware/Components/CSRFTokenValidationException: The provided X-CSRF-Token for path 7...	
08.02.2017 08:54:31	core	ERROR	Shopware/Components/CSRFTokenValidationException: The provided X-CSRF-Token for path 7...	
08.02.2017 08:54:27	core	ERROR	Shopware/Components/CSRFTokenValidationException: The provided X-CSRF-Token for path 7...	
08.02.2017 08:54:11	core	ERROR	Shopware/Components/CSRFTokenValidationException: The provided X-CSRF-Token for path 7...	
08.02.2017 08:53:11	core	ERROR	Shopware/Components/CSRFTokenValidationException: The provided X-CSRF-Token for path 7...	
08.02.2017 08:52:11	core	ERROR	Shopware/Components/CSRFTokenValidationException: The provided X-CSRF-Token for path 7...	
08.02.2017 08:51:11	core	ERROR	Shopware/Components/CSRFTokenValidationException: The provided X-CSRF-Token for path 7...	
08.02.2017 08:50:11	core	ERROR	Shopware/Components/CSRFTokenValidationException: The provided X-CSRF-Token for path 7...	
08.02.2017 08:50:05	core	ERROR	Could not send user registration email to address asd@asd.de	
08.02.2017 08:49:11	core	ERROR	Shopware/Components/CSRFTokenValidationException: The provided X-CSRF-Token for path 7...	
08.02.2017 08:48:11	core	ERROR	Shopware/Components/CSRFTokenValidationException: The provided X-CSRF-Token for path 7...	
08.02.2017 08:47:11	core	ERROR	Shopware/Components/CSRFTokenValidationException: The provided X-CSRF-Token for path 7...	
08.02.2017 08:46:11	core	ERROR	Shopware/Components/CSRFTokenValidationException: The provided X-CSRF-Token for path 7...	

Fehlermeldungen per E-Mail senden lassen

In den Grundeinstellungen unter *System* > *Log* hast Du die Möglichkeit, Dir Fehlermeldungen per E-Mail senden zu lassen.



Diese Einstellung sendet Dir sämtliche Fehler im Shop per E-Mail zu, die auftreten, auch wenn diese den normalen Betrieb nicht stören. Daher ist es nicht sinnvoll, sich im Produktivmodus die Fehler per E-Mail senden zu lassen.

Wie lese ich Fehlermeldungen?

Wir erklären dir hier allgemein, wie Fehlermeldungen aus Shopware analysiert und verstanden werden können.

Bereich	Hinweise zur Analyse
Stack trace	Hier findest Du Informationen zu den zuletzt aufgerufenen Funktionen. So kannst Du herausfinden, an welcher Stelle eine Funktion aufgerufen wurde, die zum Fehler führte. Häufig werden hier auch Plugin-Namen mit aufgeführt. Dieses Plugins sollten dann der erste Ansatzpunkt sein, wenn Du in die Fehleranalyse gehst.
Time	Diese Information bietet die Option weiter einzugrenzen, ob es regelmäßig oder ggf. zu geplanten Operationen zu diesem Verhalten kommt.
uri	Hier wird die aufgerufene URL angezeigt. Mit einem manuellen Aufruf der URL kannst Du eingrenzen, ob beispielsweise "nur" der Aufruf einer nicht (mehr) existenten URL, beispielsweise durch einen Bot, die Fehlermeldung hervorgerufen hat. Anderenfalls kann der Ort des Auftretens näher bestimmt werden.
query	Liefert weitere Informationen, ob das Verhalten im Frontend/Backend auftritt, welcher Controller angesprochen wurde, etc.
shopId / shopName	Wenn Sprach- und/oder Subshops eingesetzt werden, kannst Du hier sehen, in welchem Shop der Fehler auftrat.

PHP Fehlermeldungen

PHP-Fehler sind etwas schwerer zu debuggen als die anderen, da Du zumeist nur einen "503" oder "500" vom Server zurück bekommst und die Meldung nicht direkt sehen kannst. Um die die Fehler auszugeben, kannst Du ins Error-Log-Verzeichnis deines Servers schauen. Falls Du nicht weißt, wo Du das Verzeichnis findest, frag deinen Hoster, dieser kann Dir die Antwort geben. Danach gibt es 2 Wege:

- Schau einmal die Logs durch, taucht für deinen Fehler keine Meldung auf oder es wird nichts geloggt, bitte deinen Hoster, das "Debug-Level" zu erhöhen, damit auch "unkritischere" Fehler geloggt werden. Danach solltest Du eine Ausgabe bekommen und kannst debuggen.
- Wenn Du über das nötige technische Wissen verfügst, kannst du, sofern Du weißt, wo der Fehler im Quellcode auftritt, an dieser Stelle ein "die" mit Fehlerausgabe einbauen und schauen, was Dir für Fehler angezeigt werden. Vergiss aber nicht, die Datei nach dem debuggen wieder in den Ursprungszustand zu versetzen, damit deine Besucher keine Fehlerseite mit detaillierter Fehlerausgabe sehen.

Template Fehlermeldungen

Template Fehler werden Dir je nach Fehler entweder mittels Error Reporter direkt im Frontend, oder aber in der Log-Datei ausgegeben und sehen so aus:

```
[2016-02-22 09:43:20] core.ERROR: exception 'SmartyException' with message 'Unable to load tem
Stack trace:
#0 /home/shopware/www/sw513/engine/Library/Smarty/sysplugins/smarty_internal_template.php(286)
#1 /home/shopware/www/sw513/var/cache/production_201602150844/templates/frontend_Responsive_de
#2 /home/shopware/www/sw513/engine/Library/Smarty/sysplugins/smarty_internal_templatebase.php(
```

```
#3 /home/shopware/www/sw513/engine/Library/Enlight/View/Default.php(274): Smarty_Internal_Temp
#4 /home/shopware/www/sw513/engine/Library/Enlight/Controller/Plugins/ViewRenderer/Bootstrap.p
#5 /home/shopware/www/sw513/engine/Library/Enlight/Controller/Plugins/ViewRenderer/Bootstrap.p
#6 /home/shopware/www/sw513/engine/Library/Enlight/Controller/Plugins/ViewRenderer/Bootstrap.p
#7 [internal function]: Enlight_Controller_Plugins_ViewRenderer_Bootstrap->onPostDispatch(Obje
#8 /home/shopware/www/sw513/engine/Library/Enlight/Event/Handler/Default.php(91): call_user_fu
#9 /home/shopware/www/sw513/engine/Library/Enlight/Event/EventManager.php(210): Enlight_Event_
#10 /home/shopware/www/sw513/engine/Library/Enlight/Controller/Action.php(201): Enlight_Event_
#11 /home/shopware/www/sw513/engine/Library/Enlight/Controller/Dispatcher/Default.php(523): En
#12 /home/shopware/www/sw513/engine/Library/Enlight/Controller/Front.php(226): Enlight_Control
#13 /home/shopware/www/sw513/engine/Shopware/Kernel.php(153): Enlight_Controller_Front->dispat
#14 /home/shopware/www/sw513/vendor/symfony/http-kernel/HttpCache/HttpCache.php(492): Shopware
#15 /home/shopware/www/sw513/engine/Shopware/Components/HttpCache/AppCache.php(255): Symfony\C
#16 /home/shopware/www/sw513/vendor/symfony/http-kernel/HttpCache/HttpCache.php(449): Shopware
#17 /home/shopware/www/sw513/vendor/symfony/http-kernel/HttpCache/HttpCache.php(349): Symfony\C
#18 /home/shopware/www/sw513/engine/Shopware/Components/HttpCache/AppCache.php(178): Symfony\C
#19 /home/shopware/www/sw513/vendor/symfony/http-kernel/HttpCache/HttpCache.php(213): Shopware
#20 /home/shopware/www/sw513/engine/Shopware/Components/HttpCache/AppCache.php(114): Symfony\C
#21 /home/shopware/www/sw513/shopware.php(101): Shopware\Components\HttpCache\AppCache->handle
#22 {main} [] {"uid": "51a7455"}
```

Die Meldung sagt aus, dass ein Template Snippet nicht gefunden werden konnte. In 'frontend/index' wird irgendwo versucht 'frontend/index/headers.tpl' zu laden, welche es aber nicht gibt (oder fehlende Dateirechte aufweist). Prüfe also erst einmal, ob die Datei überhaupt vorhanden ist und ob diese die passenden Rechte besitzt. Falls die Datei nicht vorhanden ist, gibt es 2 Wege: *Fehlt die Datei?* oder *Wird hier die falsche Datei aufgerufen?*

Den Fehler kannst Du ausbügeln, in dem Du hier im Beispiel alle Dateien unter 'frontend/index/' öffnest und schaust, wo eine "headers.tpl" inkludiert (eingebunden) wird. Hast Du die Datei gefunden (Im Beispiel 'frontend/index/index.tpl'), vergleiche diese Datei mit der selben Datei des Standardtemplates. Wird im Standardtemplate die gleiche Datei inkludiert, fehlt diese und Du musst die Datei wieder beschaffen. Wird die Datei im Standard nicht inkludiert, wird diese Datei in deinem Template falsch abgefragt. Im Beispiel liegt hier ein Schreibfehler vor und statt *header.tpl* wird *headers.tpl* inkludiert. Du musst also die Datei, in der die falsche Datei inkludiert wird, ändern und wieder speichern. Danach wird Dir der Fehler nicht mehr angezeigt.

MySQL Fehlermeldungen

MySQL Fehler können Dir zum Beispiel beim Arbeiten im Backend begegnen und diverse Ursachen haben. Merke dir, was Du gemacht hast, als Du diesen Fehler bekommen hast, zum Beispiel einen Artikel oder Eigenschaften gespeichert.

```
exception 'PDOException' with message 'SQLSTATE[23000]: Integrity constraint violation: 1062 D
```

Duplicate entry '185-38' besagt, dass versucht wird, einen Wert mit 185-38 zu speichern, den es aber schon gibt. Das kann daran liegen, dass mehrere Benutzer im Backend gerade exakt das gleiche tun (Beide versuchen im gleichen Moment, eine neue Eigenschaft anzulegen) oder es passiert ein Fehler beim Neu-Installieren eines Plugins (Beim Deinstallieren wird etwas nicht aus der Datenbank entfernt, was beim Install aber wieder hinzugefügt werden soll). So kannst Du den Fehler eingrenzen und in der Datenbank schauen, welcher Eintrag hier falsch ist und diesen ggf. händisch löschen.

Wenn Du Änderungen an der Datenbank vornimmst, mach vorher unbedingt ein Backup! Änderungen an der Datenbank sind immer mit absoluter Vorsicht zu genießen, tu das nur, wenn Du dich damit auskennst!

Bekannte Fehlermeldungen und deren Ursache

In diesem Abschnitt findest Du Fehlermeldungen, die uns aus Support-Anfragen und der Community bekannt sind. Passend dazu gibt es auch mögliche Lösungen bzw. Ursachen.

Allgemein oder "Ups! Ein Fehler ist aufgetreten"

DemoShop

Home

Ups! Ein Fehler ist aufgetreten!

Wir wurden bereits über das Problem informiert und arbeiten an einer Lösung, bitte versuchen Sie es in Kürze erneut.

"Ups! Ein Fehler ist aufgetreten!" wird im Frontend in den meisten Fällen angezeigt, wenn ein sogenannter 503 ("Service Unavailable") Fehler vorhanden ist. Hierbei handelt es sich um einen Skriptfehler, der durch verschiedene Faktoren ausgelöst werden kann. In vielen Fällen sorgen Plugins oder inkonsistente Datensätze in der Datenbank für Fehlermeldungen. Das [erweiterte Debugging](#) ist daher unerlässlich für das weitere Vorgehen, da Du ja ersteinmal wissen musst, was genau der Fehler ist bzw. woher er kommt.

The provided X-CSRF-Token is invalid

DemoShop

Home

Ups! Ein Fehler ist aufgetreten!

Die nachfolgenden Hinweise sollten Ihnen weiterhelfen.

The provided X-CSRF-Token for path "/5220/account/login/Target/account/s/TargetAction/index" is invalid. Please go back, reload the page and try again. In engine/Shopware/Components/CSRFTokenValidator.php on line 158

Stack trace:

```
#0 [internal function]: Shopware\Components\CSRFTokenValidator->checkFrontendTokenValidation(Object(Enlight_Controller_ActionEventArgs))
#1 engine/Library/Enlight/Event/Handler/Default.php(93): call_user_func_array(Object(Enlight_Controller_ActionEventArgs))
#2 engine/Library/Enlight/Event/EventManager.php(214): Enlight_Event_Handler_Default->execute(Object(Enlight_Controller_ActionEventArgs))
#3 engine/Library/Enlight/Controller/Action.php(143): Enlight_Event_EventManager->notify('Enlight_Controller_Action', Object(Enlight_Controller_ActionEventArgs))
#4 engine/Library/Enlight/Controller/Dispatcher/Default.php(23): Enlight_Controller_Action->dispatch('login/index')
#5 engine/Library/Enlight/Controller/Front.php(233): Enlight_Controller_Dispatcher_Default->dispatch(Object(Enlight_Controller_Request_RequestHttp), Object(Shopware_Kernel.php(188): Enlight_Controller_Event->dispatch())
#6 vendor/symfony/http-kernel/HttpCache/HttpCache.php(487): Shopware_Kernel->handle(Object(Symfony\Component\HttpFoundation\Request), 1, true)
#7 engine/Shopware/Components/HttpCache/AppCache.php(255): Symfony\Component\HttpKernel\HttpCache\HttpCache->forward(Object(Symfony\Component\HttpFoundation\Request))
#8 vendor/symfony/http-kernel/HttpCache/HttpCache.php(158): Shopware\Components\HttpCache\AppCache->forward(Object(Symfony\Component\HttpFoundation\Request))
#9 vendor/symfony/http-kernel/HttpCache/HttpCache.php(275): Symfony\Component\HttpKernel\HttpCache\HttpCache->pass(Object(Symfony\Component\HttpFoundation\Request))
#10 engine/Shopware/Components/HttpCache/AppCache.php(133): Symfony\Component\HttpKernel\HttpCache\HttpCache->invalidate(Object(Symfony\Component\HttpFoundation\Request))
#11 vendor/symfony/http-kernel/HttpCache/HttpCache.php(286): Shopware\Components\HttpCache\AppCache->invalidate(Object(Symfony\Component\HttpFoundation\Request))
#12 engine/Shopware/Components/HttpCache/AppCache.php(114): Symfony\Component\HttpKernel\HttpCache\HttpCache->handle(Object(Symfony\Component\HttpFoundation\Request))
#13 Shopware.php(137): Shopware\Components\HttpCache\AppCache->handle(Object(Symfony\Component\HttpFoundation\Request))
#14 {main}
```

exception 'Shopware\Components\CSRFTokenValidationException' with message 'The provided X-CSRF-Token is invalid'

Stack trace:

```
#0 [internal function]: Shopware\Components\CSRFTokenValidator->checkFrontendTokenValidation(Object(Enlight_Controller_ActionEventArgs))
#1 /var/www/vhosts/domain.com/httpdocs/shopware/engine/Library/Enlight/Event/Handler/Default.php(93): call_user_func_array(Object(Enlight_Controller_ActionEventArgs))
#2 /var/www/vhosts/domain.com/httpdocs/shopware/engine/Library/Enlight/Event/EventManager.php(214): Enlight_Event_Handler_Default->execute(Object(Enlight_Controller_ActionEventArgs))
#3 /var/www/vhosts/domain.com/httpdocs/shopware/engine/Library/Enlight/Controller/Action.php(143): Enlight_Event_EventManager->notify('Enlight_Controller_Action', Object(Enlight_Controller_ActionEventArgs))
#4 /var/www/vhosts/domain.com/httpdocs/shopware/engine/Library/Enlight/Controller/Dispatcher/Default.php(23): Enlight_Controller_Action->dispatch('login/index')
#5 /var/www/vhosts/domain.com/httpdocs/shopware/engine/Library/Enlight/Controller/Front.php(233): Enlight_Controller_Dispatcher_Default->dispatch(Object(Enlight_Controller_Request_RequestHttp), Object(Shopware_Kernel.php(176): Enlight_Controller_Event->dispatch())
#6 /var/www/vhosts/domain.com/httpdocs/shopware/engine/Shopware/Kernel.php(176): Enlight_Controller_Event->dispatch()
#7 /var/www/vhosts/domain.com/httpdocs/shopware/vendor/symfony/http-kernel/HttpCache/HttpCache.php(487): Shopware_Kernel->handle(Object(Symfony\Component\HttpFoundation\Request), 1, true)
#8 /var/www/vhosts/domain.com/httpdocs/shopware/engine/Shopware/Components/HttpCache/AppCache.php(255): Symfony\Component\HttpKernel\HttpCache\HttpCache->forward(Object(Symfony\Component\HttpFoundation\Request))
#9 /var/www/vhosts/domain.com/httpdocs/shopware/vendor/symfony/http-kernel/HttpCache/HttpCache.php(158): Shopware\Components\HttpCache\AppCache->forward(Object(Symfony\Component\HttpFoundation\Request))
#10 /var/www/vhosts/domain.com/httpdocs/shopware/vendor/symfony/http-kernel/HttpCache/HttpCache.php(275): Symfony\Component\HttpKernel\HttpCache\HttpCache->pass(Object(Symfony\Component\HttpFoundation\Request))
#11 /var/www/vhosts/domain.com/httpdocs/shopware/engine/Shopware/Components/HttpCache/AppCache.php(133): Symfony\Component\HttpKernel\HttpCache\HttpCache->invalidate(Object(Symfony\Component\HttpFoundation\Request))
#12 /var/www/vhosts/domain.com/httpdocs/shopware/engine/Shopware/Components/HttpCache/AppCache.php(286): Shopware\Components\HttpCache\AppCache->invalidate(Object(Symfony\Component\HttpFoundation\Request))
#13 /var/www/vhosts/domain.com/httpdocs/shopware/engine/Shopware/Components/HttpCache/AppCache.php(114): Symfony\Component\HttpKernel\HttpCache\HttpCache->handle(Object(Symfony\Component\HttpFoundation\Request))
#14 /var/www/vhosts/domain.com/httpdocs/shopware/shopware.php(101): Shopware\Components\HttpCache\AppCache->handle(Object(Symfony\Component\HttpFoundation\Request))
#15 {main}
```

Die Bedeutung

Seit Shopware 5.2.0 verwenden wir zur Erhöhung der Sicherheit den sogenannten X-CSRF-Token (cross site request forgery). Damit möchten wir verhindern, dass vom Kunden nicht eigenwillig durchgeführte Aktionen ausgeführt werden. Dafür gleichen wir beim Austausch von Daten einen Sicherheits-Token ab. Wird der Token nicht richtig generiert, ist falsch oder gar nicht vorhanden, erscheint eine Fehlermeldung, die über den invaliden Token informiert. Das Eingrenzen der Fehler ist hierbei in vielen Fällen schwierig, da Du zu aller erst ein reproduzierbares Muster brauchst, um den Fehler "auf Abruf" nachzustellen, da nur so die Beseitigung möglich ist. Es kann aber auch passieren, dass der X-CSRF-Token zwar Ursache ist, aber nicht der entsprechende Fehler angezeigt wird, daher solltest du zusätzlich folgende Dinge prüfen:

- **Individuelles Theme:** Prüfe, ob das Verhalten auch im Standard Responsive Theme auftaucht. Änderungen bei der Übergabe der Formulardaten o.ä. können einen invaliden CSRF-Token hervorrufen.
- **Plugins:** Deaktiviere zunächst alle Plugins in Deinem Shop und führe im Anschluss den Workflow zum Reproduzieren des Fehlers durch. Sollte dieser nicht mehr auftauchen können die Plugins schrittweise aktiviert werden. Wenn dann nach Aktivierung eines Plugins der Fehler wieder auftritt, ist die Ursache zumeist das Plugin.
- **Updates:** Verwende stets die aktuellste Shopware Version, da wir regelmäßig Verbesserungen in diesem Bereich durchführen.

Deaktivieren des X-CSRF-Tokens

Wenn "alle Stricke reißen" empfiehlt es sich, den Token-Schutz temporär zu deaktivieren. In manchen Fällen können "normale" Fehlermeldungen hierdurch verschleiert werden, die im Anschluss analysiert werden sollten. Das Deaktivieren des Schutzes bedeutet nicht automatisch, dass Dein Shop jetzt völlig schutzlos ist. Hier bietet Shopware weitere Sicherheitsmaßnahmen, die auch weiterhin greifen - für eine optimale Sicherheit sollte man die Ursache für einen invaliden Token aber stets herausfinden. Unter [CSRF-Schutz deaktivieren](#) findest du das passende Konfigurations-Schnipsel.

Integrity constraint violation

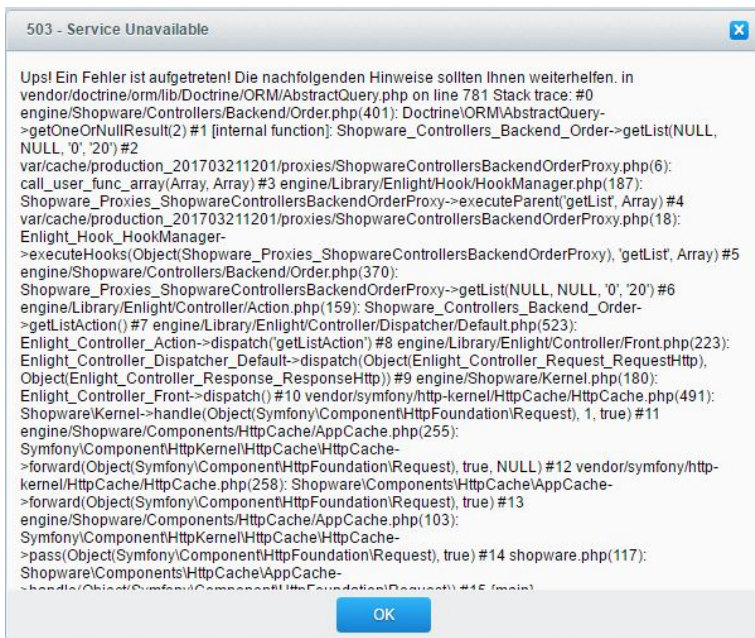
Die Bedeutung

Diese SQL Fehlermeldung deutet auf ein Problem mit den Datensätzen in der Datenbank hin, die durch defekte **Foreign Keys** oder **doppelte Einträge** hervorgerufen werden.

Möglichkeiten zum Beheben

Zunächst ist es ratsam eine [Foreign Key Reparatur](#) durchzuführen, dadurch werden die Beziehungen zwischen den Standard Core-Tabellen neu eingepflegt - beachte jedoch, dass Plugin Tabellen davon nicht betroffen sind! Sollte in der Fehlermeldung auf einen **"Duplicate Entry"** hingewiesen werden, empfiehlt es sich die betroffene Tabelle auf doppelte Einträge zu prüfen und diese Einträge zu löschen.

Bestellübersicht lässt sich nicht mehr öffnen



Ups! Ein Fehler ist aufgetreten! Die nachfolgenden Hinweise sollten Ihnen weiterhelfen. in vendor/doctrine/orm/lib/Doctrine/ORM/AbstractQuery.php on line 781 Stack trace: #0 engine/Shopware/Controllers/Backend/Order.php(1038): Doctrine\ORM\AbstractQuery->getOneOrNullResult(2) #1 var/cache/production_2017042110836/proxies/ShopwareControllersBackendOrderProxy.php(6): Shopware_Proxies_ShopwareControllersBackendOrderProxy->getList(NULL, NULL, 0, 20) #2 engine/Library/Enlight/Hook/HookManager.php(186): Shopware_Proxies_ShopwareControllersBackendOrderProxy->executeParent('getList', Array) #3 var/cache/production_2017042110836/proxies/ShopwareControllersBackendOrderProxy.php(38): Enlight_Hook_HookManager->executeHooks(Object(Shopware_Proxies_ShopwareControllersBackendOrderProxy), 'getList', Array) #4 engine/Shopware/Controllers/Backend/Order.php(271): Shopware_Proxies_ShopwareControllersBackendOrderProxy->getListAction() #5 engine/Library/Enlight/Controller/Action.php(159): Shopware\Controllers\Backend\Order->getListAction() #6 engine/Library/Enlight/Controller/Dispatcher/Default.php(523): Enlight_Controller_Action->dispatch('getListAction') #7 engine/Library/Enlight/Controller/Front.php(223): Enlight_Controller_Dispatcher_Default->dispatch(Object(Enlight_Controller_Request_RequestHttp), Object(Enlight_Controller_Response_ResponseHttp)) #8 engine/Shopware/Kernel.php(182): Enlight_Controller_Front->dispatch() #9 vendor/symfony/http-kernel/HttpCache/HttpCache.php(491): Shopware\Kernel->handle(Object(Enlight_Controller_Request_RequestHttp), 1, true) #10 engine/Shopware/Components/HttpCache/AppCache.php(266): Symfony\Component\HttpKernel\HttpCache\HttpCache->forward(Object(Symfony\Component\HttpFoundation\Request), true, NULL) #11 vendor/symfony/http-kernel/HttpCache/HttpCache.php(258): Shopware\Components\HttpCache\AppCache->forward(Object(Symfony\Component\HttpFoundation\Request), true) #12 engine/Shopware/Components/HttpCache/AppCache.php(103): Symfony\Component\HttpKernel\HttpCache\HttpCache->pass(Object(Symfony\Component\HttpFoundation\Request), true) #13 shopware.php(117): Shopware\Components\HttpCache\AppCache->handle(Object(Symfony\Component\HttpFoundation\Request)) #15 [main]

Die Fehlermeldung

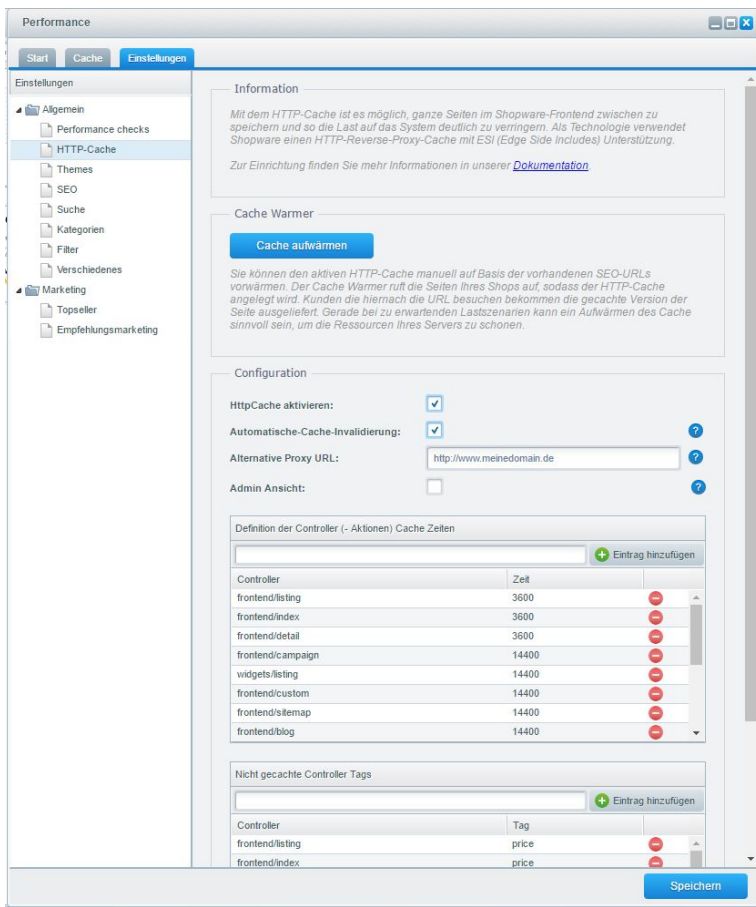
In den meisten Fällen existieren hier mehrere Bestellungen mit gleicher Bestellnummer - in Shopware darf eine Bestellnummer nur einmal vorkommen, weshalb es im Standard auch nicht möglich ist eine solche doppelt anzulegen. Schnittstellen zu anderen Verkaufsportalen oder andere Plugins können diesen Mechanismus in manchen Fällen aushebeln, falls dies der Fall ist, setze Dich mit dem Anbieter der jeweiligen Schnittstelle in Verbindung.

Die Behebung

Hierfür empfiehlt sich unser [Beispiel SQL-Statement](#), welches automatisch doppelte Bestellnummern ermittelt. Diese müssen im Anschluss bereinigt werden und die Funktionalität ist wiederhergestellt.

Einkaufswelten-Modul lässt sich nicht öffnen

Die Fehlermeldung



Hier ist zunächst die Ursache beim Hoster zu suchen. In vielen Fällen reicht es jedoch bereits aus, die Shop-URL unter **Einstellungen > Caches/Performance > Einstellungen > HTTP Cache** unter "Alternative Proxy URL" zu hinterlegen. Es kann jedoch auch ein externer Proxy durch den Hoster genutzt werden, weshalb du hier im Zweifel direkt den Hoster kontaktieren solltest.

Tweaks für die config.php

Die config.php ist die erste Konfiguration, die Shopware bei der Ausführung lädt. Durch das Hinzufügen von neuen Konfigurations-Snippets lassen sich viele nützliche Tools freischalten, die dir unter anderem auch beim Debugging helfen. Du kannst dir alle verfügbaren Konfigurationen in der "Default Config" auf [Github](#) anschauen.

Erweitertes Error-Debugging

Den Error Reporter aktivierst Du in der config.php durch hinzufügen der folgenden Zeilen:

```
'front' => array(
    'showException' => true,
    'noErrorHandler' => false,
),
//Low-Level PHP-Fehler ab Shopware 5.2.0
'phpsettings' => [
    'display_errors' => 1,
]
```

Deine config.php sieht dann in etwa so aus:

```
<?php return array (
    'db' =>
        array (
            'host' => 'HOST',
            'port' => 'PORT',
            'username' => 'USER',
            'password' => 'PASSWORD',
            'dbname' => 'DBNAME',
        )
    , 'front' => array(
        'showException' => true,
        'noErrorHandler' => false,
    )
    , //Low-Level PHP-Fehler ab Shopware 5.2.0
    'phpsettings' => [
        'display_errors' => 1,
    ]
);
```

Nachdem Du die Datei gespeichert hast, werden die Fehler direkt im Frontend ausgegeben.

HTTP-Cache Debugging aktivieren

Manchmal möchte man überprüfen, ob die Seiten die man aufruft auch wirklich im Cache sind. Hier bietet sich folgendes Snippet an, welches den Symfony Debug-Cache aktiviert.

```
// Http-Cache
'httpCache' => [
    'enabled' => true, // true or false
    'debug' => true,
],
```

Öffnest du nun die [Entwickler-Konsole](#) deines Browsers und schaust dir unter "Netzwerk" die Antwort des initialen Document-Request an, siehst du dort folgendes:

```
X-Content-Digest:enldac22cb10b9a66cb6590a7819384d54fcb637e838f65c40b655b07c85cd5f7d
X-Shopware-Allow-Nocache:price-1
X-Shopware-Cache-Id:/c3;
X-Symfony-Cache: GET /: miss, store;
    GET /?action=shopMenu&controller=index&module=widgets: fresh;
    GET /?controller=compare&module=widgets: miss, store;
    GET /?action=menu&controller=index&group=gLeft&module=widgets: miss, store;
    GET /?action=info&controller=checkout&module=widgets: miss, store
```

- **miss** = Keine Cache-Datei vorhanden
- **fresh** = Controller / Seite liegt im Cache

CSRF-Schutz deaktivieren

Falls es zu Invaliden Tokens im Frontend kommt, kann es hilfreich sein, temporär diese Funktion zu deaktivieren.

Füge folgende Code-Zeilen in die config.php ein:

```
'csrfProtection' => [
    'frontend' => false,
    'backend' => false
]
```

Versendete Mails als Datei auf dem Server speichern

Hat man Probleme mit dem Versand von Mails, weiß man erstmal nicht wo man anfangen soll. Durch das Aktivieren der Mail-File Config sorgst Du dafür, dass alle Mails als Datei gespeichert werden.

```
'mail' => [
    'type' => 'file'
],
```

Ziel-Pfad festlegen:

```
'mail' => [
    'type' => 'file',
    'path' => $this->DocPath().'mails'
],
```

Beispiel Konfiguration für Entwickler

Hier deaktivieren wir sämtliche Caches, sowie den Template-Cache. Dadurch sind Änderungen direkt sichtbar, ohne das Theme konfigurieren zu müssen.

Wende diese Konfiguration nicht in deinem Produktivsystem an, da die Performance dadurch massiv beeinflusst wird.

```
'front' => [
    'showException' => true,
    'throwExceptions' => true,
    'noErrorHandler' => false,
],
//Zeige Low-Level PHP-Fehler
'phpsettings' => [
    'display_errors' => 1,
],
// Template-Cache
'template' => [
    'forceCompile' => true,
],
// Backend-Cache
'cache' => [
    'backend' => 'Black-Hole',
    'backendOptions' => [],
    'frontendOptions' => [
        'write_control' => false
    ],
],
// Model-Cache
'model' => [
    'cacheProvider' => 'Array' // supports Apc, Array, Wincache and Xcache
],
// Http-Cache
'httpCache' => [
    'enabled' => true, // true or false
    'debug' => true,
],
```

Ich benötige Unterstützung bei der Ermittlung der Ursache

Shopware bietet dir hier verschiedene Möglichkeiten um Unterstützung bei der Analyse einer Fehlermeldung zu erhalten. Primär unterstützen wir Dich gerne im Technischen Support sofern Du eine kostenpflichtige [Software-Subscription](#) für Deine kommerzielle Shopware Version gebucht hast. Hier profitierst Du vom Service des Herstellers mit gesicherten Reaktionszeiten. Alternativ kannst Du Dich auch an unsere Community im [Forum](#) oder an einen unserer zahlreichen [Shopware Partner](#) wenden.

Foreign Keys reparieren

In bestimmten Konstellationen und auch nur bei manchen Hostern, kann es vorkommen, dass die Foreign-Keys, die Shopware in der Datenbank benötigt, verloren gehen. Die Modelgenerierung der Attributmodels benötigt diese Keys, damit die Attribute korrekt erstellt werden können.

Dies kann zu Fehlermeldungen führen, ähnlich wie Diese:

```
[Semantical Error] line 0, col 114 near 'billing WHERE': Error: Class Shopware\Models\Attribut  
Stack trace: ...
```

Gehe hier bitte wie folgt vor.

- 1. Erstelle ein Backup der Datenbank.**
- 2. Lade die Datei unter "Verfügbare Downloads" auf dieser Seite herunter.**
- 3. Entpacke die heruntergeladene Datei auf Deinem lokalen System**
- 4. Wähle im PHPMysqlAdmin Deine Shopware Datenbank aus. Auf der rechten Seite findest Du den Reiter "Import". Über diese Funktion kannst Du eine "zu importierende Datei" auswählen. Wähle die heruntergeladene Datei vom lokalen Rechner aus. Bestätige diesen Vorgang mit (Okay) auf der Seite.**
- 5. Lösche nun via FTP den Cache, indem Du den Ordner "production_XXXX" (x = Datum des verwendeten Builds) im Ordner "var/cache" löschst.**
- 6. Überprüfe die Dateirechte der Ordner. Diese brauchen vollen Lese- und Schreibzugriff**
- 7. Rufe das Frontend im Browser erneut auf.**

Fertig

Fehlerbehandlung

...CONSTRAINT FAILED... FOREIGN KEY (`basketID`)...

Falls beim Ausführen der SQL Datei eine ähnliche Fehlermeldung wie diese erhalten, befinden sich korrupte Daten in der Datenbank:

MySQL meldet:

```
#1452 - Cannot add or update a child row: a foreign key constraint fails (`result 2 when expla
```

Diese kannst Du mit folgendem Statement entfernen. Führe das Statement in phpMyAdmin aus:

```
DELETE FROM s_order_basket_attributes WHERE basketID NOT IN (  
    SELECT id FROM s_order_basket  
);
```

Starte nun wieder bei Schritt 4 und führe die SQL Datei erneut aus.

...CONSTRAINT FAILED... FOREIGN KEY (`categoryID`)...

Falls beim Ausführen der SQL Datei eine ähnliche Fehlermeldung wie diese erhalten, befinden sich korrupte Daten in der Datenbank:

MySql meldet:

```
#1452 - Cannot add or update a child row: a foreign key constraint fails (`****`. `#sql-ac5_2d1
```

Diese kannst Du mit folgendem Statement entfernen. Führe das Statement in phpMyAdmin aus:

```
DELETE ca -- SELECT ca.*  
FROM `s_categories_attributes` ca  
LEFT JOIN s_categories c  
ON c.id = ca.categoryID  
WHERE c.id IS NULL
```

Wenn Du die Meldungen bei einer anderen Tabelle bekommst, musst Du das Query anpassen.

Beispiel:

```
DELETE ca -- SELECT ca.*  
FROM `s_MY_TABLE_attributes` ca  
LEFT JOIN s_MY_TABLE c  
ON c.id = ca.categoryID  
WHERE c.id IS NULL
```

...CONSTRAINED FAILED... FOREIGN KEY ('articleID')...

mySql meldet:

```
#1452 - Cannot add or update a child row:  
a foreign key constraint fails
```

```
(`xxxxxx`.`xxxxxxxxxxx`,  
CONSTRAINT `#xxxxxxxxxxxxx`  
FOREIGN KEY (`articleID`)  
REFERENCES `s_articles` (`id`)  
ON DELETE CASCADE ON UPDATE NO ACTION)
```

Lösung:

```
DELETE ca -- SELECT ca.*  
FROM `s_articles_attributes` ca  
LEFT JOIN s_articles c  
ON c.id = ca.articleID  
WHERE c.id IS NULL
```

Verfügbare Downloads

Vor der Reparatur sollte ein Backup der Datenbank angelegt werden.

Die aktuelle Datei ist ab **Shopware 5.6.2** kompatibel. Achte darauf, dass du bei einer älteren Version auch die ältere Foreign Key Reparatur verwendest.

Aktuelle Foreign Keys Reparatur

[Download Foreign Keys ab Shopware 5.6.2](#)

Foreign Key Reparatur für ältere Shopware Versionen

[Download Foreign Keys bis 5.2.13](#)

[Download Download Foreign Keys ab 5.2.13](#)

[Download Foreign Keys ab Shopware 5.5](#)

Installation / Update per Shell

Shopware lässt sich nicht nur per Web-Oberfläche installieren und updaten. Du kannst diese Schritte auch per Shell vornehmen, um z.B. automatisiert Installationen auszurollen und zu aktualisieren. Im folgenden Artikel erklären wir, wie das funktioniert.

Installation

Installationspaket hochladen und entpacken

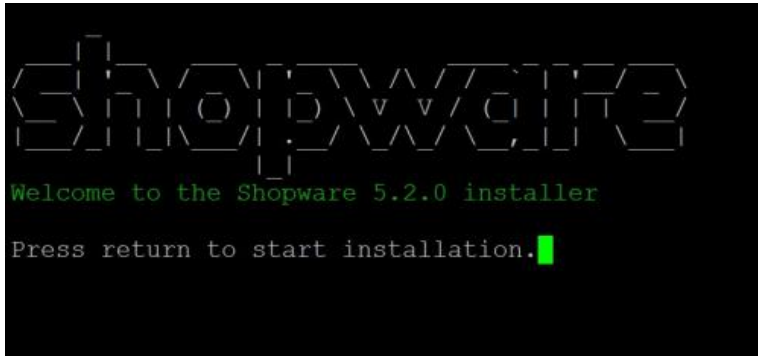
Zuerst müssen die Dateien vorhanden sein. Lade dazu wie gewohnt das Shopware Installationpaket auf Deinen Server und entpacke das Paket. Das selbe gilt für die Datenbank. Falls diese nicht schon vorhanden ist, erstelle eine neue Datenbank.

Installation via Wizard

Die Installation beginnt mit dem folgenden Befehl (davon ausgehend, dass Du Dich im Shopware-Verzeichnis befindest):

```
// Installation starten
php recovery/install/index.php
// Hilfe zur Installation erhalten
php recovery/install/index.php --help
```

Drücke ENTER, um die Installation zu starten:



Zuerst geht's um die Datenbank, hier werden die folgenden Daten abgefragt:

Um die Vorschläge (in Klammern) zu bestätigen, bestätige mit ENTER, oder gib Deine entsprechenden Werte hier ein. Bei Bedarf kannst Du auch eine neue Datenbank erzeugen, falls Dein Server dies zulässt.

Nach Eingabe der Daten startet die Installation der Datenbank:

Nachdem die Datenbank installiert ist, folgen Grundinformationen wie Lokalisierung, Shop Host, Pfad, Name, E-Mail & Währung:

Danach geht's mit den Backend Admin-Daten weiter:

Nachdem der Admin-User angelegt ist, wirst Du nach der Lizenz gefragt. Wählst Du hier "ce" wird die Installation fortgesetzt, wählst Du "cm", wirst Du aufgefordert, eine Lizenz einzutragen:

Nach der Eingabe der Lizenz ist die Installation abgeschlossen.

One-Line Installation

Um die Installation zu beschleunigen oder zu automatisieren, ist ein Wizard eher hinderlich. Daher zeigen wir Dir einmal, wie eine Installation mit nur einem Befehl aussehen kann:

```
php recovery/install/index.php --no-interaction --quiet --no-skip-import --db-host="HOST" --db-
```

"--no-interaction" steht hierbei für "keine Interaktion". Der Installer fragt hier also nicht nach, sondern nutzt die gegebenen Daten. "--quiet" Installiert Shopware komplett im Hintergrund ohne Rückmeldung. "--no-skip-import" importiert die Datenbank, auch wenn bereits Daten darin enthalten sind.

Bei den Lokalisierungen kannst Du im Standard zwischen den folgenden wählen:

- de_DE => Deutschland
- en_GB => Vereinigtes Königreich

Bei den Währungen stehen Dir die folgenden Werte zur Auswahl:

- EUR => Euro
- USD => US Dollar
- GBP => Britische Pfund

Update

Updatepaket hochladen und entpacken

Lade das Updatepaket hoch und entpacke dieses im Shopwareverzeichnis. Die vorhandenen Dateien müssen dabei immer überschrieben werden.

Update per Wizard

Das Wizard-Update wird mit dem folgenden Befehl angestoßen:

```
php recovery/update/index.php
```

Der Updater öffnet sich:

Mit ENTER startest Du das Update:

Nachdem das Update abgeschlossen ist, meldet der Updater, dass die /update-assets noch gelöscht werden müssen, damit der Wartungsmodus des Shops entfernt wird:

Die /update-assets kannst Du mit dem folgenden Befehl löschen:

```
// update-assets löschen  
rm -r update-assets/
```

One-Line Update

Auch das Update kann automatisiert ausgeführt werden, hierzu stehen die folgenden Attribute zur Auswahl:

- --quiet => Verarbeitung im Hintergrund
- --no-interaction => Keine Interaktion

Der Befehl zum Update lautet dann:

```
php recovery/update/index.php --no-interaction --quiet && rm -r update-assets/
```

Internationalisierung

Internationalisierung ist ein wichtiges Thema, um auch Zielgruppen in anderen Ländern zu bedienen. Dieser Guide hilft Dir Deinen Shop zu internationalisieren, indem wir Dir zeigen, auf welche Dinge Du achten musst, wenn Du gut vorbereitet sein willst.

Wo will ich überhaupt hin? / Was gibt es dort zu beachten?

Bevor Du die ersten Schritte der Internationalisierung in Shopware machst, überlege Dir zuerst, wo die Reise hin gehen soll und vor allem, was es dort alles zu beachten gilt. Hier ein paar Beispiele:

- Welche Zahlarten werden im Ausland bevorzugt genutzt, brauche ich ggf. eine neue Schnittstelle, um Zahlungen aus dem Ausland abzuwickeln?
- Wie sieht die Steuersituation im Ausland aus, gibt es hier Sonderfälle, die zu beachten sind (Beispielsweise: Ist Kinderbekleidung im UK steuerfrei)?
- Muss ich abweichende Währungen einsetzen und falls ja, werden diese von meinen bisherigen Payment-Anbietern unterstützt?
- Übersetzung: Gibt es fertige Übersetzungen oder muss ich selbst übersetzen (lassen)?
- Will ich Kunden aus dem Ausland mit eigenen Preisen versorgen?
- Kann ich ins Ausland liefern oder benötige ich hier neue Schnittstellen?
- Wie hoch sollen Lieferkosten ins Ausland sein?

Ergänzend haben wir für Dich einen kostenlosen [Cross Border Guide](#) erstellt, der Dir Tipps und Infos für Deinen internationalen eComemrce-Erfolg liefert. Die besten Tools, um mit Deinem Shop international zu verkaufen, findest Du hier: [Shopware Store - Die besten Tools um international zu verkaufen](#)

Let's internationalize!

Länder aktiv schalten

Der erste Schritt ist natürlich die Registrierung von Kunden aus den entsprechenden Ländern zu ermöglichen, indem Du die Länder in der [Länderverwaltung](#) aktivierst. Achte hier vor allem darauf, dass die Länder den richtigen Länder-Zonen zugewiesen sind, da dies später für die Steuerregeln wichtig wird.

Kategorien anlegen

Die Kategorien kannst Du im Backend unter **Artikel > Kategorien** mittels der Weltkugel direkt für Deine Sprachen übersetzen.

Währungen anlegen

Um ins Ausland zu verkaufen, brauchst Du unter Umständen auch andere Währungen, wie Du diese konfigurierst, erklären wir im [Artikel "Währungen"](#).

Steuern

Da im Ausland unter Umständen auch anders versteuert wird, musst Du Deine Steuerregeln für die jeweiligen Länder einrichten, wie das geht, zeigen wir im [Artikel "Steuern"](#).

Shop anlegen

Je nachdem, ob Du einen Sprachshop oder einen Subshop für Deine internationalen Kunden nutzen willst, musst Du zuerst den Shop anlegen. Wie das geht, erfährst Du hier:

[Tutorial: Einrichten von Sprach- & Subshops](#)

Achte hier besonders auf die korrekte Angabe der Lokalisierung und ggf. des Punktes "Übersetzungen übernehmen von" sowie die korrekte Kategoriezuweisung.

Abweichende Konfiguration für Deinen Shop

Eventuell willst Du Deinen Shop auch für das Ausland anders konfigurieren, zum Beispiel ein anderes Adressformat nutzen oder Deine Payment-Schnittstelle braucht für Deine ausländischen Kunden andere Zugangsdaten. Auch diese Dinge solltest Du berücksichtigen und an den entsprechenden Stellen konfigurieren. Die abweichende Konfiguration ist dann verfügbar, sobald Du einen neuen Shop erstellst und ist dann als Tab mit dem Shopnamen erkennbar, in dem Du separate Einstellungen treffen kannst:

Mein deutscher Shop | Mein englischer Shop

Name des Shops: Demoshop

Shopbetreiber E-Mail: info@shopware.com

Adresse: shopware AG
Ebbinghoff 10
48624 Schöppingen
Deutschland

Bankverbindung: Musterbank Musterstadt
IBAN: DE12 3456 7890 1234 5678 90
BIC: ABCDFFG1HUI

Einstellungen aus Dropdown-Feldern können vom Hauptshop vererbt oder gesondert eingestellt werden.

Übersetzungen

Textbausteine (Snippets)

Shopware liefert im Standard die Textbausteine in Deutsch und Englisch aus. Bei Bedarf kannst Du diese natürlich noch erweitern, indem Du entweder selbst übersetzt oder ein [Sprachplugin](#) einsetzt.

Default / de_DE | Default / en_GB | Mein englischer Shop / en_GB

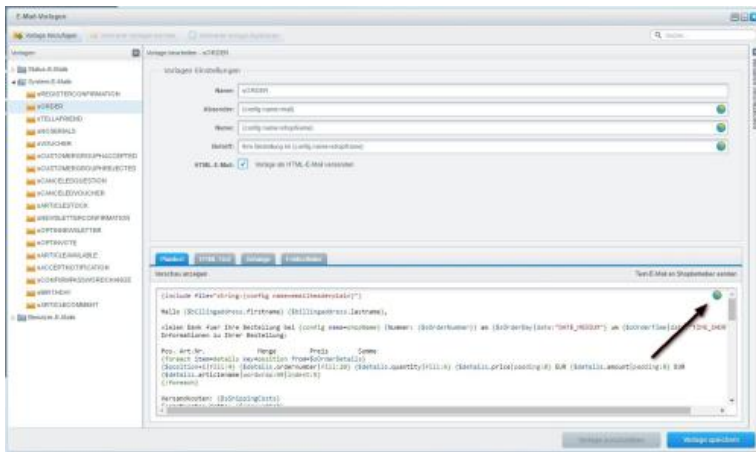
Nur leere Textbausteine zeigen | Markierte Textbausteine bearbeiten | Textbausteine hinzufügen | Suche

Namespace	Name	Wert
backend/activate/skeleton	WindowTitle	Cache leeren
backend/analytics/view/main	chart/country/sales	Verkäufe
backend/analytics/view/main	chart/country/sum	Gesamtverkäufe
backend/analytics/view/main	chart/dispatch/sales	Verkäufe
backend/analytics/view/main	chart/dispatch/sum	Gesamtverkäufe

Wusstest du, dass Du über [Crowdin](#) jederzeit Änderungen an den bestehenden Textbausteinen vorschlagen kannst und Shopware auch in weitere Sprachen übersetzen kannst? Schau einfach mal vorbei: [Shopware auf Crowdin](#)

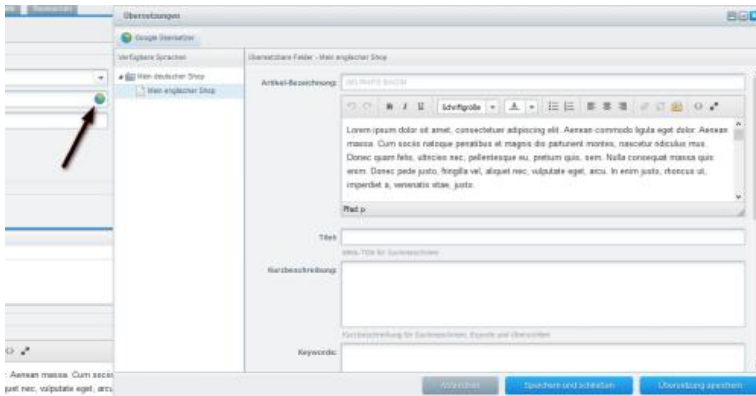
E-Mails

Um international erfolgreich zu sein, müssen natürlich auch Deine E-Mails korrekt übersetzt werden. Dies kannst Du direkt in Deinen E-Mail-Vorlagen unter "Einstellungen > E-Mail-Vorlagen" tun, indem Du die gewünschte E-Mail-Vorlage öffnest, die jeweilige Vorlage (Plaintext oder HTML-Text) öffnest und hier im Textbereich auf den kleinen Globus klickst. Hier kannst Du nun wie in allen anderen Modulen auch deine E-Mails übersetzen.



Artikel übersetzen

Natürlich musst Du neben Deinen Textbausteinen auch Deine Artikel übersetzen, dies kannst du, indem Du in den zur Übersetzung nötigen Feldern auf die blaue Weltkugel oben rechts am Feld klickst.



In dem folgenden Popup Fenster kannst Du dann die Übersetzung für Deinen Sprach- oder Subshop eingeben und diese speichern.

Folgende artikelbezogenen Daten sind übersetzbar:

- Artikelname
- Beschreibung
- Meta-Informationen
- ggf. Freitextfelder
- Eigenschaftssets / -gruppen / -optionen
- Meta Informationen in Bildern
- Variantengruppen / -optionen

Testen

Wenn alle notwendigen Dinge eingestellt sind, kannst Du Deinen internationalisierten Shop einem gründlichen Test unterziehen. Hier solltest Du schauen, ob alle Übersetzungen richtig geladen werden, alle Textbausteine passen, alle Einstellungen greifen und die Versand- und Zahlungsregeln korrekt funktionieren. Wenn alles klappt, hast Du Deinen Shop erfolgreich internationalisiert.

Nachträgliche Änderung des MwSt.-Satzes

Wir stellen zusätzlich das Plugin [MwSt. Änderung / Senkung](#) in unserem Community Store bereit, über das Du die notwendigen Anpassungen für die MwSt.-Änderung vornehmen kannst.

Solltest Du die Anpassungen manuell vornehmen wollen, erklären wir Dir in diesem Tutorial, was Du bei der nachträglichen Änderung des MwSt.-Satzes beachten solltest.

Grundsätzlich empfehlen wir bei allen Änderungen in diesem Bereich zunächst, ein Backup der Datenbank anzulegen! So hast Du jederzeit die Möglichkeit, zu dem jetzigen Stand zurückzukehren.

Steuersatz anpassen

Im ersten Schritt solltest Du Dir darüber klar sein, dass sich der Preis des Artikels ändert, wenn Du den Steuersatz anpasst. Du kannst nun selber entscheiden, ob Du den Bruttopreis anpassen möchtest und der Nettopreis des Artikels der gleiche bleibt, oder ob Du den Nettopreis anpassen möchtest, sodass der Bruttopreis für den Kunden der gleiche bleibt.

In den Grundeinstellungen unter Artikel > sonstige MwSt.-Sätze kannst Du eigene MwSt.-Sätze für Gutscheine und Rabatte definieren.

Bruttopreis anpassen (Variante 1)

Wenn der Nettopreis des Artikels der gleiche bleiben soll, und Du somit den Bruttopreis anpassen möchtest, reicht es einfach den Steuersatz im Backend zu ändern. Hierzu öffnest Du den Menüpunkt **Einstellungen > Grundeinstellungen > Shopeinstellungen**

Details

Name: 19%

Standard-Steuersatz: 19

Steuerregeln

Name	Region	Land	Bundesland	Steuersatz
+ Hinzufügen				Shopkunden

Keine Einträge

Ändere hier den Standard-Steuersatz auf die gewünschte Steuerhöhe (Bsp. 16%). Da Shopware in der Datenbank die Nettopreise speichert, werden nun im Frontend und im Backend automatisch die neuen Bruttopreise angezeigt und berechnet.

In der Bestellübersicht werden im Reiter Positionen in der Spalte Steuern der Name des ausgewählten Steuersatzes angezeigt.

Wenn Du den Namen des Steuersatzes anpasst, werden auch in älteren Bestellungen unter Steuern der Name des Steuersatzes angezeigt, wie er jetzt heißt. (Bspw. 16%)

Bruttopreis anpassen (Variante 2)

Eine weitere Möglichkeit, die Steuern so anzupassen, dass der Nettopreis erhalten bleibt, ist es, einen komplett neuen Steuersatz anzulegen. Das kannst Du im Backend unter Einstellungen > Grundeinstellungen > Shopeinstellungen > Steuern tun.

Anschließend sorgst Du durch folgendes SQL Query dafür, dass den gewünschten Artikeln der neue Steuersatz zugeordnet wird.

```
UPDATE `s_articles` SET s_articles.`taxID` = '2' WHERE `s_articles`.taxID='1';
```

In diesem Beispiel werden bei allen Artikeln, welche den Steuersatz mit der TaxID 1 verwenden, der Steuersatz mit der TaxID 2 hinterlegt. Die genaue TaxID findest Du in der Datenbank Tabelle: s_core_tax.

Wenn Du in einer Versandart oder in einem Gutschein feste Steuersätze hinterlegt hast, solltest Du diese ebenfalls durch den neuen Steuersatz ersetzen.

Zusätzlich solltest Du Deine Plugins überprüfen, ob diese auf den Standard Steuersatz zurückgreifen. Ggf. muss der Steuersatz hier dann ebenfalls angepasst werden.

Nettopreis anpassen

In Shopware werden in der Datenbank die Nettopreise gespeichert. Wenn Du den Bruttopreis für den Kunden beibehalten und somit die Nettopreise anpassen möchtest, sind Änderungen in der Datenbank notwendig.

Da in diesem Fall der Bruttopreis gleich bleibt wird hier der Nettopreis geändert welcher in der Datenbank hinterlegt ist. Wenn Du Kundengruppen hast, denen der Nettopreis direkt im Frontend angezeigt wird, ändert sich dieser entsprechend auch.

Wenn Du in deinem Standard Steuersatz Steuerregeln für andere Länder hinterlegt hast solltest Du beachten, dass diese sich auf den Nettopreis der Artikel beziehen. Hast Du also für Österreich eine Steuerregel von 20% eingerichtet, werden diese auf den veränderten Nettopreis gerechnet. Der Bruttopreis für Österreich ändert sich dann also in Deinem Shop.

Der erste Schritt ist wie im Schritt Bruttopreis anpassen der, den jeweiligen Steuersatz im Backend unter **Einstellungen > Grundeinstellungen > Shopeinstellungen > Steuern** anzupassen.

Wenn Du nur Artikel mit einem Steuersatz hast, kannst Du dich an den folgenden Beispielen orientieren. Es wird hierzu zunächst durch den neuen Steuersatz dividieret und anschließend mit dem alten Steuersatz multipliziert


```
UPDATE s_articles_prices SET price = price/1.neuer-Steuersatz*1.alter-Steuersatz
```

Beispiele

Steuersenkung

Als Beispiel dient die Änderung des alten Steuersatzes **19%** auf den neue Steuersatz **16%**.

```
UPDATE s_articles_prices SET price = price/1.16*1.19
```

Steuererhöhung

Umgekehrt sieht das Query für die Erhöhung des Steuersatzes von **16%** auf **19%** aus:

```
UPDATE s_articles_prices SET price = price/1.19*1.16
```

Verschiedene Steuersätze berücksichtigen

Wenn Du Artikel aus mehreren Steuersätzen hast, Du aber nur einen der Steuersätze anpassen möchtest, führst Du stattdessen diesen SQL-Query aus:

```
UPDATE s_articles_prices, s_articles SET s_articles_prices.price = s_articles_prices.price/1.
neuer-Steuersatz*1.alter-Steuersatz
WHERE (`s_articles`.id = `s_articles_prices`.articleID AND `s_articles`.taxID='
ID-des-Steuersatzes')
```

Die genaue TaxID findest Du in der Datenbank Tabelle: s_core_tax.

Beispiel

```
UPDATE s_articles_prices, s_articles SET s_articles_prices.price = s_articles_prices.price/1.1
WHERE (`s_articles`.id = `s_articles_prices`.articleID AND `s_articles`.taxID='3')
```

In diesem Beispiel wird der Steuersatz von **19%** auf **16%** geändert. Dies passiert für alle Artikel, die den Steuersatz mit der TaxID 3 verwenden. Die genaue TaxID findest Du in der Datenbank Tabelle: s_core_tax.

Netto-Händlerpreise nicht ändern

Solltest Du neben den normalen Shopkunden-Preisen ebenfalls Nettopreise für die Händler-Kundengruppe gepflegt haben und diese nicht ändern wollen, kannst Du im SQL-Query eine zusätzliche Prüfung auf die "pricegroup" durchführen.

Ein Query, das nur die Preise für die Kundengruppe "EK" ändert, könnte z.B. so aussehen

```
UPDATE s_articles_prices, s_articles SET s_articles_prices.price = s_articles_prices.price/1.1
```

Dieses Query kann entsprechend angepasst werden, um den Preis auch für andere Kundengruppen zu ändern. Eine Übersicht der vorhandenen Kundengruppen findest Du im Backend in den Grundeinstellungen

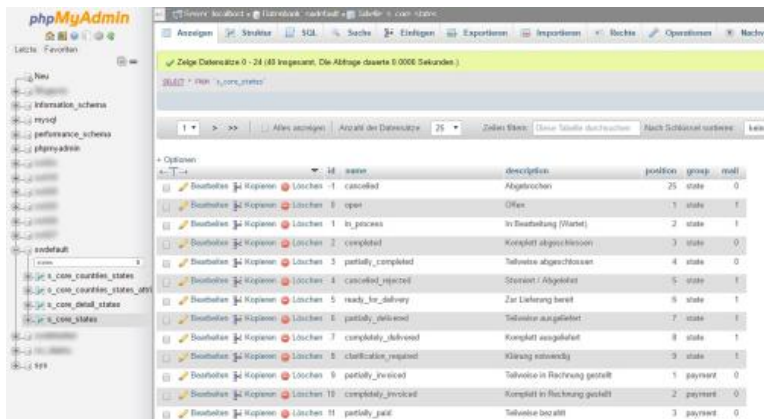
unter Shopeinstellungen > Kundengruppen. Alternativ kannst Du die vorhandenen Kundengruppen auch in der Datenbank in der Tabelle `s_core_customergroups` einsehen.

Neuen Bestell- und Zahlstatus erstellen

In diesem Artikel erklären wir, wie Du einen eigenen Bestell- oder Zahlstatus erstellen und diesen mit einer E-Mail-Vorlage verknüpfen kannst.

Status anlegen

Als erstes musst Du den gewünschten Status in der Datenbank anlegen. Öffne dazu in Deinem phpMyAdmin in Deiner Datenbank die Tabelle "`s_core_states`":



id	name	description	position	group	mail
1	cancelled	Abgebrochen	25	state	0
2	open	Offen	1	state	1
3	in process	In Bearbeitung (Wartet)	2	state	1
4	completed	Konflikt abgeloöst	3	state	0
5	partially_completed	Teilweise abgeschlossen	4	state	0
6	cancelled_rejected	Storniert / Abgelehnt	5	state	1
7	ready_for_delivery	Zur Lieferung bereit	6	state	1
8	partially_invoiced	Teilweise angefordert	7	state	1
9	completely_invoiced	Konflikt angefordert	8	state	1
10	classification_required	Klassifizierung notwendig	9	state	1
11	partially_invoiced	Teilweise in Rechnung gestellt	1	payment	0
12	completely_invoiced	Komplett in Rechnung gestellt	2	payment	0
13	partially_paid	Teilweise bezahlt	3	payment	0

Schau nun, welche ID du verwenden kannst. Im Standard ist die erste freie ID die 36. Füge nun einen neuen Datensatz in die Tabelle ein:

Spalte	Typ	Funktion	Null	Wert
id	int(11)			36
name	varchar(55)		<input type="checkbox"/>	mein_neuer_Status
description	varchar(255)			mein neuer status
position	int(11)			10
group	varchar(25)			state
mail	int(1)			1

Ignorieren

Spalte	Typ	Funktion	Null	Wert
id	int(11)			
name	varchar(55)		<input checked="" type="checkbox"/>	
description	varchar(255)			
position	int(11)			
group	varchar(25)			
mail	int(1)			

- **name** ist ein intern verwendeter Name
- **description** ist der öffentliche Name dieses Status, den auch der Kunde zu Gesicht bekommen wird.
- **position** beschreibt die Position des Status innerhalb der Statusliste im Backend.
- **group** wählt die Zugehörigkeit des Status, "state" für Bestellstatus und "payment" für Zahlstatus.
- **mail** definiert, ob zu diesem Status eine Mail versendet werden soll (1) oder nicht (0)

E-Mail-Vorlage erstellen

Der Status ist erstellt, nun muss noch eine E-Mail-Vorlage für ihn erstellt werden. Diese E-Mail-Vorlage muss den Namen "sORDERSTATEMAIL"+"ID" tragen, in unserem Fall also "sORDERSTATEMAIL36". Erstelle diese E-Mail-Vorlage und fülle Sie mit dem gewünschten Inhalt:

Mail-Vorlagen

Vorlage bearbeiten: sORDERSTATEMAIL36

Name: sORDERSTATEMAIL36

Abkürzung: sORDERSTATEMAIL

Name: sORDERSTATEMAIL

Inhalt:

HTML: Vorlage als HTML-E-Mail versenden

Neuer E-Mail-Vorlage

Leere nun den Cache und lade das Backend einmal neu.

Statusnamen im Backend bereitstellen

Als erstes öffnest Du die Textbausteinverwaltung und erstellst unter "backend/static/payment_status" für Zahlungsstatus oder "backend/static/order_status" für Bestellstatus einen neuen Textbaustein, zum Beispiel "MeinNeuerStatus" als technischem Namen, welchen Du mit Deinem Wunschtext füllst:




The screenshot shows a dialog box titled "Textbaustein erstellen" with a close button (X) in the top right corner. It contains the following fields:

- Locale: de_DE (dropdown menu)
- Shop: Demoshop (dropdown menu)
- Namespace: backend/static/payment_status (text input)
- Name: MeinNeuerStatus (text input)
- Wert: Mein neuer Status (text input)

A blue "Speichern" button is located at the bottom left of the dialog.

Status im Frontend bereitstellen

Als erstes öffnest Du die Textbausteinverwaltung und erstellst unter "frontend/account/order_item" einen neuen Textbaustein, zum Beispiel "OrderItemInfoMeinNeuerStatus", welchen Du mit Deinem Wunschtext füllst:



The screenshot shows a dialog box titled "Textbaustein erstellen" with a close button (X) in the top right corner. It contains the following fields:

- Locale: de_DE (dropdown menu)
- Shop: Demoshop (dropdown menu)
- Namespace: frontend/account/order_item (text input)
- Name: OrderItemInfoMeinNeuerStatus (text input)
- Wert: Mein neuer Status (text input)

A blue "Speichern" button is located at the bottom left of the dialog.

Nun kannst Du den Textbaustein im Frontend bereitstellen. Erstelle dazu auf dem Server eine Templatedatei "order_item.tpl" im Verzeichnis /themes/Frontend/EIGENERTHEMENAME/frontend/account/. Dort muss nun der folgende Block mit Deinem neuen Status integriert werden:

```
{extends file='parent:frontend/account/order_item.tpl'}  
{block name="frontend_account_order_item_status_value"}  
<div class="column--value">  
  <span class="order--status-icon status--{$offerPosition.status}"></span>  
  {if $offerPosition.status==0}
```

```

    {s name="OrderItemInfoNotProcessed"}/{/s}
  {elseif $offerPosition.status==1}
    {s name="OrderItemInfoInProgress"}/{/s}
  {elseif $offerPosition.status==2}
    {s name="OrderItemInfoCompleted"}/{/s}
  {elseif $offerPosition.status==3}
    {s name="OrderItemInfoPartiallyCompleted"}/{/s}
  {elseif $offerPosition.status==4}
    {s name="OrderItemInfoCanceled"}/{/s}
  {elseif $offerPosition.status==5}
    {s name="OrderItemInfoReadyForShipping"}/{/s}
  {elseif $offerPosition.status==6}
    {s name="OrderItemInfoPartiallyShipped"}/{/s}
  {elseif $offerPosition.status==7}
    {s name="OrderItemInfoShipped"}/{/s}
  {elseif $offerPosition.status==8}
    {s name="OrderItemInfoClarificationNeeded"}/{/s}
  {elseif $offerPosition.status==36}
    {s name="OrderItemInfoMeinNeuerStatus"}/{/s}
  {/if}
</div>
{/block}

```

Damit nun noch das entsprechende Icon neben dem Status angezeigt wird ist es nötig, eine eigene "all.less" Datei zu erstellen. Der dort verwendete LESS-Code wird vom LESS-Compiler in normales CSS kompiliert.

Wechsle dazu in das Verzeichnis /themes/Frontend/EIGENERTHEMENENAME/frontend/_public/src/less und erstelle dort die Datei all.less

Füge dann bitte den folgenden Code ein:

```

.order--status-icon {
  position: static;

  &.status--36 { //Mein neuer Status
    background: @highlight-info;
  }
}

```

Testen

Nachdem Du diese Änderungen vorgenommen hast, kannst Du den Status Im Bestellungsmdoul wie gewohnt nutzen:

The screenshot shows a web interface with a search bar at the top. Below it, there are two dropdown menus: 'Aktueller Bestellstatus' and 'Aktueller Zahlungsstat'. The 'Aktueller Bestellstatus' dropdown is open, showing a list of status options: 'Offen', 'In Bearbeitung (Wartet)', 'Komplett abgeschlossen', 'Teilweise abgeschlossen', 'Storniert / Abgelehnt', 'Zur Lieferung bereit', 'Teilweise ausgeliefert', 'Komplett ausgeliefert', 'Klärung notwendig', and 'Mein neuer Status'. The 'Mein neuer Status' option is highlighted at the bottom of the list.

Auch im Frontend wird der neue Status nun angezeigt:

Bestellungen

Hier finden Sie Ihre Bestellungen nach Datum sortiert.

Datum	Bestellnummer	Versandart	Bestellstatus	Aktionen
19.10.2016 15:47:00	20001	Standard Versand	■ Mein neuer Status	Anzeigen

Seite 1 von 1

Performance Tipps

Allgemeine Tipps

Topic	Standard	Optimal	Info
PHP-Version	-	7.4.x	PHP 7.4.x bietet je nach Szenario eine zwischen 25 - 30 % höhere Ausführungsgeschwindigkeit (PHP-Stack). Des Weiteren sollte das Memory_Limit - je nach Anzahl der verwendeten Plugins und Datensätze - mindestens 512MB betragen.
Bytecode-Cache	APC	ZendOpcache + APCu	Es sollte in jedem Fall ein Bytecode-Cache installiert und korrekt konfiguriert sein - dies beschleunigt die Ausführung des PHP-Stacks nochmal um bis zu 25 %
MySQL	MySQL 5.7.x	MySQL 5.7.x oder MariaDB (10.x.x)	MySQL sollte in der aktuellen für Produktivzwecke freigegebenen Major-Version eingesetzt werden, da es hier deutliche Verbesserungen bei der Ausführung von Joins, SubSelects und Lese-Zugriffen auf die Datenbank gegeben hat.
Webserver	Apache 2.2 oder 2.4	Nginx	Bei Shops mit hohem Traffic lohnt sich auch der Einsatz eines alternativen Webservers wie z.B. Nginx. Stelle sicher, dass das mod_rewrite Module aktiv ist und die Möglichkeit besteht die Webserver-Einstellungen in der .htaccess zu ändern.
Debug-/Profiling	-	-	Achte darauf, dass auf dem Produktivserver keine Profilingtools wie xDebug oder XhProf aktiv sind - diese reduzieren die Ausführungsgeschwindigkeit von PHP deutlich. Ebenfalls sollten die Shopware Debug & Benchmark-Plugins nie im Produktivbetrieb verwendet werden

Mit jedem neuen Shopware-Release fließen auch Performance-Verbesserungen in die Entwicklung ein. Daher ist es ratsam, jeweils zeitnah auf die neu verfügbare, aktuellste Shopware Version umzusteigen.

Auswahl eines geeigneten Hosters

Der Hoster muss zum geplanten Einsatz-Szenario passen und sollte einen Upgrade-Pfad für das gewählte Hosting-Paket anbieten, sodass dieses zusammen mit den Anforderungen wachsen kann. Auf folgende Punkte solltest du achten:

- Shared Hosting / VServer / Dedicated Server
- Software-Stack sollte sich mit unseren Empfehlungen (siehe oben) decken
- Shopware Erfahrung vorhanden?
- Shopware Referenz-Shops?

Grundsätzlich empfehlen wir einen unserer zertifizierten Hoster zu nutzen. Eine aktuelle Übersicht findest Du unter folgendem Link: [Shopware Hosting Partner](#)

Wie messe ich die Performance?

Die angegebenen Ladezeiten sind nur als Indikator zu verstehen, da diese je nach Anzahl / Struktur des Shops und dem aktuellen Traffic abweichen können.

Die nachfolgenden Analyse-Schritte sollten Sie in einem Test-System durchführen - dieses können Sie in einem Unterordner anhand dieser Anleitung anlegen: <https://docs.shopware.com/de/shopware-5-de/tutorials-und-faq/shop-transfer-live-system-testumgebung>

- ausnahmslos alle Drittanbieter Erweiterungen deaktivieren
- Standard Responsive Theme zuweisen
- Cache aufwärmen
- Performance prüfen

Sollten diese Schritte zu einer Verbesserung geführt haben, gilt es die Faktoren Schritt-für-Schritt auszuschließen. Falls es weiterhin eine nicht zufriedenstellende Performance gibt, empfiehlt es sich die folgenden Benchmark-Werte zu ermitteln:

- Wie viele Artikel hat der Shop? Wie viele Varianten?
- Wie viele Kategorien hat der Shop? Wie viele Kategoriezuweisungen?
- Wie viele gleichzeitige Zugriffe gibt es?
- An welcher Stelle macht sich die schlechte Performance bemerkbar?
- Kann hier eine Uhrzeit definiert werden, bei der es zu den Problemen kommt?

Der Cronjob "Aufräumen" ist elementar für eine Bereinigung des Systems - folgende Datenbank Tabellen werden hier aufgeräumt:

- *_s emarketing_lastarticles* - Artikelverläufe die älter als 30 Tage sind
- *_s statistics_search*- Ähnliche Suchbegriffe die älter als 30 Tage sind
- *_s core_log* - Backend-Log Einträge die älter als 30 Tage sind
- *_s order_notes* - Merklisten die älter als ein Jahr sind und keine Benutzer-Zuordnung haben
- *_s statistics_referrer* - Referrer die älter als 90 Tage sind (Standard)
- *_s statistics_article_impression* - Artikelaufrufe die älter als 90 Tage sind (Standard)

Ebenfalls überprüfen Sie bitte die Anzahl der Datensätze in den folgenden Datenbank-Tabellen:

- *_s articles_categories_ro*
- *_s articles_details*
- *_s articles_prices*
- *_s articles_similar_shown_ro*
- *_s articles_also_bought_ro*

Messen der Performance über Firebug

Damit die Ergebnisse vergleichbar sind, wird empfohlen das Shopware Demo-Daten Package als Test-Grundlage zu verwenden. Installiere die Firefox-Extension Firebug. Anschließend kannst Du über den Netzwerk-Tab die Ladezeiten der einzelnen Seiten-Komponenten einsehen.

Relevant ist die Ladezeit der Seite selbst - das ist in der Regel der erste Eintrag in der Liste. Rufe dann die Startseite, Listing und Detailseite jeweils 5 bis 10 mal auf und ermittle aus den einzelnen Ausführungszeiten einen Mittelwert.

Messen der Performance mit Apache-Benchmark

Aufruf über Linux-Shell:

```
ab -n 10 -c 1 http://URL_DER_STARTSEITE / KATEGORIE / DETAILSEITE
```

Dort dann jeweils den Median der Total-Time (Connection-Times) als Wert übernehmen.

Analyse Möglichkeiten Performance

Task	Ansprechpartner	Info
------	-----------------	------

Problematische Komponenten identifizieren	Kunde / Hoster	Wo genau treten Performance-Engpässe auf? Webserver, Datenbankserver, PHP-Stack, etc.
SQL-Slow-Query Auswertung	Hoster	Welche Datenbank-Queries beeinflussen die Gesamtperformance des Shops?
Hosting-Plattform Analyse	Hoster / Kunde	Ist die Hosting-Umgebung geeignet für die Größe, den Traffic, des Shops? Aktuelle / Empfohlene Server-Komponenten im Einsatz? Dedicated Server - Also kein VHost oder Webspace-Paket - Keine Shopware fremden Applikationen auf dem Server im Einsatz?
Anbindungen	Kunde	Werden regelmäßig durch die Anbindung an ein ERP-System große Datenmengen hinzugefügt / verändert? Werden regelmäßig andere Tasks auf dem Shop-Server ausgeführt, die die Performance beeinträchtigen könnten?

Mysql Konfiguration optimieren

Die Standard-Mysql-Konfiguration vieler Distributionen ist nicht für den Produktivbetrieb optimiert. Das Script `mysqltuner` bietet **Anhaltspunkte**, welche Konfigurationsparameter noch nicht ideal sind.

```
wget http://mysqltuner.com/mysqltuner.pl
chmod +x mysqltuner.pl
./mysqltuner.pl
```

Engpässe durch eine hohe Anzahl an Eigenschaften

Setze die Eigenschaft `optimizer_search_depth=0` in der MySQL-Konfiguration. Weitere Informationen erhältst du auch unter folgendem Link: http://www.mysqlperformanceblog.com/2012/04/20/joining-many-tables-in-mysql-optimizer_search_depth/

Tipps & Tricks

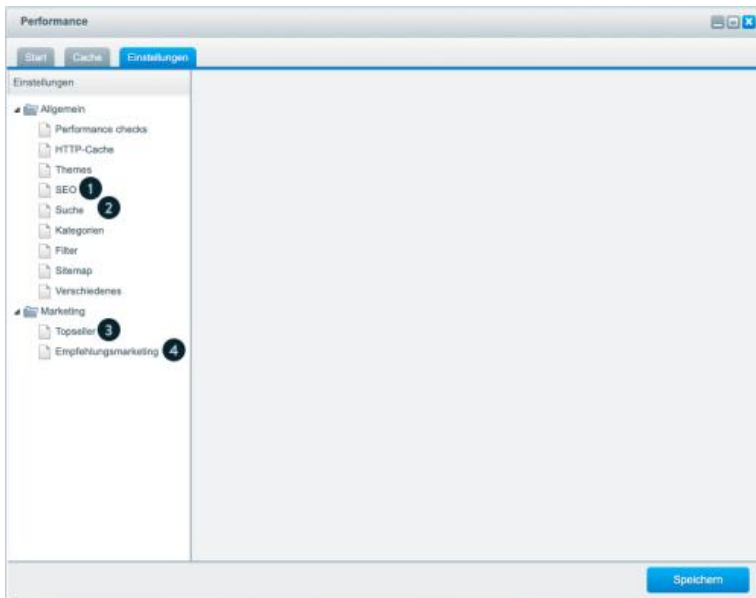
Standard-Sortierung Kategorie-Listen

Sofern Du hier ein eigenes Sortier-Kriterium definiert hast, solltest Du darauf achten, dass für die ausgewählte Tabellen-Spalte ein Index in der Datenbank hinterlegt ist.

Umstellung der Aktualisierungsstrategie

Aktuell gibt es im Shop verschiedene Prozesse, die bei jedem Seitenaufruf ausgeführt werden. Das ist beispielsweise die Erzeugung von SEO-Links, der Aufbau des Suchcaches und auch die Generierung von Topsellern und der "Kunden haben sich ebenfalls angesehen"-Funktion.

Hier solltest Du einmal im Detail prüfen, ob Du diese Funktionen nutzt und die Aktualisierungsstrategie von Live (pro Seitenaufruf) auf Cronjob (gesammelt, zeitgesteuert) umstellen. Dies kann im Shopware-Backend über **Einstellungen > Caches/Performance (anklicken) > Reiter Einstellungen** durchgeführt werden:



Dort findest Du in den Punkten **SEO (1)**, **Suche (2)**, **Topseller (3)** und **Empfehlungsmarketing (4)** jeweils eine Einstellung zur Aktualisierungsstrategie. Um den Cronjob nun in Shopware einzurichten, musst Du die Plugins "Cron" und "CronRefresh" installieren. Im Anschluss muss der Aufruf dann zeitgesteuert (am besten in der Nacht) durch den Server erfolgen. Hierzu findest Du eine Dokumentation in unserem Wiki: [Cronjobs](#)

Aktivierung des HTTP-Caches

Shopware bietet zudem die Möglichkeit nicht jede Seite live zu berechnen, sondern cachen zu lassen. Hier wird eine html-Cache-Datei angelegt, die alle wichtigen Informationen des Artikel-Listings (Kategorie) und der Detailseite (Artikel) beinhaltet, damit diese nicht bei jedem Aufruf aus der Datenbank abgeholt werden muss. Dies führt ebenfalls vor allem bei der Navigation durch den Shop zu einem großen Performancezuwachs. Dieser Cache wird bei Preisänderungen invalidiert, damit die Preise immer aktuell dargestellt werden.

Durch den oben eingerichteten Cronjob wird der Cache auch nicht zu groß, da er täglich (im Standard alle 24h) geleert wird. Ein gut genutzter Http-Cache kann am Tag 1-2GB Volumen haben. Hier solltest Du einmal prüfen, ob alle Änderungen/Plugins kompatibel sind oder dies ggf. in der Testumgebung einmal probelaufen lassen. Den HTTP-Cache kannst Du über das Performance-Modul aktivieren (**Einstellungen > Caches/Performance (anklicken) > Einstellungen > HTTP-Cache**).

Plugin-Prüfung

In vielen Fällen führen Plugins auch zu einer deutlichen Verschlechterung der Performance des Shops. Hierzu wäre es notwendig alle Fremdanbieter-Plugins zu deaktivieren und anschließend die Performance im Frontend erneut zu testen.

Der einfachste Schritt ist dieses mittels SQL Query in der Datenbank einer Testumgebung zu machen. Die beiden nachfolgenden Queries können im phpMyAdmin ausgeführt werden, damit alle Plugins, die Shopware nicht standardmäßig mitliefert, deaktiviert werden:

```
UPDATE `s_core_plugins` SET `active`= 0 WHERE `source` <> 'Default' AND author <> 'shopware AG'
```

Eine genaue Anleitung hierzu findest Du auch in unserer [Shopware SQL Trickkiste](#).

Alternativ kannst Du auch einfach den [Sicherheitsmodus](#) im Plugin Manager aktivieren, der Dir ab Shopware 5.3 zur Verfügung steht.

Wenn der Shop dann vernünftig läuft, kann man durch schrittweises Aktivieren der Plugins den "Übeltäter" finden. Wenn Du Plugins installiert hast, die Du aktuell nicht einsetzt, macht es auch Sinn diese zu deaktivieren. Hier solltest Du auch prüfen, ob wirklich alle Plugins benötigt werden.

Plugin "Debug"

Im Shopware-Standard gibt es das Plugin "Debug", welches beispielsweise einzelne Prozesse auf dem Server benchmarken kann. Hier kannst Du relativ genau sehen, wie lange ein Seitenaufruf braucht und auch welche Prozesse (Datenbankabfragen, Controller, usw.) den Aufruf bremsen.

Serveroptimierung

Aktuell setzt Du eventuell noch kein erweitertes Caching wie ZendOP und APCu ein. Dies solltest Du zusätzlich installieren/aktivieren, um nochmal einen Performance-Boost zu erreichen. Die Performance-Checks findest Du unter **Einstellungen > Caches/Performance (anklicken) > Einstellungen > Performance checks**:

Systemprüfung

Unter **Einstellungen > Systeminfo** kannst Du Informationen über dein System finden:

Name	Benötigt	Version	Status
Einstellungen			
ini_set	1	1	✓
include_path	1	1	✓
session.auto_start	0	0	⚠
memory_limit	256M	4G	⚠
max_execution_time	30	120	✓
upload_max_filesize	6M	64M	✓
post_max_size	8M	32M	✓
allow_url_fopen	1	1	✓
curl_exec	1	1	✓
curl_multi_exec	1	1	✓
file_uploads	1	1	✓
Suhosin.session.cryptkey	0	0	⚠
Suhosin.get_max_value_length	2000	2000	✓
parse_ini_file	1	1	✓
Allgemein			
php	7.4.2	8.0.23	✓
mysql	5.7.0	8.0.32	✓
mysql_strict_mode	0	0	⚠
opcache.use_cwd	1	1	✓
Erweiterungen			
pdo	1	1	✓
pdo_mysql	1	1	✓
gd	2.0.0	2.3.3	✓
gd_jpeg	1	1	✓
freezope	1	1	✓

Dort gibt es beispielsweise das Merkmal "memory_limit" (**1**), welches bei Unterschreitung entscheidend auf die Performance einwirken kann. Empfohlen sind mindestens 512MB (abhängig von der Anzahl der Plugins sowie Datensätze). Hier ist zu beachten, dass dem System auch nur vorgegaukelt werden kann, dass hier ein ausreichender Wert erzielt wird, sodass hier das System generell einmal überprüft werden sollte.

Prinzipiell ist hierzu anzumerken, dass die Mindestvoraussetzung auf einen Shopware-Basis Shop ohne Plugins ausgelegt ist und an dein System angepasst werden sollte.

Compiler Caching

Unter **Einstellungen > Theme Manager > Einstellungen** kannst Du das Compiler Caching deaktivieren. Dies sollte im Produktivmodus niemals gesetzt sein, da es den Shop extrem verlangsamt und nur zu Entwicklungszwecken angedacht ist.

Shopware Plugin-Lizenzen und Subscriptions - FAQ

Lizenzen FAQ

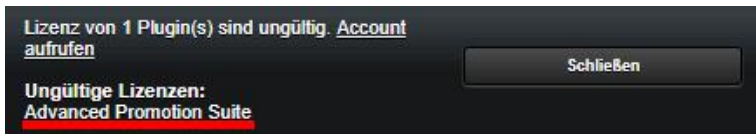
Was bedeutet das Fenster "Plugin Warnung" beim Öffnen meines Backends?

Hast Du ein Plugin im Einsatz, für das Du keine gültige Lizenz besitzt, so taucht eine entsprechende Hinweismeldung beim Öffnen des Backends auf. Mindestens eine Lizenz der von Dir eingesetzten Plugins ist dann abgelaufen.



Wie finde ich heraus, welche Plugins nicht lizenziert sind?

Neben der Warnhinweis-Box (s. o.) wird Dir im Shopware Backend noch eine weitere, kleinere Hinweisbox angezeigt. Diese zeigt Dir den Namen des betroffenen Plugins an. Hierbei macht Shopware keinen Unterschied, ob das Plugin im Plugin-Manager den Status "Aktiviert", "Deaktiviert" oder "Deinstalliert" besitzt. **Nicht lizenzierte Plugins müssen komplett gelöscht werden und dürfen nicht mehr im Plugin-Manager Deines Shops auftauchen!** Bei diesem Beispiel taucht das unlizenzierte Plugin "Advanced Promotion Suite" noch im Plugin-Manager auf. Es muss komplett gelöscht werden oder die benötigte Lizenz wird über den Shopware Store erworben.



Hinweis: Sollte diese Meldung nicht in Deinem Backend auftauchen, so lösche in Deinem Browser alle Shopware-Cookies und lade das Backend neu.

Wie lösche ich das betroffene Plugin?

Um die Lizenzverletzung zu beheben, muss das Plugin komplett aus dem Pluginmanager gelöscht werden. Es reicht nicht, dass es nur deaktiviert ist, da das Plugin dann jederzeit wieder aktiviert werden kann.

Schritt 1: Öffne im Backend Deines Shops den Plugin-Manager, wähle dort den Reiter "Installiert". Dann deaktiviere das Plugin im Bereich "Aktiv" (sofern noch aktiv) durch einen Klick auf den grünen Haken:

Installiert (11 Plugins)									
Custom Products	4.2.3	24.09.2010	29.10.2010	Testversion bis: 14.09.2020	Shopware AG	✓	✖	🗑️	🔧
Shopware ImportExport	2.0.2	27.10.2010	09.10.2010		Shopware AG	✓	✖	🗑️	🔧
Advanced Promotion Suite	2.4.1	19.07.2016	17.10.2016	Kaufversion	Shopware AG	✓	✖	🗑️	🔧
Publication	1.0.0	20.04.2016	20.04.2016		Shopware AG	✓	✖	🗑️	🔧
CrossRating	1.0.0	26.12.2014	26.12.2014		Shopware AG	✓	✖	🗑️	🔧
Erweiterbares Menü	1	16.12.2014	16.12.2014		Shopware AG	✓	✖	🗑️	🔧
CrossFeedback	1.0.0	14.12.2014	14.12.2014		Shopware AG	✓	✖	🗑️	🔧
Crash	1.0.0	14.12.2014	14.12.2014		Shopware AG	✓	✖	🗑️	🔧
Shopware Auto Updates	1.0.0	06.05.2014	06.05.2014		Shopware AG	✓	✖	🗑️	🔧
Statistics	1	10.10.2010	10.10.2010		Shopware AG	✓	✖	🗑️	🔧

Schritt 2: Das Plugin taucht nun unter "Deaktiviert" auf. Klicke dort auf den Button "Installieren/Deinstallieren".

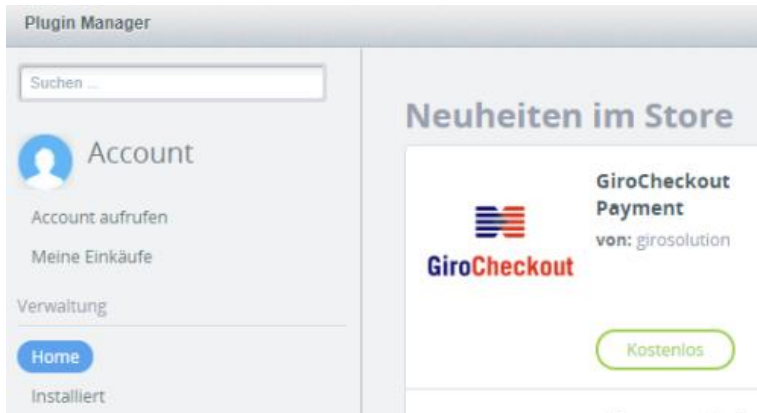
Deaktiviert (1 Plugin)									
Advanced Promotion Suite	2.4.1	19.07.2016	17.10.2016	Kaufversion	Shopware AG	✖	✖	🗑️	🔧
Advanced Promotion Suite	2.4.0	01.02.2016	19.08.2016		Shopware AG	✖	✖	🗑️	🔧

Schritt 3: Das Plugin wird nun unter "Deinstalliert" gelistet. Klicke dort auf den Button "Löschen". Somit wird das Plugin endgültig aus dem Plugin-Manager entfernt.



Hinweis: Sollte sich das Plugin nicht löschen lassen, kann es in den meisten Fällen direkt aus der Datenbank gelöscht werden. Sollte es dabei Komplikationen geben, wende Dich an den Pluginhersteller, der Dir dazu Unterstützung bieten kann.

Schritt 4: Melde Dich im Pluginmanager unter dem Punkt "Meine Einkäufe" mit Deinen Accountzugangsdaten an, damit die Löschung des Plugins bei uns angezeigt wird.



Achtung: Falls eine Installation im Unterordner oder eine lokale Installation unter der Domain vorliegt, muss das Plugin auch dort komplett gelöscht werden.

Wieso erscheint im Backend immer noch die Meldung "Testlizenz abgelaufen", obwohl ich eine gültige Mietlizenz hinterlegt habe?

In diesem Fall wird es so sein, dass im Lizenz-Manager im Backend von Shopware neben der Kauflizenz immer noch die Testlizenz für das betroffene Plugin hinterlegt ist.

Lösche einfach die Testlizenz im Lizenz-Manager im Backend. Danach sollte die Meldung "Testlizenz abgelaufen" für das betroffene Plugin nicht mehr erscheinen.

Subscriptions FAQ

Die Informationen aus diesem Artikel beziehen sich auf bereits bestehende Subscriptions. Neue Subscriptions für Shopware 5 Plugins können nicht mehr abgeschlossen werden. Weitere Informationen zum End of life von Shopware 5 findest Du [hier](#).

Was ist eine Plugin-Subscription?

Mit einer gültigen Plugin-Subscription stellst Du sicher, dass Du vom Plugin-Hersteller mit Plugin-Updates versorgt wirst. Updates können neue Plugin-Funktionen, Optimierungen, Kompatibilität zu neuen Shopware-Versionen oder die Behebung von Sicherheitslücken beinhalten.

Hinweis: Nicht alle Plugins besitzen eine Subscription-Prüfung. Kostenlose Plugins, wie beispielsweise das Shopware PayPal-Plugin, können geupdated werden, ohne dass hierfür eine Subscription gebucht werden muss.

Warum ist meine Plugin-Subscription nicht mehr gültig?

Die Subscriptions sind zeitlich begrenzt und müssen von Dir über Deinen Shopware Account erneuert werden. Der Kauf eines Plugins beinhaltet im Normalfall eine 1 Jahr gültige Subscription. Dies bedeutet: Du wirst 1 Jahr lang seitens des Plugin-Herstellers mit Plugin-Updates versorgt. Solltest Du die Plugin-Subscription nicht verlängern, kannst Du das Plugin nicht updaten und es erscheint eine entsprechende Meldung beim Öffnen Deines Backends:



Hinweis: Die Subscription-Prüfung checkt alle im Plugin-Manager gelisteten Plugins. Die Prüfung unterscheidet nicht, ob das Plugin "Aktiv", "Deaktiviert" oder "Deinstalliert" ist. Alle im Plugin-Manager gelisteten Plugins werden geprüft.

Welche Auswirkungen hat der Einsatz eines Plugins ohne gültige Subscription?

Eine fehlende Plugin-Subscription hat keinerlei Auswirkungen auf die Funktionalität von Shopware oder des betroffenen Plugins. Du solltest jedoch bedenken, dass Du in diesem Fall nicht mehr von Verbesserungen und Kompatibilitäten seitens des Plugin-Herstellers profitieren kannst.

Wie finde ich heraus, welches meiner im Backend im Plugin-Manager gelisteten Plugins keine Subscription mehr besitzt?

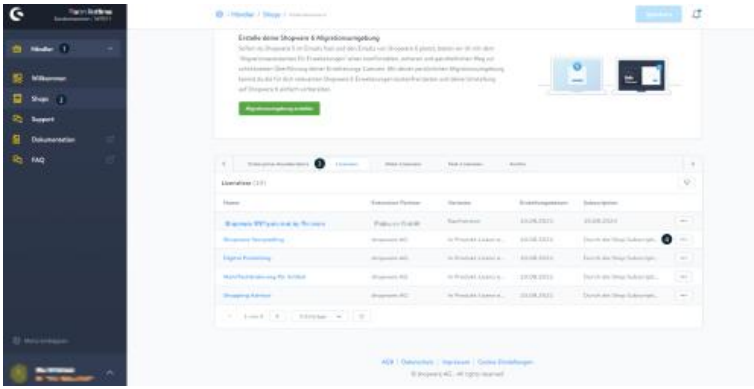
Die Hinweis-Meldung im Backend zeigt Dir an, welches Plugin keine gültige Subscription besitzt. Im Beispiel besitzen die zwei Plugins "Advanced Promotion Suite" und "Payment Icons/Zahlungs-Icons/Zahlungsarten für Header, Footer & Checkout" keine gültige Subscription mehr und können daher nicht mehr geupdated werden.



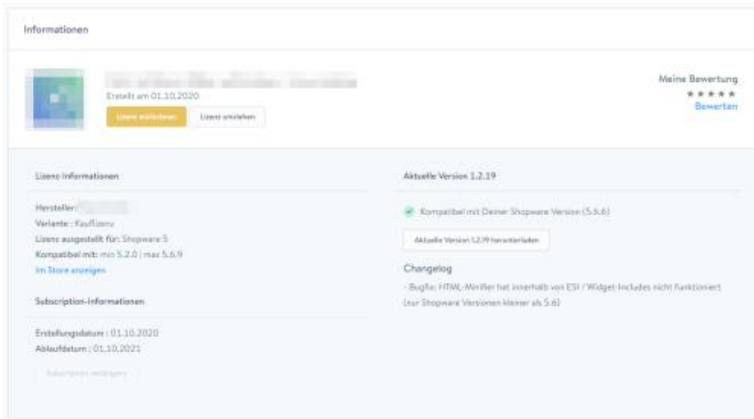
Wie kann ich eine Plugin-Subscription verlängern?

In Deinem [Shopware Account](#) findest Du auf der linken Seite den Menüpunkt **Händler (1)**. Nachdem Du diesen ausgewählt hast, kannst Du auf den Unterpunkt **Shops (2)** klicken. Dort werden Dir alle Domänen angezeigt, die mit Deinem Shopware Account verknüpft sind. Anschließend klickst Du die gewünschte Domäne an, damit hierfür die **Lizenzen (3)** angezeigt werden können. Innerhalb der Auflistung kannst Du erkennen, welche Plugins für Deine Domäne lizenziert sind und für welche Plugins eine Subscription aktiv

ist. Wenn Du auf die **...-Schaltfläche (4)** klickst, öffnet sich ein Kontextmenü, in dem Du dann auf **Details** klicken kannst.



Nach dem Klick auf **Details** wird Dir eine Übersicht über das ausgewählte Plugin angezeigt. Hier hast Du die Möglichkeit die Plugin-Subscription zu verlängern, allerdings erst 4 Wochen vor Ablauf der Subscription.

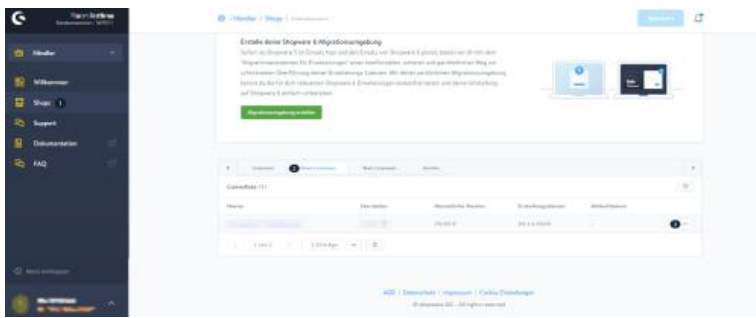


Hinweis: Ohne eine gültige Plugin-Subscription stehen Dir keine Plugin-Updates seitens des Herstellers zur Verfügung.

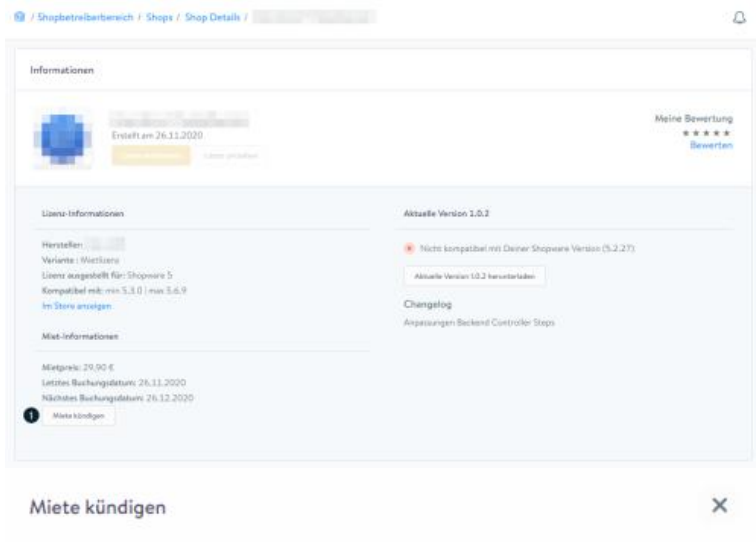
Wir empfehlen, die Plugins aktuell zu halten. Andernfalls verzichtest Du auf Verbesserungen, Bugfixes und eventuelle Sicherheitsupdates seitens des Plugin-Herstellers.

Wie kann ich eine Miet-Lizenz kündigen?

Wenn Du ein oder mehrere Plugins gemietet hast und diese Miete kündigen möchtest, musst Du Dich hierfür in Deinen Shopware Account unter <https://account.shopware.com/> einloggen. Dort findest Du im Menü unter **Händler > Shops (1)** Deine aktuellen Shops. Sobald Du auf den gewünschten Shop geklickt hast, erhältst Du eine Übersicht über Deine Lizenzen. Hierunter findest Du auch die **Miet-Lizenzen (2)**. Damit Du Deine Miet-Lizenz kündigen kannst, musst Du auf die **...-Schaltfläche (3)** klicken, um das Kontextmenü zu öffnen. In diesem Kontextmenü klickst Du dann auf **Details**.



Mit dem Klick auf **Details** gelangst Du auf die Übersichtsseite des ausgewählten Plugins. Du kannst nun mit dem Betätigen der Schaltfläche **Miete kündigen (1)** die Miet-Lizenz beenden.



Eine Kündigung der Miete des Plugins [redacted] ist zum 26.12.2020 möglich. Die Nutzung des Plugins ist ab dem 26.12.2020 untersagt und muss vollständig gemäß unserer [Anleitung](#) aus dem Shop entfernt werden. Dies wird in den kommenden Aktualisierungs- und Wartungsprozessen überprüft.

Ich bestätige, dass das Plugin ab dem Kündigungsdatum nicht mehr im Shop verwendet wird.

Nachdem Du den Haken in das Feld gesetzt hast, wird noch der Grund für die Beendigung der Miete abgefragt und anschließend kann die Miete gekündigt werden. Der Status der Kündigung wird Dir im Anschluss direkt auf der Übersichtsseite des Plugins angezeigt.

Ansprechpartner bei Rückfragen zum Account, Lizenzen oder Subscriptions

Bei Fragen stehen wir Dir unter store@shopware.com oder telefonisch unter +49 (0) 2555 928850 zur Verfügung.

Quelloffenheit Shopware 5.5

Shopware ist im Kern seit 5.2.0 quelloffen, Plugins waren aber noch länger verschlüsselt. Mit Shopware 5.5 endet nun auch diese Ära. Wir erklären Dir hier, wie Du Deinen Shop komplett quelloffen betreiben kannst.

Plugins prüfen

Wenn Plugins mit Shopware 5.5 kompatibel sein wollen, dürfen Sie nicht mehr verschlüsselt sein.

Prüfe alle im Store gekauften Plugins auf die Kompatibilität mit Shopware 5.5, du solltest vor dem Update von Shopware sicherstellen, dass alle Plugin kompatibel sind!

Nutzt Du individuelle Plugins von Drittanbietern, die nicht im Store verfügbar sind, erfrage beim Hersteller, ob die Plugins verschlüsselt sind und erfrage eine unverschlüsselte Version, die mit 5.5 kompatibel ist.

Wie prüfe ich die Plugins überhaupt?

Du kannst Plugins ganz einfach auf Verschlüsselung prüfen, indem Du die Bootstrap.php des jeweiligen Plugin öffnest.

Eine quelloffene Bootstrap sieht wie folgt aus:

```
1 <?php
2
3 class Shopware_Plugins_Frontend_SwagDemoDataDE_Bootstrap extends
4 Shopware_Components_Plugin_Bootstrap
5 {
6     public function getCapabilities()
7     {
8         return array(
9             'install' => true,
10            'update' => true,
11            'enable' => true,
12            'secureUninstall' => false
13        );
14    }
15
16    public function install()
17    {
18        if (!$this->assertMinimumVersion('5.4.0')) {
19            return false;
20        }
21    }
22}
```

Eine verschlüsselte dagegen sieht so aus:

```

1 <?php //00535
2 // Copyright shopware AG (www.shopware.com) [101dad2]
3 //
4 if(!extension_loaded('ionCube
* Loader')){$_oc=strtolower(substr(PHP_UNAME(),0,3));$_ln='ioncube_lo
* ader_'.$_oc.'.'.substr(PHP_VERSION(),0,3).(($_oc=='win')?''.dll':'.so
* ');if(function_exists('dl')){@dl($_ln);}if(function_exists('_il_exec
* ')){return
* _il_exec();}$_ln='/ioncube/'.$_ln;$_oid=$_id=realpath(ini_get('ex
* tension_dir'));$_here=dirname(__FILE__);if(strlen($_id)>1&&$_id[1]
* ==':'){$_id=str_replace('\\','/',substr($_id,2));$_here=str_replac
* e('\\','/',substr($_here,2));}$_rd=str_repeat('../',substr_count($_
* _id,'/')).$_here.'/';$_i=strlen($_rd);while($_i--){if($_rd[$_i]
* =='/'){$_lp=substr($_rd,0,$_i).$_ln;if(file_exists($_oid,$_lp))
* {$_ln=$_lp;break;}}if(function_exists('dl')){@dl($_ln);}}else{die
* ('The file '.__FILE__.' is
* corrupted.\n');}if(function_exists('_il_exec')){return
* _il_exec();}echo("Site error: the
* ".(PHP_SAPI_NAME()=='cli'? 'ionCube': '<a
* href="http://www.ioncube.com">ionCube</a>')." PHP Loader needs to be
* installed. This is a widely used PHP extension for running ionCube

```

Shopware aktualisieren

Im zweiten Schritt wird Shopware aktualisiert. Wenn Deine Shopware-Installation in Version 5.5.0 oder höher läuft, kannst Du Plugins aus dem Store nur noch unverschlüsselt herunterladen.

Plugins aktualisieren

Nach dem Update solltest Du alle Store-Plugins auf eine 5.5-kompatible Version aktualisieren, damit alle Store-Plugins unverschlüsselt auf Deinem System liegen.

Verschlüsselte, nicht über den Store vertriebene Plugins müssen spätestens hier ebenfalls selbst durch unverschlüsselte Versionen ersetzt werden!

Lizenz-Manager entfernen

Nachdem alle Plugins quelloffen vorliegen, kannst Du den Lizenz-Manager löschen und vom System entfernen.

Der Lizenz-Manager wird zwar aus Kompatibilitätsgründen auch in Shopware 5.5 lauffähig sein, jedoch wird er nicht mehr weiterentwickelt, sodass auch der Einsatz von modernen PHP-Versionen ab 7.1 mit installiertem Lizenz-Manager nicht möglich ist, wir empfehlen daher dringend, den Lizenz-Manager zum Abschluss zu entfernen!

Shop Transfer Live-System / Testumgebung

Dieser Artikel zeigt Dir, wie Du Deine (lokale) Testumgebung in eine Produktiv-Umgebung umwandelst sowie umgekehrt.

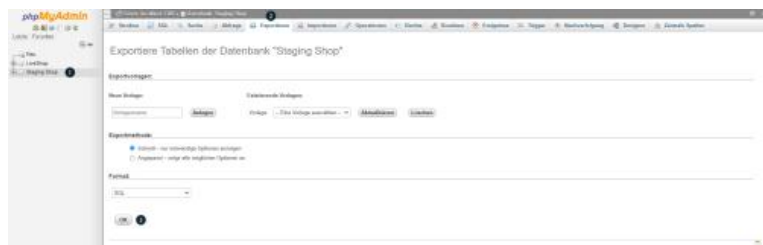
Testumgebung zu Live-System

Hier erläutern wir Dir, wie Du nach Deiner Entwicklung Deine Testumgebung auf das Live-System übertragen kannst.

Kopieren der Testumgebung

Öffne als erstes Deinen FTP-Client und kopiere die Dateien von Deiner Testumgebung in das Hauptverzeichnis Deines Shops. Achte darauf, nur das Shopware-Verzeichnis zu kopieren und die **Cache-Order vom Kopiervorgang auszuschließen**

MySQL-Datenbank kopieren

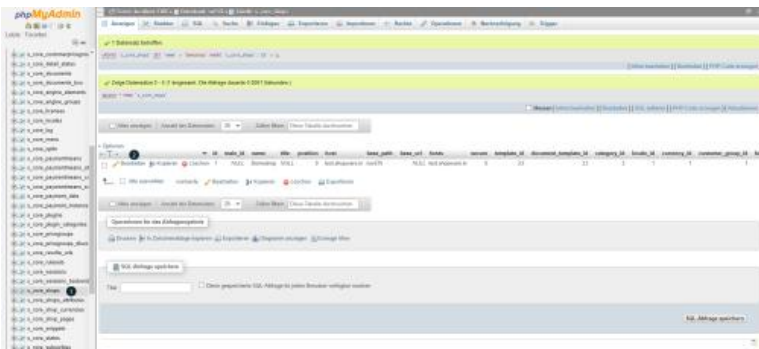


In diesem Schritt wird beschrieben, wie Du einen kompletten Dump (Kopie) Deiner Test-Datenbank anlegst und in Deine Live-Datenbank importierst. Öffne phpMyAdmin in Deiner Testumgebung und melde Dich dort an. Wähle dann Deine **Staging-Datenbank von Shopware (1)** aus und klicke danach in den Tab **Exportieren (2)** und starte über den Button **OK (3)** den Export Deiner Test-Datenbank.

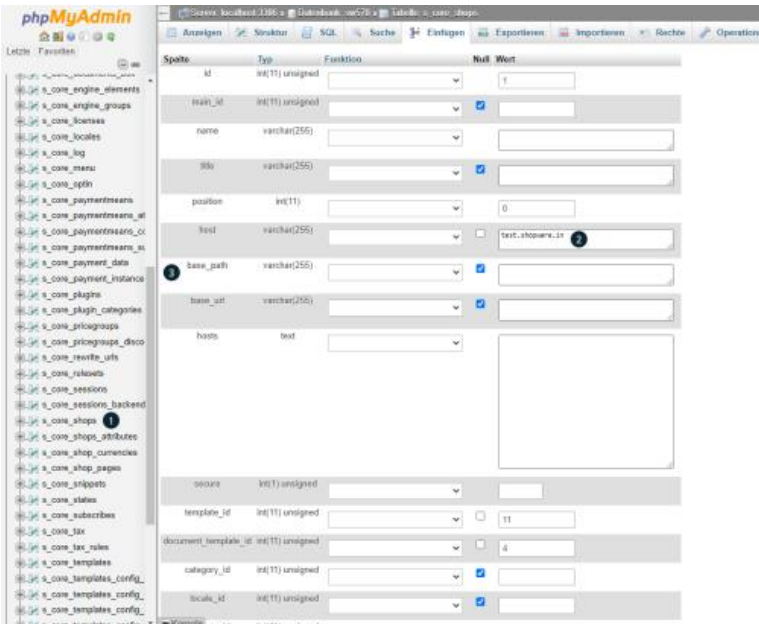


Spiele nun den zuvor erstellten MySQL-Dump deiner Test-Datenbank in Deine Produktiv-Umgebung ein. Öffne in Deiner Produktiv-Umgebung phpMyAdmin und melde dich dort an. Falls nicht bereits eine Datenbank vorhanden ist, erstelle dort eine Datenbank mit zum Beispiel dem Namen **shopware (1)** und wähle diese aus. Wechsel danach in den Tab **Importieren (2)**. Wähle dort den **zuvor erstellten SQL-Dump (3)** aus und importiere diesen über den Button **OK (4)**.

Anpassen der Shop-URL



Damit Dein Shop in der neuen Produktiv-Umgebung auch unter der korrekten Shop-URL erreichbar ist, müssen die Shop-URLs in der Datenbank geändert werden, wechsel dazu in phpMyAdmin in die Tabelle **s_core_shops (1)** und editiere den **Host-Eintrag des Hauptshops (2)**.



In unserem Beispiel soll die Testumgebung direkt unter <http://test.shopware.in> erreichbar sein. Entsprechend muss der Eintrag **Host (2)** in test.shopware.in geändert werden. Da wir unseren Shop nicht in einem Unterverzeichnis betreiben, muss der Eintrag **base_path (3)** leer sein, falls hier also ein Eintrag vorhanden ist, lösche diesen. Speichere danach die von Dir getätigten Änderungen ab.

Anpassung der config.php

Abschließend muss noch die Verbindung der Dateibasis zur MySQL-Datenbank auf dem Webserver wiederhergestellt werden. Öffne hierzu in Deiner Live-Umgebung das gerade kopierte Shop-Verzeichnis und editiere die Datei **config.php** im Hauptverzeichnis. Hinterlege hier den Benutzernamen und das Passwort zu Deiner MySQL-Datenbank und ggf. auch den abweichenden MySQL-Server. In der Regel sind diese gleichlautend mit Deinen Zugangsdaten zu phpMyAdmin.

Kontrolliere bzw. ändere hier folgende Einträge:

```
'username' => 'root',
'password' => 'root',
'host' => 'localhost',
```

```
'port' => '3306',  
'dbname' => 'shopware',
```

Nach Änderung der config.php ist es notwendig, den Cache einmal über FTP zu leeren. Falls Du den Cache nicht schon beim Kopieren weggelassen hast, kannst Du das wie folgt tun: Führe entweder die Datei `/var/cache/clear_cache.sh` über die Konsole des Servers aus oder lösche unter `var/cache` die `production_xxxx`-Ordner.

Sobald Du nun Deinen Shop aufrufst, wird das Cache-Verzeichnis automatisch neu erzeugt.

Rechte der Shopware-Verzeichnisse

Du kannst Dich nun im Backend einloggen. Rufe dort *Einstellungen* > *Systeminfo* auf und wechsel in den Reiter **Shopware-Verzeichnisse**. Falls hier Shopware-Ordner bemängelt werden, setze deren Rechte auf 0755. Setze Order- und Dateirechte außer beim Cache **niemals** auf 0777, da hiermit **jeder** Schreibzugriff auf Deine Daten hat, das ist unbedingt zu vermeiden! Nach diesem Schritt sind nochmals alle Caches (außer SEO- und Suchcache) im Backend zu leeren.

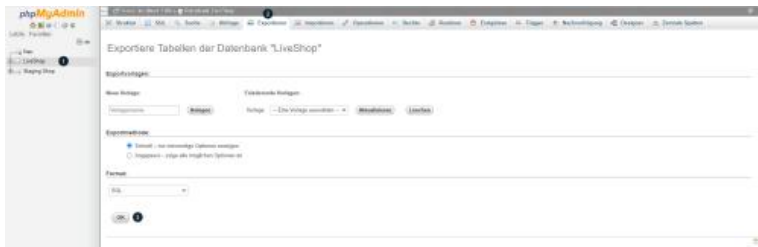
Live-System zu Testumgebung

Solltest Du Dir eine Testumgebung aufsetzen, um z.B. eigene Programmierungen, Templates etc. zu testen bzw. zu entwickeln, kannst Du die folgende Beschreibung nutzen.

Kopieren des Live-Systems

Öffne einen FTP-Client und transferiere die Dateien von Deinem **Web-Server** in das **Lokale Webserver-Verzeichnis** (Je nach Entwicklungsumgebung kann dies an unterschiedlichen Orten liegen, unter Linux in der Regel aber unter `/var/www`). Überschreibe dabei alle Dateien und Ordner.

MySQL-Datenbank auf lokalen MySQL-Server kopieren



In diesem Schritt wird beschrieben, wie Du einen kompletten Dump (Kopie) Deiner Produktiv-Datenbank anlegst und in Deine Testumgebung importierst. Öffne phpMyAdmin in Deiner Produktivumgebung und melde Dich dort an. Wähle dann die **Produktiv-Datenbank von Shopware (1)** aus und klicke danach in den Tab **Exportieren (2)** und starte nachfolgend den **Export Deiner Datenbank (3)**.

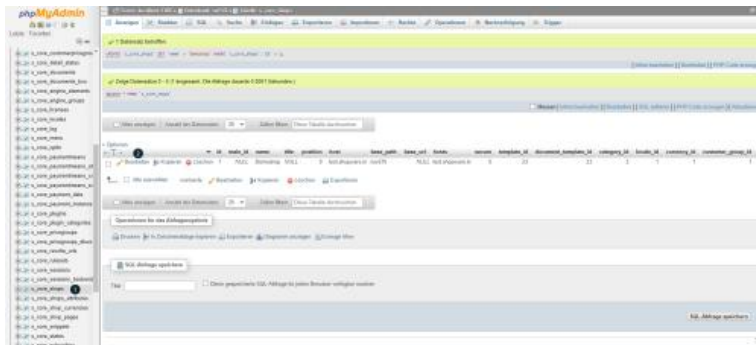
Wechsel nun in Deine lokale Testumgebung und rufe dort phpMyAdmin auf. Lege zuerst eine **neue Datenbank** mit zum Beispiel dem Namen **Testumgebung** an.



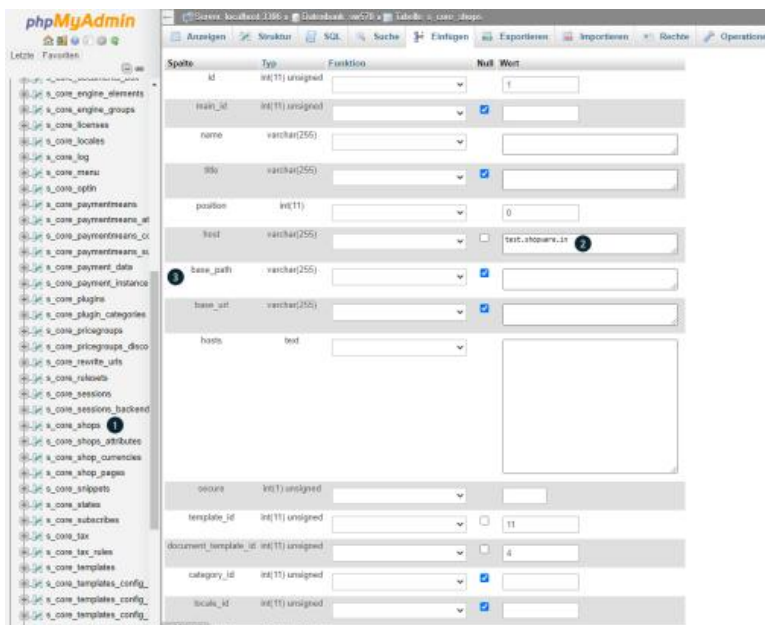
Wechsel anschließend in die Admin-Oberfläche der **neu angelegten Datenbank (1)** und **importiere (2)+(3)** die zuvor angelegte Export-Datei.

Es kann vorkommen, dass der zu importierende SQL-Dump zu groß ist und aufgrund dessen der Import abbricht. In diesem Fall kann dieser vor dem Import einfach gezippt werden. Falls der gezippte SQL-Dump immer noch zu groß ist, musst Du auf Import-Tools wie z.B. BigDump zurückgreifen. Mit diesen Tools können auch größer Datenbanken importiert werden.

Anpassen der Shop-URL



Damit Dein Shop in der lokalen Umgebung auch unter der Domain z.B.: "http://localhost" erreichbar ist, muss die Shop-URL in der Datenbank entsprechend angepasst werden. Wechsel in phpMyAdmin in die Tabelle **s_core_shops (1)** und editiere den **Host-Eintrag des Hauptshops (2)**.



Änder nun die Einträge **host (2)** und **base_path (3)**. In unserem Beispiel soll die Testumgebung direkt unter "http://localhost" erreichbar sein. Entsprechend muss der Eintrag **host** in localhost geändert werden. Da wir unseren Shop nicht in einem Unterverzeichnis betreiben, muss der Eintrag **base_path (3)** leer sein. Speichere danach die von Dir getätigten Änderungen ab.

Anpassung der config.php

Nun muss noch die Verbindung zur lokalen MySQL-Datenbank hergestellt werden. Öffne hierzu das lokale Hauptverzeichnis Deines Shops und editiere die Datei **config.php** und ersetze ggf. Daten:

```
'username' => 'root',
'password' => 'root',
'host' => 'localhost',
'port' => '3306',
'dbname' => 'testumgebung',
```

Du kannst Dich nun im Backend unter http://localhost/backend einloggen. Leere dort noch alle Caches außer den SEO- und Such-Cache.

Tipps & Tricks

Testsysteme vor der Indexierung durch Google schützen

Sofern Du Dein Testsystem nicht lokal, sondern auf einem Webserver im Internet aufgesetzt hast, so empfehlen wir Dir den Shop vor der Indexierung durch Suchmaschinen zu schützen.

Wartungsmodus aktivieren und für IP freischalten

Aktiviere in der Testumgebung den Wartungsmodus. Zusätzlich hast Du die Möglichkeit, in den Einstellungen für den Wartungsmodus Deine eigene IP-Adresse, mit welcher Du auf die Testumgebung

zugreifst, als Ausnahme zu hinterlegen. Wie Deine eigene IP-Adresse lautet, kannst Du auf der Seite www.meineip.de auslesen.

Testumgebung vor Crawlern schützen

Erstelle eine Datei namens robots.txt und lade diese in das Root-Verzeichnis deiner Shopware-Installation. Die robots.txt muss folgenden Inhalt haben:

```
User-agent: *  
Disallow: /
```

Somit werden alle Robots abgewiesen. Vergesse nicht, die Datei nach Umwandlung in die Live-Umgebung zu entfernen.

Shopware CLI

Shopware bietet einige nützliche Tools die über die Konsole ausgeführt werden können. Die sogenannten "CLI" Befehle können außerhalb von Skript-Limitierungen (Beispiel: Skriptlaufzeit des Servers) ausgeführt werden und sind daher bei der Verarbeitung großer Datenmengen empfehlenswert. Damit die Befehle ausgeführt werden können wird ein Terminal-Programm wie "PuTTY" benötigt, welches kostenlos erhältlich ist. Hier wird via "SSH" eine Verbindung zum Webserver hergestellt, auf dem die Befehle anschließend ausgeführt werden können.

Die Befehle können nach folgendem Schema ausgeführt werden:

```
bin/console <befehl> ODER php bin/console <befehl>
```

Viele Befehle enthalten ebenfalls eine Hilfefunktion, in der mögliche Parameter aufgelistet werden.

```
bin/console sw:plugin:list -h ODER bin/console sw:plugin:list --help
```

Nachfolgend einige wichtige Kommandos welche den Shopware Alltag erleichtern.

Thumbnails generieren

Ändert man die Standard Bildgrößen oder fügt man neue Bilder hinzu, ist es nötig, auch die Thumbnails neu zu generieren. Ein Problem hierbei ist, dass dies durch skriptseitige Beschränkungen über das Backend nur mit einer sehr geringen Stapelgröße möglich ist. Empfehlenswert ist daher die Generierung der Thumbnails über die Konsole, welche abhängig von der Anzahl der Bilder und verfügbarer Ressourcen des Servers einiges an Zeit in Anspruch nehmen kann. Grundsätzlich sollten die Thumbnails nicht regelmäßig neu generiert werden, da es sich hierbei um einen sehr performancelastigen Prozess handelt, der in der Regel nur einmalig bei Änderung der Thumbnailgrößen erforderlich ist.

```
php bin/console sw:thumbnail:generate
```

HTTP-Cache aufwärmen

Dieser Befehl ist besonders für die Verwendung als Cronjob hilfreich, da hierdurch die Ladezeiten des Shops nach Leeren des Caches deutlich verringert werden. Über den regulären Shopware Cronjob ist dieser Prozess nicht abgedeckt, weshalb dies als separater Cronjob in der Serverkonsole hinterlegt werden muss.

```
php bin/console sw:warm:http:cache
```

Durch den Zusatz "--help" kannst Du Dir alle verfügbaren Optionen anzeigen lassen, darunter die Option, um den bestehenden Cache vorher zu leeren oder die Aufrufe in mehrere Prozesse zu unterteilen.

Weitere Informationen & nützliche Kommandos für Cronjobs findest du im Artikel [Shopware Cronjobs](#).

Bitte beachte, dass dieser Prozess abhängig von der Artikel- bzw. Kategorieanzahl einiges an Zeit in Anspruch nehmen kann. Dieser Prozess wird für jeden (Sub-)Shop automatisch ausgeführt.

Kategoriebaum neu aufbauen

In manchen Fällen ist es erforderlich, den Kategoriebaum neu aufzubauen, jedoch sollte dies gerade bei vielen Kategorien/Artikeln nicht über das Backend erfolgen. Hier empfiehlt sich die Ausführung über die Konsole.

```
php bin/console sw:rebuild:category:tree
```

Bitte beachte, dass dieser Prozess abhängig von der Artikel- bzw. Kategorieanzahl einiges an Zeit in Anspruch nehmen kann. Dieser Prozess wird für jeden (Sub-)Shop automatisch ausgeführt.

Befehlsübersicht

Alle möglichen Befehle kannst Du dir über folgendes Kommando ausgeben lassen

```
php bin/console
```

Die wichtigsten Befehle haben wir für Dich hier aufgelistet:

Core Befehle

sw:thumbnail:cleanup	Entfernt ungenutzte Album Vorschaubilder.
sw:thumbnail:generate	Generiert ein neues Thumbnail.

sw:generate:attributes	Generiert die Attribut-Modelle.
sw:warm:http:cache	Wärmt den HTTP-Cache zur Performance-Steigerung auf.
sw:media:cleanup	Sammelt alle unbenutzten Artikelbilder und verschiebt diese in den Papierkorb
sw:snippets:find:missing	Findet die fehlenden Snippets in der Datenbank und schreibt diese in .ini Dateien.
sw:snippets:remove	Entfernt die Snippets aus der Datenbank für einen angegebenen Ordner.
sw:snippets:to:db	Lädt die Snippets aus dem "/snippets" Ordner im Hauptverzeichnis der Installation in die Datenbank. Zu finden ist der Ordner hier
sw:snippets:to:ini	Schreibt die Snippets von der Datenbank in .ini Dateien.
sw:snippets:to:sql	Lädt die Snippets von einer .ini Datei in eine SQL Datei.

Plugin Befehle

sw:plugin:activate	Aktiviert ein Plugin.
sw:plugin:config:list	Listet die Plugin Konfiguration auf.
sw:plugin:config:set	Setzt eine Plugin Konfiguration.
sw:plugin:deactivate	Deaktiviert eine Plugin Konfiguration.
sw:plugin:delete	Entfernt ein Plugin.
sw:plugin:install	Installiert ein Plugin.
sw:plugin:list	Listet alle Plugins auf.
sw:plugin:refresh	Aktualisiert die Plugin Liste.

sw:plugin:uninstall	Deinstalliert ein Plugin.
sw:plugin:update	Aktualisiert ein Plugin.

Eigene Befehle mit einem Plugin anbieten

Um eigene Befehle zu erstellen, schaue hierfür in unsere [DevDocs](#).

Shopware Updates Debuggen

Tabelle s_schema_version

Da die Deltas des Update per php-script eingespielt werden, wird die Ausführung der einzelnen Deltas sowie entsprechende Fehlermeldung beim Einspielen in der Tabelle s_schema_version geloggt. Fehlermeldungen tauchen in der Spalte **error_msg** auf. Steht hier ein *Null*, so wurde das Delta erfolgreich ausgeführt. Falls es einen Fehler beim Einspielen eines Deltas gibt, so wird die Fehlermeldung in die Spalte **error_msg** geschrieben.

Die Deltas findest Du unter /update-assets/migrations. Ob diese auch eingespielt wurden, kannst Du in der o.g. Datenbanktabelle kontrollieren. Überprüfe, ob für die Migration ein Eintrag vorhanden ist und ob die Felder *start_date* und *end_date* ein korrekter Timestamp ist (YYYY-MM-DD HH:MM:SS) und das Feld *error_msg* auf *Null* steht.

Falls es zu Problemen beim Einspielen der Deltas kommt, so empfehlen wir zuerst das komplette Datenbank-Backup zurückzuspielen und nachfolgend die aufgezeigten Fehler zu korriegern. Alternativ kannst Du das Delta auch per Hand einspielen (ausführen). In diesem Fall muss der entsprechende Eintrag in der s_schema_version gesetzt werden (s.o.). Sollte das Delta bereits korrekt in der Datenbank gesetzt sein, so kannst Du bei "error_msg" das NULL und bei "end_date" ein Enddatum setzen. Wird nun der Updater erneut ausgeführt, so wird dieses Delta übersprungen.

FAQ

Update unmöglich: Software-Subscription

Der Auto Updater kann die folgende Meldung anzeigen:

"Für ihre eingesetzte Shopware Version konnte keine aktive Software-Subscription ermittelt werden. Das Auto-Update kann aktuell nicht durchgeführt werden."

Hast Du eine aktive Subscription und erscheint diese Meldung dennoch, ist eine Lizenz in der Datenbank hinterlegt, die nicht aktuell ist. Prüfe dann folgende Schritte. Möglicherweise kannst Du nicht alle Schritte

prüfen, da der Lizenz-Manager nicht mehr vorhanden ist oder der Bereich "Shopware-Lizenz" lädt:

- Unter **Einstellungen > Grundeinstellungen > Weitere Einstellungen > Lizenz-Manager** dürfen nur aktuell gültige Lizenzträge vorhanden sein, die als Host auch die Domain haben, die aktuell im Shopware Account registriert ist
- Unter **Einstellungen > Grundeinstellungen > Weitere Einstellungen > Shopware-Lizenz** trage bitte die aktuelle Lizenz aus Deinem Shopware Account ein
- In der Tabelle **s_core_licenses** muss zunächst der Eintrag mit dem Label **Shopware Core** gelöscht werden. Die Shopware Lizenz sollte danach über das Backend unter **Grundeinstellungen > Weitere Einstellungen > Shopware-Lizenz** neu eingepflegt werden. Sofern dies das Verhalten nicht behebt, muss die Tabelle zudem auf abgelaufene Plugin-Lizenzen überprüft werden. Diese Einträge müssen dann ebenfalls gelöscht werden, damit ein Update durchgeführt werden kann.

Foreign-Key Errors

Frontend:

```
No mapping found for field '***' on class 'Shopware\Models\Attribute\***'. in vendor/doctrine
```

oder

```
Fatal error: Cannot redeclare Shopware\Models\Attribute\CustomerGroup::$customerGroup in /var/
```

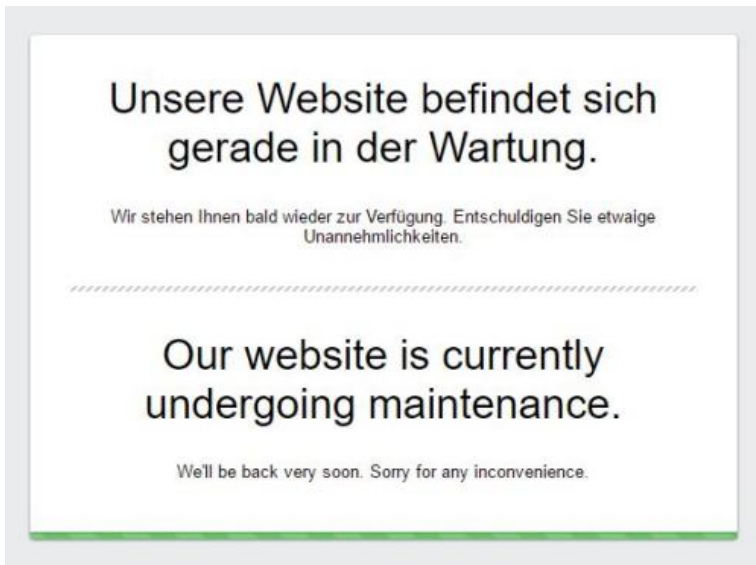
Backend:

```
Fatal error: Cannot redeclare Shopware\Models\Attribute\User::$user in /var/www/production/sho
```

Mögliche Ursache können defekte Beziehungen in den Datenbank-Tabellen sein, zur Korrektur kannst Du die [Foreign-Keys reparieren](#). Leere im Anschluss den Cache manuell auf Datei-Ebene unter `/var/cache` und `web/cache`!

Wartungsmodus nicht beendet

Sollte Dein Frontend oder Backend nach dem Update weiterhin den Wartungsmodus anzeigen, musst Du einen Ordner im Shopware-Verzeichnis löschen. Unten in der Beschreibung erklären wir Dir, welche dies sind.



Der Wartungsmodus sollte automatisch vom Updatescript von Shopware beendet werden. Falls dies nicht der Fall ist, hatte das Script nicht genügend Rechte um den Updateordner wieder zu löschen. Bitte gib dem Benutzer, mit dem der PHP-Prozess läuft, genügend Rechte um diese Aktion durchzuführen, damit das Update beim nächsten Mal komplett automatisch durchgeführt werden kann. Lösche bitte als nächstes die folgenden Ordner:

Beim Auto-Update: /files/update

Beim manuellen Update: /update-assets

Zuletzt angeschaut funktioniert nicht korrekt

Sollte der Artikelverlauf "zuletzt angeschaut" bei Dir nicht mehr korrekt funktionieren, so dass z.B. der Footer der Seite nicht mehr korrekt dargestellt werden kann, so kann die Ursache sein, dass das Template abgeleitet wurde und somit die Änderung aus dem Standard-Template nicht korrekt greifen kann.

Prüfe daher bitte folgende Datei: /templates/emotion_meintemplate/frontend/index/index.tpl

Ggf. ist dort noch folgender Inhalt enthalten:

```
{block name="frontend_index_footer"}
  {if $sLastArticlesShow}
    {action module=widgets controller=lastArticles}
  {/if}
{/block}
```

Dieser Teil ist dann durch folgenden neuen Code zu ersetzen:

```
{block name="frontend_index_footer"}
  {if $sLastArticlesShow && !$isEmotionLandingPage}
  {include file="frontend/plugins/index/viewlast.tpl"}
  {/if}
{/block}
```

Wir empfehlen, zusätzlich zu prüfen, ob die Datei: /templates/_default/frontend/plugins/index/viewlast.tpl dem Standard entspricht. Leere im Anschluss den Cache des Shops.

initTopSeller?offset=0&limit=15

Erhältst du die Meldung "*Internal Server Error: initTopSeller?offset=0&limit=15*" ist die Topseller-Box deaktiviert worden (chartInterval = 0). Diese Konfiguration kann unter Umständen beim Update zu diesem Fehler führen. Einfach vor dem Update die Topseller-Box aktivieren.

The target-entity Shopware\Models\Attribute

Eine Meldung wie "*The target-entity Shopware\Models\Attribute\... cannot be found in ...*" erscheint, wenn die Attribut Models nicht korrekt generiert werden konnten. Falls möglich leere den kompletten Cache im Backend oder verwende das Consolen-Command bin/console sw:generate:attributes

importDatabase?XXX - Forbidden

Bekommst du die Meldung "*importDatabase?offset=XXX&totalCount=XXX - Forbidden*" kontrolliere, ob mod_evasive die Requests ablehnt und deaktiviere dieses Modul gegebenenfalls.

Choose a smaller batch size

Erhältst Du die Meldung "*Choose a smaller batch size*" beim Auto-Update kontrolliere, ob im Ordner /files defekte Update-Dateien liegen (z.b. 0 Byte Dateien) und entferne diese.

Could not apply migration

Fehlermeldung:

```
Error
Received the following error message:
Could not apply migration: SQLSTATE[HY000]: General error: 1005 Can't create table 'database.#

Please try to fix this error and restart the update.
Response
{"valid":false,"errorMsg":"Could not apply migration: SQLSTATE[HY000]: General error: 1005 Can
```

In diesem Fall kann ein konkreter Migrationsschritt nicht ausgeführt werden. Die Migrationsschritte werden im Ordner update-assets/migrations oder auf [GitHub](#) mit der jeweiligen Migrationsnummer angezeigt. Die Nummer der Migration wird Dir in der Meldung ausgegeben und Du kannst dann in die Datei mit der passenden Nummer die einzelnen Queries sehen, die dieser Migrationsschritt ausführen würde.

Mögliche Ursachen:

- 1. Datenbank Strict-Mode noch aktiv.
- 2. Foreign Keys defekt ([zum Wiki Artikel](#)).
- 3. Prüfe, ob noch genug Speicherplatz frei ist.
- 4. Systemvoraussetzung: Datenbank bzw. MySQL Version wird nicht erfüllt.
- 5. Fehler beim Anwenden des SQL-Deltas (Überprüfung Tabelle s_schema_version - siehe oben).
- 6. Tabelle soll angelegt werden, ist aber bereits vorhanden, z.B. da nach einem fehlerhaften Update die Datenbank nicht zurückgesetzt wurde bevor das Backup zurückgespielt wird.

SmartyCompilerException

Eine Meldung wie diese '*SmartyCompilerException*' with message '*Syntax Error in template "/x/x/x.tpl" ... {s name='IndexXmlLang'} {/s} unknown tag "s"*' stammt von einem inkompatiblen Pluign oder Theme-anpassungen. Mit dem Debug-Modus kann die Exception komplett ausgelesen werden:

<https://developers.shopware.com/developers-guide/shopware-config/#exceptions>

Sitemap-Anpassungen über config.php

Du hast die Möglichkeit, über die config.php Ausschlüsse für die Sitemap zu definieren oder eigene URLs hinzuzufügen.

Die config.php findest Du im Shopware-Hauptverzeichnis auf dem Webserver.

Beachte bitte, dass durch fehlerhafte Eintragungen in der config.php ggf. Dein Shop nicht mehr (korrekt) aufgerufen werden kann. Daher empfehlen wir, die im folgenden beschriebenen Anpassungen, nur von versierten Anwendern durchführen zu lassen.

Eine zusätzliche URL hinzufügen

Um eine zusätzliche URL in die Sitemap zu integrieren, wird ein Eintrag vom Typ custom_urls hinzugefügt. Es werden folgende Informationen benötigt:

- url: Die gewünschte URL, die mit in die Sitemap aufgenommen werden soll
- priority: Priorität dieser URL im Vergleich zu anderen URL Deines Shops. Dies hat keine Auswirkungen auf den Vergleich zu anderen Websites/Shops. Gültige Eingaben sind Werte zwischen 0 und 1; der Standard ist 0.5
- changeFreq: Wie oft wird der Inhalt der Seite voraussichtlich wechseln. Möglich sind: always, hourly, daily, weekly, monthly, yearly, never
- lastMod: Datum, wann die Seite zuletzt geändert wurde. Format Y-m-d H:i:s

- shopId: Für welche Shop-ID soll die URL in die Sitemap geschrieben werden? 0 = alle Shops. Die IDs der Shops findest Du in der Datenbanktabelle "s_core_shops" in der Spalte "id".

Beispiel

Es soll die URL "meineUrl.de" für alle Shops eingebunden werden. Als Priorität wird 0.4 hinterlegt und die Änderung soll wöchentlich erfolgen.

```
'sitemap' => [
  'custom_urls' => [
    [
      'url' => 'meineUrl.de',
      'priority' => 0.4,
      'changeFreq' => 'weekly',
      'lastMod' => '2019-01-01 12:00:12',
      'shopId' => 0
    ]
  ]
],
```

Bestimmte URLs aus der Sitemap entfernen

Sollen bestimmte Seiten nicht in der Sitemap mit aufgeführt werden, kann dies über excluded_urls näher definiert werden.

Hierzu werden folgende Informationen benötigt:

- resource: Definiert, welcher Seitentyp ausgeschlossen werden soll. Möglich sind:
 - '\Shopware\Models\Article\Article::class': Ausschluss von Produkten.
 - '\Shopware\Models\Emotion\Emotion::class': Ausschluss von Einkaufswelten.
 - '\Shopware\Models\Article\Supplier::class': Ausschluss von Herstellerseiten.
 - '\Shopware\Models\Blog\Blog::class': Ausschluss von Blogartikeln.
 - '\Shopware\Models\Category\Category::class': Ausschluss von Kategorieseiten.
 - '\Shopware\Models\Site\Site::class': Ausschluss von Shopseiten.
- identifier: Die ID der ausgewählten Ressource. 0 = komplett ausgeschlossen. Die IDs der Entitäten aus der Ressource findest Du an folgenden Stellen
 - Artikel: Datenbanktabelle "s_articles" in der Spalte "id".
 - Einkaufswelten: Datenbanktabelle "s_emotion" in der Spalte "id".
 - Hersteller: Datenbanktabelle "s_articles_supplier" in der Spalte "id".
 - Blog: Datenbanktabelle "s_blog" in der Spalte "id"
 - Kategorie: Im Backend in der Kategorieverwaltung in der Überschrift "Allgemeine Einstellungen" im Eintrag System-ID. Oder in der Datenbanktabelle "s_categories" in der Spalte "id".
 - Shopseiten: Im Backend in der Shopseitenverwaltung (Inhalte > Shopseiten) im Menü auf der linken Seite in Klammern hinter dem Namen der Shopseite. Oder In der Datenbanktabelle "s cms_static" in der Spalte "id".

- shopId: Die ID des Shops, für den der Ausschluss gelten soll. 0 = Ausschluss für alle Shops. Die IDs der Shops findest Du in der Datenbanktabelle "s_core_shops" in der Spalte "id".

Beispiele

Produkt

Das Produkt mit der ID 123 soll nicht in der Sitemap vom Shop mit der ID 2 enthalten sein.

```
'sitemap' => [
  'excluded_urls' => [
    [
      'resource' => \Shopware\Models\Article\Article::class,
      'identifier' => '123',
      'shopId' => 2
    ],
  ],
],
```

Einkaufswelt

Die Einkaufswelt mit der ID 21 soll für alle Shops aus der Sitemap entfernt werden.

```
'sitemap' => [
  'excluded_urls' => [
    [
      'resource' => \Shopware\Models\Emotion\Emotion::class,
      'identifier' => '21',
      'shopId' => 0
    ],
  ],
],
```

Kategorie und Shopseiten kombiniert

Die Kategorie mit der ID 246 sowie alle Shopseiten sollen für alle Shops aus der Sitemap entfernt werden.

```
'sitemap' => [
  'excluded_urls' => [
    [
      'resource' => \Shopware\Models\Category\Category::class,
      'identifier' => '246',
      'shopId' => 0
    ],
    [
      'resource' => \Shopware\Models\Site\Site::class,
      'identifier' => '0',
      'shopId' => 0
    ],
  ],
],
```

SQL Tipps & Tricks

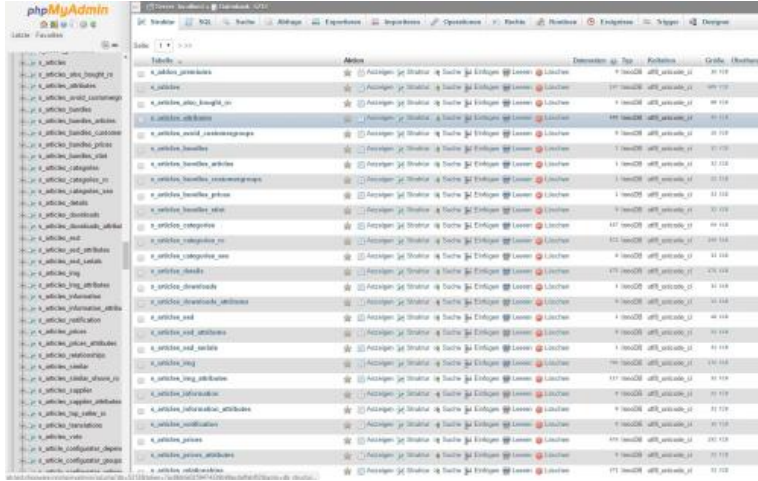
Es wird empfohlen diese Statements nur mit den nötigen Vorkenntnissen auszuführen.

Die hier zur Verfügung gestellten Informationen wurden zum Zeitpunkt der Veröffentlichung dieses Artikels

getestet, können sich zwischenzeitlich aber geändert haben. **Die Durchführung dieses Tutorials geschieht auf eigene Gefahr!**

Lege Dir vor der Durchführung ein **Backup** an, sodass Du dieses im Zweifel zurückspielen kannst. Beachte, dass es sich hier um Anpassungen von Shopware handelt und der Inhalt dieses Tutorials daher **nicht offiziell supportet** wird!

In diesem Artikel findest Du einige nützliche SQL-Statements, die Dir bei bestimmten Fehlern oder Anforderungen helfen. Für die Verwaltung der Datenbank empfehlen wir als Administrationsoberfläche phpMyAdmin. Diese ermöglicht die graphische Darstellung der Datenbank und erleichtert die Ausführung der Befehle. Natürlich können die unten genannten Beispiele mit jedem anderen MySQL-Client durchgeführt werden.



Kundendaten

Doppelte Kundennummern filtern

Sofern Kundennummern im System doppelt vorhanden sind, kann es zu Fehlern bzw. inkonsistenten Datensätzen führen. Mit diesem Statement kannst Du ganz leicht die doppelten Kundennummern anzeigen lassen.

```
SELECT userID, COUNT( userID ) AS number
FROM `s_user_billingaddress`
GROUP BY `userID`
HAVING (COUNT( userID ) >1
)
```

Artikel Daten

Doppelte Artikelnummern filtern

Über das Backend ist die Anlage von Artikeln mit bereits bestehender Artikelnummer nicht möglich. Schnittstellen oder Import-Skripte können diesen Mechanismus jedoch aushebeln, weshalb es in solchen Fällen zu Fehlern kommt. Mit diesem Statement lassen sich doppelt vorhandene Artikel schnell herausfinden:

```
SELECT ordernumber, COUNT( ordernumber ) AS anzahl
FROM `s_articles_details`
GROUP BY `ordernumber`
HAVING (COUNT( ordernumber ) >1 );
```

Doppelte SEO-URL's in der s_core_rewrite_urls filtern

Falls ein bestimmter SEO-Link in Deinem Shop bzw. Subshop nicht generiert werden kann, ist die Ursache oft ein bereits vorhandener Eintrag. Dieses Statement sortiert die Einträge und zeigt Dir mögliche Duplikate.

```
SELECT *
FROM `s_core_rewrite_urls`
WHERE main =1
GROUP BY `org_path`, `main`, `subshopID`, `path`
HAVING COUNT( * ) >1
```

Aktive Produkte filtern, welche im Frontend dennoch nicht angezeigt werden

In manchen Fällen (oft auch durch fehlerhafte Synchronisierung von der Warenwirtschaft) werden im Backend aktive Artikel im Frontend nicht ausgegeben. Hintergrund hierfür kann sein, dass der Eintrag in der "s_articles" Tabelle die Spalte "active = 1" hat, jedoch der assoziierte Eintrag in der "s_articles_details" diese Spalte auf "active = 0" hat. Mit diesem Statement lassen sich solche Artikel anzeigen.

```
SELECT a.name, ad.ordernumber, a.active AS aktiv1, ad.active AS aktiv2
FROM `s_articles` AS a JOIN s_articles_details AS ad ON a.id = ad.articleID
WHERE a.active <> ad.active
```

Defekte Artikel ohne Eintrag in der s_article_details ausgeben lassen

Durch Schnittstellen oder Plugins können Artikel defekt in der Datenbank angelegt sein. Ein Indiz hierfür ist der fehlende Eintrag in der "s_articles_details". Mit diesem Statement lassen sich solche Artikel anzeigen. Diese sollten umgehend neu angelegt werden.

```
SELECT *
FROM s_articles a LEFT JOIN s_articles_details ad ON ad.articleID = a.id
WHERE ad.id IS NULL
```

Kaputte Varianten / Varianten-Leichen entfernen

Mit diesem Statement lassen sich nicht (mehr) zugeordnete Varianten Optionen aufzeigen. Diese können für Dateninkonsistenzen und fehlende Darstellungen im Frontend sorgen.

configurator_set_id IS NOT NULL

```
DELETE d.*, atr.*
-- SELECT d.*, atr.*
FROM s_articles_details d
JOIN s_articles a ON d.articleID = a.id AND a.configurator_set_id IS NOT NULL
JOIN s_articles_attributes atr ON d.id = atr.articledetailsID
LEFT JOIN s_article_configurator_option_relations cr ON cr.article_id = d.id
WHERE cr.id IS NULL AND d.articleID = a.id;
```

```
DELETE r
-- SELECT r.*
FROM `s_article_configurator_option_relations` r LEFT JOIN s_articles_details d ON d.id = r.ar
```

configurator_set_id IS NULL und zusätzliche Prüfung auf verwaiste Varianten

```
DELETE d.*, atr.*
-- SELECT d.*,atr.*
FROM s_articles_details d
JOIN s_articles a ON d.articleID = a.id AND a.configurator_set_id IS NULL
JOIN s_articles_attributes as atr ON d.id = atr.articleDetailsID
LEFT JOIN s_article_configurator_option_relations cr ON cr.article_id = d.id
WHERE cr.id IS NULL AND d.articleID = a.id and d.kind = 2;
```

Varianten Vorauswahl setzen

Mit diesem Statement werden Variantenartikel, die keine Vorauswahl haben, automatisch mit der ersten Varianten ausgewählt. Ohne Vorauswahl ist keine Anzeige des Variantenartikels im Frontend möglich.

```
UPDATE s_articles a LEFT JOIN s_articles_details d ON d.id = a.main_detail_id
SET a.main_detail_id = ( SELECT id FROM s_articles_details WHERE articleID = a.id LIMIT 1 )
WHERE d.id IS NULL;
```

```
UPDATE s_articles a, s_articles_details d SET d.kind = 1 WHERE d.id = a.main_detail_id;
```

Fehlende Kategorie Zuweisungen in der s_article_categories_ro beheben

Sofern diese Zuweisung fehlt, werden Artikel im Frontend als "nicht verfügbar" angezeigt - mit diesem Statement wird die besagte Tabelle für alle Artikel neu befüllt. Gerade bei vielen Artikeln ist dies performanter als den Kategorie-Baum neu aufzubauen.

```
INSERT IGNORE INTO `s_articles_categories_ro` (`articleID`, `categoryID`, `parentCategoryID`)
SELECT ac.articleID, c2.id AS categoryID, c.id AS parentCategoryID FROM s_articles_categories
JOIN s_categories c ON c.id = ac.categoryID JOIN s_categories c2 ON c2.id = c.id OR c.path
LIKE CONCAT('%|', c2.id, '|%') ORDER BY ac.articleID, c.id, c2.id
```

Vorschaubilder für Vorschlagsuche korrigieren

Hiermit können fehlende Vorschaubilder in verschiedenen Bereichen (Artikelansicht im Backend, Vorschlagsuche, etc.) korrigiert werden. Dies ist in manchen Fällen nach einem Update von Shopware 4 auf Shopware 5 notwendig.

```
UPDATE s_articles_img i, s_media m
SET i.media_id = m.id
WHERE m.path = CONCAT('media/image/', i.img, '.', i.extension);
```

Eigenschaftensoptionen ohne Artikelzuordnung löschen

Hiermit können Eigenschaftensoptionen gelöscht werden die aktuell keinem Artikel zugewiesen sind. Dabei werden auch ggf. vorhandene Freitextfelder zu diesen Optionen gelöscht.

```
DELETE v, va
FROM `s_filter_values` v LEFT JOIN s_filter_articles a ON a.valueID = v.id
JOIN s_filter_values_attributes va ON v.id = va.valueID
WHERE a.valueID IS NULL;
```

Bestelldatensätze

Doppelte Bestellnummern auslesen

Doppelte Bestellnummern in Shopware führen zu Fehlern im Backend und sollten daher nicht vorhanden sein. Durch Plugins oder Schnittstellen kann es jedoch zu einem solchen Verhalten kommen. Mit diesem Statement kannst Du Dir doppelte Bestellnummern ausgeben lassen.

```
SELECT ordernumber, COUNT(ordernumber) AS anzahl
FROM s_order
GROUP BY ordernumber
HAVING ( COUNT(ordernumber) > 1 );
```

Alternativ besteht auch die Möglichkeit die Bestellungen in eine temporäre Order Tabelle zu kopieren und dann die doppelten Bestellnummern in der eigentlichen Tabelle zu kennzeichnen.

```
CREATE TABLE `s_order_tmp` LIKE `s_order`;
INSERT INTO `s_order_tmp` SELECT * FROM `s_order`;
UPDATE s_order
SET ordernumber = CONCAT(ordernumber , '-doppelt')
WHERE id IN (SELECT max(id) FROM s_order_tmp GROUP BY ordernumber
HAVING ( COUNT(ordernumber) > 1 )) AND ordernumber <> 0;
DROP TABLE `s_order_tmp`;
```

In diesem Fall wird an die doppelte Bestellnummer ein "-doppelt" angehängen.

Beachte hierbei, dass es sich bei Bestellungen mit der Bestellnummer "0" um Abbruchwarenkörbe handelt und diese Nummer mehrfach vergeben sein kann.

Bestellungen und Kunden aus Sicherung migrieren

Es kann vorkommen, dass durch ein Problem oder den Einsatz eines Test-Systems einige Bestellungen nicht in der Live-Datenbank vorhanden sind, die IDs der Bestellungen jedoch bereits durch andere neue Bestellungen wieder belegt sind. Über das Backend von Shopware ist es aktuell nicht möglich Bestellungen und Kunden aus einer Sicherung zu übernehmen, dieses Query sollte hier abhilfe schaffen.

Bitte beachte dabei, dass Dein System ggfs. noch weitere Spalten, Tabellen oder Anforderungen enthält, welche du ergänzen musst. Auch müssen Zahlungen hier ggfs. manuell abgewickelt werden.

Anschließend sollte noch das Query für doppelte Bestellnummern ausgeführt werden, damit die Bestellnummern nach der Migration im Backend überprüft werden können.

DBOLDNAME ersetzt du einfach mit dem Namen der Datenbank aus der du die Datensätze migrieren willst und die IdOfFirstCustomerToMigrate und NumberOfFirstOrderToMigrate durch die jeweilige ID bzw. Bestellnummer des ersten zu migrierenden Datensatzes.

```
ALTER TABLE s_user ADD idold INT(11) NULL AFTER customernumber;
INSERT INTO s_user (idold, password, encoder, email, active, accountmode, cONfirmationkey, pa
INSERT INTO s_user_attributes (userID) SELECT id FROM s_user WHERE id NOT IN (SELECT userID FR
INSERT INTO s_user_addresses (user_id, company, department, salutation, title, firstname, last
INSERT INTO s_user_addresses_attributes (address_id) SELECT id FROM s_user_addresses WHERE id
INSERT INTO `s_user_billingaddress` (`userID`, `company`, `department`, `salutation`, `firstna
```

```

INSERT INTO s_user_billingaddress_attributes (billingID) SELECT id FROM s_user_billingaddress
INSERT INTO `s_user_shippingaddress` (`userID`, `company`, `department`, `salutation`, `firstn
INSERT INTO s_user_shippingaddress_attributes (shippingID) SELECT id FROM s_user_shippingaddre
UPDATE s_user AS u JOIN s_user_addresses AS ua ON u.id = ua.user_ID SET u.default_billing_addr
UPDATE s_user AS u JOIN s_user_addresses AS ua ON u.id = ua.user_ID SET u.default_shipping_add
ALTER TABLE s_order ADD idold INT(11) NULL AFTER deviceType;
ALTER TABLE s_order_details ADD didold INT(11) NULL AFTER pack unit;
INSERT INTO s_order (idold, ordernumber, userID, invoice_amount, invoice_amount_net, invoice_s
INSERT INTO s_order_attributes (orderid, attributel, attribute2, attribute3, attribute4, attri
SELECT o.id, oold.attributel, oold.attribute2, oold.attribute3, oold.attribute4, oold.attribut
INSERT INTO s_order_details (didold, orderID, ordernumber, articleID, articleordernumber, pric
SELECT posold.id, o.id, posold.ordernumber, posold.articleID, posold.articleordernumber, posol
INSERT INTO s_order_details_attributes (detailID, attributel, attribute2, attribute3, attribut
SELECT d.id, dold.attributel, dold.attribute2, dold.attribute3, dold.attribute4, dold.attribut
INSERT INTO s_order_billingaddress (userID, orderID, company, department, salutation, customer
INSERT INTO s_order_billingaddress_attributes (billingID) SELECT id FROM s_order_billingadres
INSERT INTO s_order_shippingaddress (userID, orderID, company, department, salutation, firstna
SELECT u.id, o.id, oldship.company, oldship.department, oldship.salutation, oldship.firstname,
INSERT INTO s_order_shippingaddress_attributes (shippingid) SELECT id FROM s_order_shippingadd
ALTER TABLE `s_user` DROP `idold`;
ALTER TABLE `s_order` DROP `idold`;
ALTER TABLE `s_order_details` DROP `didold`;

```

System / Einstellungen

Nicht standardmäßig vorhandene Plugins deaktivieren

Dieses Statement sorgt dafür, dass alle Plugins (über den Plugin Manager hochgeladen, aus dem Community Store installiert) deaktiviert werden. Dies ist besonders für das Debuggen von Fehlern sehr hilfreich, da Du so schnell den Einfluss von Plugins ausschließen kannst.

```

CREATE TABLE `s_core_plugins_tmp` LIKE `s_core_plugins`;
INSERT INTO `s_core_plugins_tmp` SELECT * FROM `s_core_plugins`;

```

Nun setzt Du mit diesem Statement alle Plugins auf inaktiv.

```

UPDATE `s_core_plugins` SET `active` = 0 WHERE `source` <> 'Default' AND author <> 'shopware AG';

```

Für die Wiederherstellung der alten Plugins kopierst Du die Daten aus der temporären Tabelle in die reguläre und löschst im Anschluss die nicht mehr benötigte Tabelle.

```

UPDATE s_core_plugins AS p JOIN s_core_plugins_tmp AS pt ON p.id = pt.id SET p.active = pt.act
DROP TABLE s_core_plugins_tmp;

```

Datenbank zurücksetzen bzw. Daten löschen

Beachte immer dass Daten untereinander verknüpft sein können. Kunden und Artikel kommen in Bestellungen vor und können nach dem Löschvorgang ggfs. nicht mehr vollständig aufgerufen werden.

Hinweis zu Foreign-Keys

Je nach Version und Konfiguration des MySQL Servers kann es erforderlich sein, die Foreign-Key-Prüfung statt als Checkbox vor dem Absenden des Queries innerhalb des Queries auszuführen, in diesem Falle füge vor Deinem gewünschten Query die Zeile SET foreign_key_checks = 0; ein, um den Foreign-Key-Check zu deaktivieren und nach dem Query (also als neue letzte Zeile) SET foreign_key_checks = 1;;, um den

Foreign-Key-Check wieder zu aktivieren.

```
SET foreign_key_checks = 0;
TRUNCATE `s_user`;
TRUNCATE `s_user_addresses`;
TRUNCATE `s_user_addresses_attributes`;
TRUNCATE `s_user_attributes`;
TRUNCATE `s_user_billingaddress`;
TRUNCATE `s_user_billingaddress_attributes`;
TRUNCATE `s_user_shippingaddress`;
TRUNCATE `s_user_shippingaddress_attributes`;
SET foreign_key_checks = 1;
```

Kunden

Mit diesem Query leerst du alle Kundendaten komplett aus den Tabellen.

```
TRUNCATE `s_user`;
TRUNCATE `s_user_addresses`;
TRUNCATE `s_user_addresses_attributes`;
TRUNCATE `s_user_attributes`;
TRUNCATE `s_user_billingaddress`;
TRUNCATE `s_user_billingaddress_attributes`;
TRUNCATE `s_user_shippingaddress`;
TRUNCATE `s_user_shippingaddress_attributes`;
```

Mit diesem Query löscht du bestimmte Kundendaten, wobei du die WHERE-Bedingung selbst noch ergänzen musst.

```
DELETE aa.*, ad.*, ua.*, ba.*, ub.*, sa.*, us.*, u.*
FROM `s_user` AS u
LEFT JOIN s_user_addresses AS ad ON u.id = ad.user_id
LEFT JOIN s_user_addresses_attributes AS aa ON ad.id = aa.address_id
LEFT JOIN s_user_attributes AS ua ON u.id = ua.userID
LEFT JOIN s_user_billingaddress AS ub ON u.id = ub.userID
LEFT JOIN s_user_billingaddress_attributes AS ba ON ub.id = ba.billingID
LEFT JOIN s_user_shippingaddress AS us ON u.id = us.userID
LEFT JOIN s_user_shippingaddress_attributes AS sa ON us.id = sa.shippingID
WHERE XXXX;
```

Bestellungen

Mit diesem Query leerst du die gesamten Bestell-Tabellen.

```
TRUNCATE `s_order`;
TRUNCATE `s_order_attributes`;
TRUNCATE `s_order_basket`;
TRUNCATE `s_order_basket_attributes`;
TRUNCATE `s_order_billingaddress`;
TRUNCATE `s_order_billingaddress_attributes`;
TRUNCATE `s_order_comparisons`;
TRUNCATE `s_order_details`;
TRUNCATE `s_order_details_attributes`;
TRUNCATE `s_order_documents`;
TRUNCATE `s_order_documents_attributes`;
TRUNCATE `s_order_esd`;
TRUNCATE `s_order_history`;
TRUNCATE `s_order_notes`;
TRUNCATE `s_order_shippingaddress`;
TRUNCATE `s_order_shippingaddress_attributes`;
```

Mit diesem Query löscht du bestimmte Bestellungen, abhängig von der WHERE-Bedingung die du noch ergänzen musst.


```

DELETE osa.*, os.*, odda.*, odd.*, oda.*, od.*, oba.*, ob.*, oa.*, o.*
FROM s_order AS o LEFT JOIN s_order_attributes AS oa ON o.id = oa.orderID
LEFT JOIN s_order_billingaddress AS ob ON o.id = ob.orderID
LEFT JOIN s_order_billingaddress_attributes AS oba ON ob.id = oba.billingID
LEFT JOIN s_order_details AS od ON o.id = od.orderID
LEFT JOIN s_order_details_attributes AS oda ON od.id = oda.detailID
LEFT JOIN s_order_documents AS odd ON o.id = odd.orderID
LEFT JOIN s_order_documents_attributes AS odda ON odd.id = odda.documentID
LEFT JOIN s_order_shippingaddress AS os ON o.id = os.orderID
LEFT JOIN s_order_shippingaddress_attributes AS osa ON os.id = osa.shippingID
WHERE XXXX;

```

Artikel entfernen

Durch dieses Query leerst du die Artikel Tabellen komplett.

```

TRUNCATE `s_addon_premiums`;
TRUNCATE `s_articles`;
TRUNCATE `s_articles_also_bought_ro`;
TRUNCATE `s_articles_attributes`;
TRUNCATE `s_articles_avoid_customergroups`;
TRUNCATE `s_articles_categories`;
TRUNCATE `s_articles_categories_ro`;
TRUNCATE `s_articles_categories_seo`;
TRUNCATE `s_articles_details`;
TRUNCATE `s_articles_downloads`;
TRUNCATE `s_articles_downloads_attributes`;
TRUNCATE `s_articles_esd`;
TRUNCATE `s_articles_esd_attributes`;
TRUNCATE `s_articles_esd_serials`;
TRUNCATE `s_articles_img`;
TRUNCATE `s_articles_img_attributes`;
TRUNCATE `s_articles_information`;
TRUNCATE `s_articles_information_attributes`;
TRUNCATE `s_articles_notification`;
TRUNCATE `s_articles_prices`;
TRUNCATE `s_articles_prices_attributes`;
TRUNCATE `s_articles_relationships`;
TRUNCATE `s_articles_similar`;
TRUNCATE `s_articles_similar_shown_ro`;
TRUNCATE `s_articles_supplier`;
TRUNCATE `s_articles_supplier_attributes`;
TRUNCATE `s_articles_top_seller_ro`;
TRUNCATE `s_articles_translations`;
TRUNCATE `s_articles_vote`;
TRUNCATE `s_article_configurator_dependencies`;
TRUNCATE `s_article_configurator_groups`;
TRUNCATE `s_article_configurator_options`;
TRUNCATE `s_article_configurator_option_relations`;
TRUNCATE `s_article_configurator_price_variations`;
TRUNCATE `s_article_configurator_sets`;
TRUNCATE `s_article_configurator_set_group_relations`;
TRUNCATE `s_article_configurator_set_option_relations`;
TRUNCATE `s_article_configurator_templates`;
TRUNCATE `s_article_configurator_templates_attributes`;
TRUNCATE `s_article_configurator_template_prices`;
TRUNCATE `s_article_configurator_template_prices_attributes`;
TRUNCATE `s_article_img_mappings`;
TRUNCATE `s_article_img_mapping_rules`;
TRUNCATE `s_filter_articles`;

```

Durch dieses Query kannst du, abhängig von einer WHERE-Bedingung die du ergänzen musst, Artikel löschen.

```

DELETE aimr.*, aim.*, acta.*, actpa.*, actp.*, act.*, acor.*, av.*, atr.*, atop.*, ASiro.*, AS
FROM s_articles AS a
LEFT JOIN s_articles_also_bought_ro AS bro ON a.id = bro.article_id
LEFT JOIN s_articles_details AS ad ON a.id = ad.articleID
LEFT JOIN s_articles_attributes AS aa ON ad.id = aa.articledetailsID
LEFT JOIN s_articles_avoid_customergroups AS acu ON a.id = acu.articleID
LEFT JOIN s_articles_categories AS ac ON a.id = ac.articleID
LEFT JOIN s_articles_categories_ro AS acr ON a.id = acr.articleID

```

```

LEFT JOIN s_articles_categories_seo acs ON a.id = acs.article_id
LEFT JOIN s_articles_downloads AS ado ON a.id = ado.articleID
LEFT JOIN s_articles_downloads_attributes AS adoa ON ado.id = adoa.downloadID
LEFT JOIN s_articles_esd AS aesd ON a.id = aesd.articleID
LEFT JOIN s_articles_esd_attributes AS aesda ON aesd.id = aesda.esdID
LEFT JOIN s_articles_esd_serials AS aesds ON aesd.id = aesds.esdID
LEFT JOIN s_articles_img AS ai ON a.id = ai.articleID
LEFT JOIN s_articles_img_attributes AS aia ON ai.id = aia.imageID
LEFT JOIN s_articles_information AS ain ON a.id = ain.articleID
LEFT JOIN s_articles_information_attributes AS aina ON ain.id = aina.informationID
LEFT JOIN s_articles_notification AS an ON ad.ordernumber = an.ordernumber
LEFT JOIN s_articles_prices AS ap ON a.id = ap.articleID
LEFT JOIN s_articles_prices_attributes AS apa ON ap.id = apa.priceID
LEFT JOIN s_articles_relationships AS ar ON a.id = ar.articleID
LEFT JOIN s_articles_similar AS ASi ON a.id = ASi.articleID
LEFT JOIN s_articles_similar_shown_ro AS ASiro ON a.id = ASiro.article_id
LEFT JOIN s_articles_top_seller_ro AS atop ON a.id = atop.article_id
LEFT JOIN s_articles_translations AS atr ON a.id = atr.articleID
LEFT JOIN s_articles_vote AS av ON a.id = av.articleID
LEFT JOIN s_article_configurator_option_relations AS acor ON ad.id = acor.article_id
LEFT JOIN s_article_configurator_templates AS act ON a.id = act.article_id
LEFT JOIN s_article_configurator_template_prices AS actp ON act.id = actp.template_id
LEFT JOIN s_article_configurator_template_prices_attributes AS actpa ON actp.id = actpa.templa
LEFT JOIN s_article_configurator_templates_attributes AS acta ON act.id = acta.template_id
LEFT JOIN s_article_img_mappings AS aim ON ai.id = aim.image_id
LEFT JOIN s_article_img_mapping_rules AS aimr ON aim.id = aimr.mapping_id
WHERE XXXX;

```

Nummernkreise zurücksetzen

Bei dieser Ausführung werden die Nummernkreise in Shopware gelöscht und mit Standardwerten wieder angelegt.

```

DROP TABLE `s_order_number`;

CREATE TABLE IF NOT EXISTS `s_order_number` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `number` int(20) NOT NULL,
  `name` varchar(255) COLLATE utf8_unicode_ci NOT NULL,
  `desc` varchar(255) COLLATE utf8_unicode_ci NOT NULL,
  PRIMARY KEY (`id`),
  UNIQUE KEY `name` (`name`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_unicode_ci AUTO_INCREMENT=10 ;

INSERT INTO `s_order_number` (`id`, `number`, `name`, `desc`) VALUES
(1, 20000, 'user', 'Kunden'),
(2, 20000, 'invoice', 'Bestellungen'),
(3, 20000, 'doc_1', 'Lieferscheine'),
(4, 20000, 'doc_2', 'Gutschriften'),
(5, 20000, 'doc_0', 'Rechnungen'),
(6, 10000, 'articleordernumber', 'Artikelbestellnummer '),
(7, 10000, 'sSERVICE1', 'Service - 1'),
(8, 10000, 'sSERVICE2', 'Service - 2'),
(9, 110, 'blogordernumber', 'Blog - ID');

```

Statistiken zurücksetzen

Einige Statistiken werden aus den Bestellungen und Kundendaten ermittelt. Die weiteren Werte werden mit diesem Befehl zurückgesetzt.

```

TRUNCATE `s_statistics_currentusers`;
TRUNCATE `s_statistics_pool`;
TRUNCATE `s_statistics_referer`;
TRUNCATE `s_statistics_search`;
TRUNCATE `s_statistics_visitors`;

```

Kategorien und Einkaufswelten zurücksetzen

Mit dieser Ausführung werden alle Kategorien sowie angelegten Einkaufswelten aus dem Shop entfernt. Es wird automatisch die Hauptkategorie "Deutsch" mit der System-ID 3 wieder angelegt.

```
TRUNCATE `s_categories`;
TRUNCATE `s_categories_attributes`;
TRUNCATE `s_categories_avoid_customergroups`;
TRUNCATE `s_emotion`;
TRUNCATE `s_emotion_attributes`;
TRUNCATE `s_emotion_categories`;
TRUNCATE `s_emotion_element`;
TRUNCATE `s_emotion_element_value`;

INSERT INTO `s_categories` (`id`, `parent`, `path`, `description`, `position`, `left`, `right`)
(1, NULL, NULL, 'Root', 0, 0, 0, 0, '2012-07-30 15:24:59', '2012-07-30 15:24:59', NULL, NULL,
(3, 1, NULL, 'Deutsch', 0, 0, 0, 0, '2012-07-30 15:24:59', '2012-07-30 15:24:59', NULL, NULL,

INSERT INTO `s_categories_attributes` (`id`, `categoryID`, `attribute1`, `attribute2`, `attrib
(1, 3, NULL, NULL, NULL, NULL, NULL, NULL);
```

SSL - Zertifikat im Shop einrichten

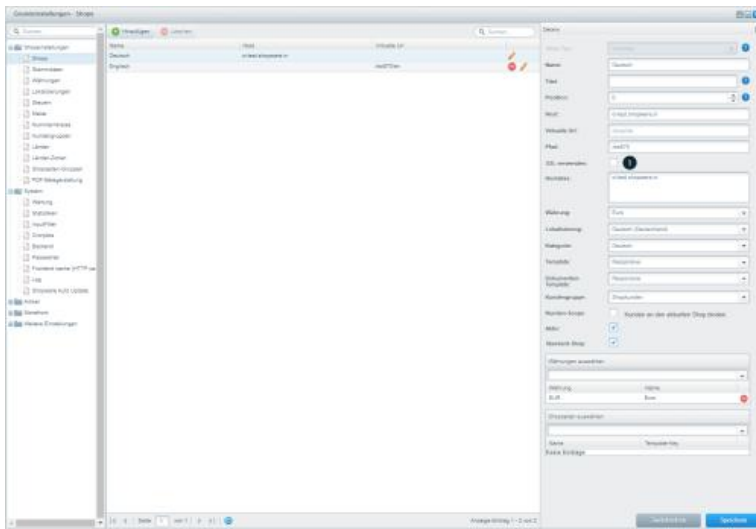
SSL / TLS

SSL / TLS steht allgemeingültig für ein Netzwerkprotokoll zur sicheren Übertragung von Daten. TLS ist hierbei der Nachfolger von SSL. Wenn Du die Verbindungen zum Shop mit SSL / TLS absicherst, werden die Daten zwischen Shop und dem verbundenen Benutzer verschlüsselt übertragen und bieten so weit weniger Angriffsfläche als eine unverschlüsselte Verbindung.

Welche Technologie Du einsetzen sollst, kannst Du selbst entscheiden, die aktuellen Browser geben hier aber schon diverse Empfehlungen ab, so sollte ab 2017 kein SHA-1 Algorithmus mehr verwendet werden, da dieser nicht mehr als sicher eingestuft wird. Grundsätzlich ist es aber empfehlenswert, auf aktuelle Technologien zu setzen wie z.B.: TLS 1.2 mit SHA-2 Algorithmus. Welches Zertifikat Du nutzt, ist für Shopware völlig nebensächlich, Hauptsache es ist technisch korrekt erstellt und installiert.

SSL aktivieren

Sobald das Zertifikat installiert ist, muss in Shopware nur noch die Verschlüsselung aktiviert werden. Wie das geht, erklären wir Dir hier.



Öffne deine Shopeinstellungen unter "*Einstellungen > Grundeinstellungen > Shopeinstellungen > Shops*" und wähle den Shop aus, für den die Verschlüsselung aktiviert werden soll. Aktiviere hier die Option **SSL verwenden**, um sämtliche Verbindungen ins Frontend zu verschlüsseln.

Umleitung auf HTTPS

Wenn Du alle eingehende HTTP Aufrufe direkt auf HTTPS umleiten willst, so kannst Du die htaccess Datei anpassen, welche sich im Hauptverzeichnis des Shops befindet:

```
RewriteCond %{HTTPS} !=on
RewriteRule (.*) https://%{HTTP_HOST}%{REQUEST_URI} [L,R=301]
```

Beispiel:

- `http://foo.example.com -> https://foo.example.com`
- `http://bar.example.com -> https://bar.example.com`

Umleitung von Subdomains

```
RewriteCond %{HTTPS} !=on
RewriteRule (.*) https://secure.example.com%{REQUEST_URI} [L,R=301]
```

Beispiel:

- `http://foo.example.com -> https://secure.example.com`
- `http://bar.example.com -> https://secure.example.com`

Spezielle Domain

Benötigst Du eine speziellere Regel für eine bestimmte Domain, so kann auf folgende Regel zurückgegriffen werden. Dieses Beispiel zeigt, wie HTTP Aufrufe für Domain `http://unsecure-domain.com` auf HTTPS von Domain `https://secure-domain.com` umgeleitet werden.

```
RewriteCond %{HTTPS} !=on
RewriteCond %{HTTP_HOST} (www\.)?unsecure-domain\.com [NC]
RewriteRule ^(.*)$ https://secure-domain\.com%{REQUEST_URI} [R=301,L]
```

Suche Debuggen

Dein Shop liefert Dir bei der Suche nicht die Ergebnisse, die Du erwartest? Dann bist Du hier genau richtig! Wir werden einige Punkte überprüfen an denen Du schnell feststellen kannst, ob die Shopware Suche auch genau das macht, was Du möchtest.

Bitte beachte, dass die Suche Dir keine Ergebnisse liefern kann, wenn mehr als $\approx 80\%$ Deines Artikelstammes als Ergebnis zurückgeliefert werden würde.

Suchindex überprüfen

Zunächst ist hier wichtig festzustellen, ob der Suchindex vollständig aufgebaut wurde. Dies kannst Du am besten direkt in der Datenbank feststellen. Gehe hierzu in Deine Datenbank und suche nach der Tabelle `s_search_keywords`. Befinden sich hier Einträge für die Spalte "Soundex" mit dem Wert NULL die keine Zahl als Eintrag haben, so ist dies ein Indiz dafür, dass der Suchindex nicht korrekt aufgebaut wurde.

+ Optionen

	id	keyword	soundex	cologne_phonetic	metaphone
<input type="checkbox"/> Bearbeiten <input type="checkbox"/> Kopieren <input type="checkbox"/> Löschen	1	seat	NULL		NULL
<input type="checkbox"/> Bearbeiten <input type="checkbox"/> Kopieren <input type="checkbox"/> Löschen	2	bmw	NULL		NULL
<input type="checkbox"/> Bearbeiten <input type="checkbox"/> Kopieren <input type="checkbox"/> Löschen	3	test	NULL		NULL
<input type="checkbox"/> Bearbeiten <input type="checkbox"/> Kopieren <input type="checkbox"/> Löschen	4	vw	NULL		NULL
<input type="checkbox"/> Bearbeiten <input type="checkbox"/> Kopieren <input type="checkbox"/> Löschen	5	audi	NULL		NULL
<input type="checkbox"/> Bearbeiten <input type="checkbox"/> Kopieren <input type="checkbox"/> Löschen	6	fiat	NULL		NULL
<input type="checkbox"/> Bearbeiten <input type="checkbox"/> Kopieren <input type="checkbox"/> Löschen	7	venom	NULL		NULL
<input type="checkbox"/> Bearbeiten <input type="checkbox"/> Kopieren <input type="checkbox"/> Löschen	8	vario	NULL		NULL
<input type="checkbox"/> Bearbeiten <input type="checkbox"/> Kopieren <input type="checkbox"/> Löschen	9	s	NULL		NULL
<input type="checkbox"/> Bearbeiten <input type="checkbox"/> Kopieren <input type="checkbox"/> Löschen	10	titanium	NULL		NULL
<input type="checkbox"/> Bearbeiten <input type="checkbox"/> Kopieren <input type="checkbox"/> Löschen	11	carbon	NULL		NULL

Strategie für den Aufbau des Such-Index

Um das beschriebene Verhalten genauer analysieren zu können solltest Du zunächst überprüfen wie der Suchindex in Deinem Shop aufgebaut wird. Hierzu schaust du unter **Einstellungen > Cash/Performance > Einstellungen (Reiter) > Suche (Links)** was hier als entsprechende Strategie für den Aufbau für den Suchindex ausgewählt ist.

Steht hier für den Aufbau von dem Such-Cache die Option "Live", so generiert diese bei jedem Seitenaufruf im Frontend eine bestimmte Anzahl von Einträgen bzw. überprüft diese auf Aktualität. Dementsprechend wird in Deinem Shop der Such-Index sehr häufig aktualisiert. Gerade da die Intelligente Suche eine "lernende" Suche ist, wären hier einige Suchanfragen notwendig damit optimale Suchergebnisse geliefert werden können. Während dieser Zeit könnten suboptimale Suchergebnisse auftreten. Die Aktualisierungsstrategie "Live" hat darüber hinaus den Nachteil, dass bei jedem Aufruf im Frontend zusätzliche Serverlast erzeugt wird. In Hinblick auf die Performance ist dieser Einstellung entsprechend nicht empfohlen.

Um dieses Verhalten zu begrenzen, empfehlen wir Dir die Umstellung der Suchstrategie auf "Cronjob". Mit der Nutzung von Cronjobs kannst Du die Last beispielsweise auch gesammelt auf Nachts verschieben.

Sollte auch dies wiedererwartend nicht weiterhelfen, so empfehlen wir Dir, dass Du versuchst den Suchindex über die Konsole aufzubauen: `php bin/console sw:refresh:search:index`

Bekommst Du hier eine Meldung ähnlich:

```
PHP Fatal error: Allowed memory size of 536870912 bytes exhausted (tried to allocate 72 bytes)
```

So solltest Du versuchen das `memory_limit` höher zu setzen. Je nach System funktioniert dies bereits mit folgender Erweiterung von dem CLI Call mit: `-d memory_limit=4096M` Somit: `php -d memory_limit=4096M bin/console sw:refresh:search:index`

Wenn sich der Suchindex aufgrund einer Standard Batch-Size von 4000 nicht aufbauen lässt, kannst Du dies schnell und einfach umgehen, indem Du die `config.php` auf Deinem Server anpasst.

Setze hierfür folgendes in Deine `config.php` ein:

```
'search' => [
  'indexer' => [
    'batchsize' => 3250,
  ],
],
```

Bei weiteren Fragen zu dem `memory_limit` kannst Du Dich am besten direkt an Deinen Hosting-Partner wenden.

Debuggen mit der Vorschau der Intelligente Suche

Findest Du mit Deiner Suche nicht die gewünschten Ergebnisse, so kannst Du hier am besten die Vorschau der Intelligenten Suche benutzen. Sollten die Informationen der Vorschau mit den Angaben von dem Faktor und der Relevanz Dir nicht die gewünschten Informationen liefern, so kannst Du die Developer Tools von Google Chrome nutzen.

Hier findest Du unter den Network einen List-Eintrag.

DeinShop.de/backend/SwagFuzzyPreview/list
?_dc=1486566464437&shopId=1&page=1&start=0&limit=20&filter=%5B%7B%22property%22%3A%22search%22

```
▼ {success: true, data: [{(name: "Main product with properties", stock: 50, shortDescription: "",-,-,-)],-}
▶ data: [{(name: "Main product with properties", stock: 50, shortDescription: "",-,-,-)]
▼ keywords: [{(id: "2589", term: "main", word: "main", relevance: "400"),-}]
▼ 0: {id: "2589", term: "main", word: "main", relevance: "400"}
  id: "2589"
  relevance: "400"
  term: "main"
  word: "main"
▼ 1: {id: "3", term: "main", word: "mail", relevance: 75}
  id: "3"
  relevance: 75
  term: "main"
  word: "mail"
▼ 2: {id: "2623", term: "main", word: "men", relevance: 50}
  id: "2623"
  relevance: 50
  term: "main"
  word: "men"
success: true
total: 9
```

Anhand von diesem Eintrag siehst Du schnell was der eigentliche Suchbegriff ist und welche Suchbegriffe gefunden und genutzt werden. Zusätzlich bekommst Du hier eine Übersicht, wieviele Suchbegriffe gefunden wurden. Würdest Du nun hier vermuten dass dein Suchbegriff ziemlich nah an den angegebenen Suchbegriffen ist jedoch trotzdem nicht als Term aufgelistet ist, so solltest Du hier Den Wert "Maximale Anzahl der angezeigten Ergebnisse und der genutzten Keyboards" für die Suche erhöhen. Hier wird festgelegt wie viel abweichende Keyboards zusätzlich für die Suche verwendet werden.

Artikel mit langen Artikelnamen werden nicht gefunden

Hast Du lange Artikelnamen, die Du nicht findest, so kannst Du für diese langen zusammengesetzten Artikelnamen einzelne Keywords anlegen. Zusätzlich solltest Du dann bei den Relevanzen diese für die Keyword erhöhen.

Beispiel: Donaudampfschiffahrtsgesellschaftskapitänsmütze

Keywords: Donau dampf schiff fahrts gesellschaft kapitän mütze Donaudampf schiffahrt gesellschaft kapitänsmütze Donaudampfschiffahrt gesellschaft schiffahrtgesellschaft kapitänsmütze

Suchlogik überprüfen

Findest Du Artikel nicht bei der Nutzung mehrerer Keywords nicht, so kannst Du am besten die verwendete Suchlogik überprüfen.

Diese findest Du unter: **Einstellungen > Grundeinstellungen > Storefront > Suche > "Und" Suchlogik verwenden: Nein**

Anschließend kannst Du den Such-Index neu aufbauen über: **Einstellungen > Cache / Performance > (Reiter) Einstellungen > Suche > Such-Index neu aufbauen .**

Wenn dies durchgeführt ist. Leerst Du den Cache (**STRG + ALT + X**) und lädst das Backend neu (**STRG + F5**) , danach kannst Du die Suchergebnisse in der Vorschau der Intelligenten Suche neu überprüfen.

Artikelnummern überprüfen

Allgemein ist wichtig bei der Suche darauf zu achten, dass Du korrekte Artikelnummern einsetzt. Hinweise zu den in Shopware unterstützten Artikelnummern findest Du in dem [Wiki-Artikel zu Artikelnummern](#).

Bei mehreren Suchergebnissen wird direkt zu einem Artikel gewechselt.

In diesem Fall wird eine defekte Artikelnummer vorliegen. Hierzu kannst Du am besten in der Datenbank in der Tabelle `s_articles_details` nachschauen, ob hier ein Artikel vorhanden ist, der keine oder nur 2 Zeichen hat als ordernumber hat.

Allgemeine Fragen zur Einrichtung der Intelligenten Suche

Informationen hierzu findest Du im [Wiki Artikel für die Intelligente Suche](#).

Systemanforderungen / Hosting

Server

Die benötigte Server-Hardware ist abhängig von der Anzahl der Artikel, die im Shop angeboten werden sollen und dem erwarteten Traffic (Besucher / Tag). Die Anforderungen werden von den meisten Hosting-Providern erfüllt und zertifizierte Hoster sollten hier eine geeignete Konfiguration zur Verfügung stellen.

Minimale Systemanforderungen (ab Shopware 5.7)

- Linux-basiertes Betriebssystem mit Apache 2.2 oder 2.4 Webserver mit mod_rewrite Modul und die Möglichkeit die Webserver Einstellungen in der .htaccess zu ändern.
- PHP 7.4 / 8.0 / 8.1 (seit 5.7.8) / 8.2 (seit 5.7.17)
7.4.14 und 8.0.1 sind nicht kompatibel
- MySQL >= 5.7 (außer 8.0.20, 8.0.21 und 8.0.29)
- Möglichkeit Cronjobs einzurichten
- Mindestens 4 GB freier Speicherplatz
Der Support von PHP 7.4 wurde von PHP bereits eingestellt. PHP 8.0 wird bald eingestellt. Weitere Informationen hierzu findest Du [hier](#).

PHP-Extensions / Webserver

- Apache Mod-Rewrite
- GD Library ab Version 2.0
- installierte cURL-Library
-

- Character type checking (ctype)
- Document Object Model (dom)
- HASH Message Digest Framework(hash)
- iconv
- JavaScript Object Notation (json)
- Multibyte String (mbstring)
- OpenSSL
- Session Handling (session)
- SimpleXML
- xml
- zip
- zlib
- PDO/MySQL
- INTL

Weitere Kompatibilitäten

- Varnish Version 5.0 oder höher
- Node-js Version 12 oder höher
- Redis 5 oder höher
- Elasticsearch 7.x

PHP OPcache

Es wird aus Performance-Gründen dringend empfohlen zu überprüfen, ob der PHP OPcache aktiviert ist.

Minimale Webserver/PHP-Einstellungen

- memory_limit > 256M
- upload_max_filesize >6M
- allow_url_fopen -> aktiviert

Empfohlene Einstellungen

- [APCu](#)
- Wenn die Shopware ESD Funktionen genutzt werden sollen, wird dringend die Nutzung von Apache **mod_xsendfile** empfohlen.

Optionale MySQL Einstellungen

- MySQL Variable **group_concat_max_len** sollte auf 2048 hochgesetzt werden. (standard: 1024)
- **innodb_buffer_pool_size** sollte so groß sein wie die komplette Datenbank.
- **query_cache_size** sollte normalerweise zwischen 100 - 200MB liegen.

Beachte bitte, dass die Werte hier als absolutes Minimum zu sehen sind. Sobald Du eine hohe Anzahl an Artikeln im Shop hast oder ein hoher Traffic zu erwarten ist und der Shop somit viele Anfragen gleichzeitig verarbeiten soll, ist es dringend zu empfehlen, die Serverressourcen zu erhöhen.

Alternatives Setup

Die oberhalb aufgeführten Anforderungen stellen das empfohlene Setup dar, um Shopware zu installieren. Generell ist der Betrieb auch auf ähnlichen Setups möglich (Mac OS, nginx, MariaDB, etc.). Bitte berücksichtige aber, dass wir keinen offiziellen Support für die Einrichtung solcher Server-Systeme und Setups anbieten können. Die Einrichtung alternativer Server-Systeme mit allen notwendigen Voraussetzungen obliegt dem jeweiligen Hosting-Partner.

Elasticsearch

Shopware unterstützt im Standard [Elasticsearch](#) Server. Weitere Informationen dazu findest Du [hier](#).

Client

Die Administration Deines Shops kann komplett online über den Webbrowser erfolgen. Hierzu ist es notwendig, dass Du Javascript und Cookies aktiviert hast. Durch den Einsatz von Cookies ist eine eindeutige Identifizierung Deiner Person möglich, somit wird ein hohes Sicherheitslevel gewährleistet. Durch den Einsatz von Javascript erhöht sich die Usability des Systems enorm und ermöglicht in Verbindung mit Ajax als Kerntechnologie von Shopware die komfortable und schnelle Administration Deines Shops.

Anforderungen

- In aktueller Version: Mozilla Firefox, Chrome, Safari oder Internet Explorer ab Version 11
- Javascript und Cookies aktiviert
- 4 GB Arbeitsspeicher
- Dual-Core CPU
- Bildschirmauflösung für das Backend: Min. 1366 x 768 Px.

Trackingnummern und Sendungsverfolgung

Vorwort

Dieses Tutorial zeigt Dir, wie Du Deinen Kunden einen Tracking-Link innerhalb der E-Mail-Templates zur Verfügung stellst. Somit können Deine Kunden sofort sehen, wo sich die Sendung gerade befindet.

Tracking-Link des Logistikpartners

Finde die benötigte Tracking-URL Deines Logistikpartners heraus. In diesem Beispiel erklären wir, wie Du einen Tracking-Link für GLS herausfindest. Dieses Beispiel ist auf fast alle anderen Logistikunternehmen übertragbar.

Auf der Internetseite der GLS-Group findest Du ein **Formular (1)**, welches Dir die Eingabe einer Tracking-ID/Code ermöglicht. Gib hier eine **gültige Paketnummer (2)** ein:



Nachdem Du eine gültige Paketnummer eingegeben hast, erscheint nun die Sendungsverfolgung des Logistik-Unternehmens. Interessant ist hier der Aufbau der URL:



In diesem Fall lautet die Tracking-URL

```
https://gls-group.eu/DE/de/paketverfolgung?match=
```

Die Paketnummer 1234567890 wird später durch die Variable ersetzt.

E-Mail-Vorlage um Tracking-URL erweitern

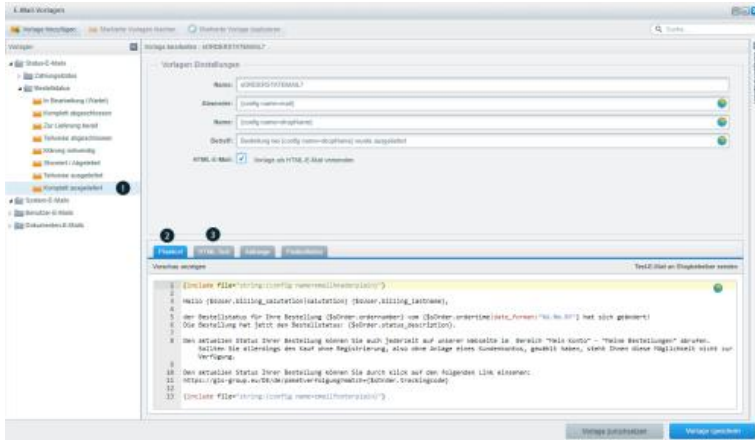
Wechsle nun in die E-Mail-Vorlagen und wähle das gewünschte Template aus. In unserem Fall haben wir uns für die E-Mail-Vorlage **Komplett ausgeliefert (1)** entschieden.

Hier haben wir mit dem Tracking-Link und der benötigten Variable für die Paketnummer hinterlegt.

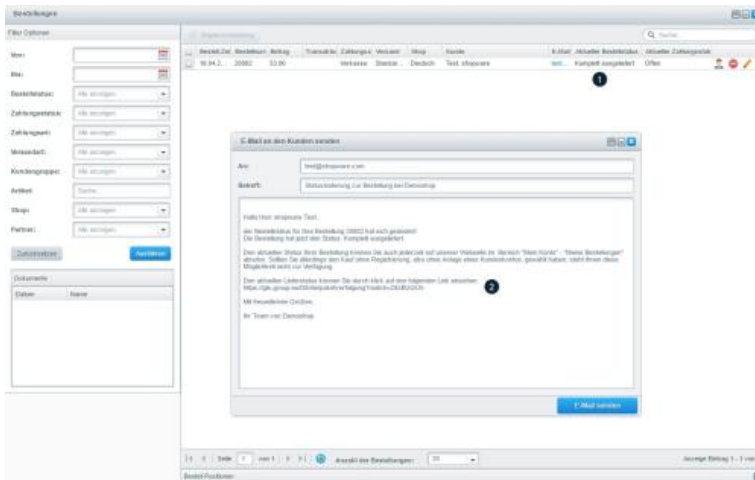
Link (Beispiel GLS) für **Text-E-Mails (2)**:

```
Sendungsverfolgung: https://gls-group.eu/DE/de/paketverfolgung?match={\$sOrder.trackingcode}
```

Ab Shopware 5.2 ist es möglich, die Status-E-Mails auch im **HTML-Format (3)** zu versenden.



Wenn Du nun in den Bestellungen den Status der Bestellung unter **Aktueller Bestellstatus (1)** auf **Komplett ausgeliefert** setzt, erscheint automatisch die in Schritt 2 angepasste E-Mail-Vorlage mit dem entsprechenden **Tracking-Link (2)**.



Innerhalb der E-Mail-Templates stehen für jedes einzelnen Template entsprechende Variablen zur Verfügung. Welche im Detail für das E-Mail-Template Komplett ausgeliefert zur Verfügung stehen, kannst Du durch einen Klick auf den Pfeil der Variablen Informationen herausfinden. Bitte beachte, dass die hier bereitgestellten Variablen nicht zwangsläufig auch in anderen E-Mail-Vorlagen einsetzbar sind.

Bereitstellen der Tracking-URL im Kundenkonto

Wechsle nun in die Versandkosten und hinterlege hier die in Schritt 1 herausgefundene **Tracking-URL (1)**. Hier darf nicht mit HTML oder JavaScript gearbeitet werden, da die URL im Template selbst zusammengebaut wird. Es darf also nur die reine URL mit der Variable für die Trackingnummer enthalten sein!

Hier ein Beispiel für die Tracking-URL von GLS:

```
https://glz-group.eu/DE/de/paketverfolgung?match={$offerPosition.trackingcode}
```

Da die Tracking-URL bereits in den Versandkosten bereitgestellt wurde, steht diese automatisch auch im Kundenkonto unter **Meine Bestellungen (1)** in den **Bestell-Details (2)** bereit. Dies gilt natürlich nur für Kunden mit einem Kundenkonto im Frontend.

Anpassung E-Mail-Vorlage bei mehreren Sendungsnummern

Bei großen Bestellungen kann es vorkommen, dass Du mehr als ein Paket versendest und so mehrere Sendungsnummern für eine Bestellung hast. In unserem Beispiel gehen wir davon aus, dass Du diese, mit einem Komma getrennt, in die Bestelldetails aufnimmst.

Bestellungs-Details: 20002

Übersicht Details Kommunikation Pläne/Best. Dokumente Status History

Bestellzeit: Tue Apr 18 2017 14:11:03 GMT+0200 (Mitteleuropäische Sommerzeit) Geräte-Typ: Desktop

Bestellnummer: 20002

Währung: EUR

Gesamtbetrag: 53,90

Gewählte Versandart: Standard Versand

Partner-ID:

E-Mail: test@shopware.com

Bestellung bearbeiten

Bezahl am:

Tracking Code: GLS12345, GLS23456, GLS34567, GLS45678

Versandkosten (EUR): 3,9

Versandkosten netto (EUR): 3,9

Bestellstatus: Offen

Zahlungsstatus: Komplett bezahlt

Freitextfelder

Freitextfelder

Für dieses Module wurden noch keine Freitextfelder konfiguriert.

In diesem Fall würde oben genanntes Beispiel nicht funktionieren, da Du pro Link immer nur eine Sendungsnummer übergeben kannst. Es ist also nötig mehrere Verfolgungslinks für einen solchen Versand zu generieren. Dies kannst Du beispielsweise über eine Erweiterung der E-Mail-Vorlage mit folgendem Inhalt erreichen:

```
{assign var='Paket' value=1}
{assign var='Tracking' value=', '|explode:{$sOrder.trackingcode}}
{foreach $Tracking as $temp}
Link zu Paket {$Paket}: {$temp}
{assign var='Paket' value={$Paket+1}}
{/foreach}
```

Der Inhalt der E-Mail könnte dann so aussehen:

E-Mail an den Kunden senden

An: test@shopware.com

Betreff: Statusänderung zur Bestellung bei Demoshop

Hallo Herr shopware Test,

der Bestellstatus für Ihre Bestellung 20002 hat sich geändert!
Die Bestellung hat jetzt den Status: Komplett ausgeliefert.

Den aktuellen Status Ihrer Bestellung können Sie auch jederzeit auf unserer Webseite im Bereich "Mein Konto" - "Meine Bestellungen" abrufen. Sollten Sie allerdings den Kauf ohne Registrierung, also ohne Anlage eines Kundenkontos, gewählt haben, steht Ihnen diese Möglichkeit nicht zur Verfügung.

Den aktuellen Lieferstatus können Sie durch Klick auf den jeweiligen Link einsehen:
Link zu Paket 1: <https://glb-group.eu/DE/de/paketverfolgung?match=GLS12345>
Link zu Paket 2: <https://glb-group.eu/DE/de/paketverfolgung?match=GLS23456>
Link zu Paket 3: <https://glb-group.eu/DE/de/paketverfolgung?match=GLS34567>
Link zu Paket 4: <https://glb-group.eu/DE/de/paketverfolgung?match=GLS45678>

Mit freundlichen Grüßen

Ihr Team von Demoshop

Natürlich kannst Du dieses Beispiel auch in der HTML-Vorlage verwenden und z.B. durch Listenelemente zur besseren Darstellung ergänzen.

Anpassung E-Mail-Vorlage: mehrere Logistik-Unternehmen

Wenn Du den Versand über verschiedene Logistik-Unternehmen abwickelst und immer den passenden Sendungs-Link senden möchtest, ist eine Anpassung der E-Mail-Vorlage erforderlich. Im Folgenden wird dies am Beispiel von DHL und GLS dargestellt:

```
{if $$Dispatch.name == "DHL"}  
Sendungsverfolgung: https://nolp.dhl.de/nextt-online-public/de/search?piececode={$sOrder.trackingcode}  
{else if $$Dispatch.name == "GLS"}  
Sendungsverfolgung: https://gls-group.eu/DE/de/paketverfolgung?match={$sOrder.trackingcode}  
{else}  
Es ist keine Sendungsverfolgung möglich  
{/if}
```

Wenn die Versandart "DHL" verwendet wird, wird der entsprechende DHL-Link verwendet, bei "GLS" der GLS-Link, bei einer anderen Versandart wird der Text "Es ist keine Sendungsverfolgung möglich" angezeigt (z.B. denkbar für Postversand)

Vorgehensweise für den Go-Live

In den meisten Fällen ist es so, dass Dein Shopware Shop nicht sofort unter der gewünschten Zieldomain live gehen kann. Beispielsweise wenn unter der Zieldomain noch ein alter Shop oder eine Webseite betrieben wird. In diesem Fall kannst Du Shopware in ein anderes Verzeichnis auf dem Server installieren, welcher beispielsweise temporär über eine Subdomain verfügbar ist.

Hier zeigen wir Dir die wichtigsten Punkte, die es zu beachten gilt, wenn Du Deinen Shop nachträglich live gehen lässt.

Manuell gepflegte Links anpassen

Unter Umständen hast Du in Deiner Testinstallation manuell gepflegte Links die zu einer Ressource in Deinem Shop leiten. Wenn Du die Domain des Shops nun änderst, solltest Du auch die Links anpassen, die zu diesem Shop führen.

An folgenden Orten werden häufig Links verwendet, die möglicherweise angepasst werden sollten.

Kategorien:

Kategorien können zum einen direkt verlinkt werden. Klicke unter Artikel->Kategorien auf die jeweilige Kategorie und pass den Link im Feld auf **externe Seite verlinken** an.

Auch im Kategorietext kann es sein, dass Du einen Link hinterlegt hast. Den Kategorietext findest Du im Bereich **CMS Funktionen**.

Artikel:

Auch in der Artikelbeschreibung können eigene Links hinterlegt werden. Klicke hierzu im Backend auf

Artikel > Übersicht, wähle Deinen Artikel und passe eventuelle Links in der Beschreibung an.

Einkaufswelt:

Eventuell hast Du auch eigene Einkaufswelten angelegt, in denen eigene Links hinterlegt wurden. Dann solltest Du auch diese anpassen. Die Einkaufswelten findest Du im Backend unter **Marketing > Einkaufswelten**. Überprüfe am besten alle dort angelegten Einkaufswelten.

Demodaten entfernen

Wenn Du zu Testzwecken die Demodaten Erweiterung installiert hast, solltest Du diese Demodaten vor dem Go-Live natürlich wieder entfernen. Das Deaktivieren oder Deinstallieren der Demodaten Erweiterung entfernt die Demodaten aber nicht wieder. Die Demodaten müssen also manuell wieder entfernt werden.

Folgende Daten werden durch die Demodaten Erweiterung hinzugefügt und sollten wieder entfernt werden:

- **Kategorien:** Lebensmittel, Bekleidung, Freizeit & Elektro inklusive der Unterkategorien.
- **Hersteller:** Shopware Fashion, Shopware Food, Shopware Freetime
- **Artikel:** Alle Artikel der genannten Hersteller.
- **Einkaufswelt:** Startseite, sofern diese nicht weiter benötigt wird.
- **Eigenschaften:** Wenn nicht mehr benötigt können die Sets Kleidung, Essen, Elektronik und Fashion sowie die Eigenschaften Gruppen Artikeltypen, Größe, Material, Zielgruppe und Zutaten entfernt werden.

Versandarten Prüfen

Überprüfe, ob die aktuell angelegten Versandarten auch im Livesystem verfügbar sein sollen. Eventuell sind hier noch temporäre Versandarten zum Test angelegt worden, welche nun entfernt werden sollten.

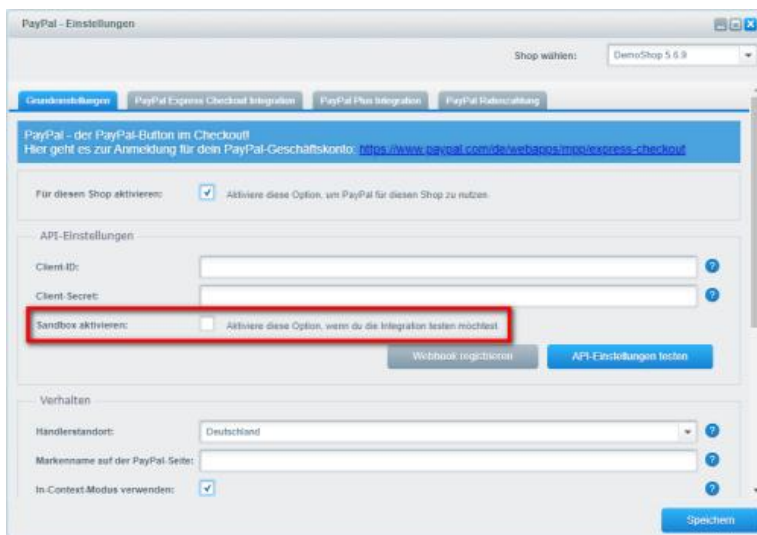
Wenn gewünscht kontrolliere zudem im Frontend, ob die Versandarten auch nur zu den Bedingungen zur Verfügung stehen, unter denen sie auch zur Verfügung stehen sollen.

Informationen zur Einrichtung der Versandkosten findest Du [hier](#).

Sandbox-Modus der Zahlungsarten deaktivieren

Wenn Du weitere Zahlungsarten hinzugefügt hast, wirst Du in der Testphase wahrscheinlich den Sandbox-Modus dieser Zahlungsarten aktiviert haben, um bei Testbestellungen keine echten Zahlungen zu generieren.

Im Falle der aktuellsten PayPal Version, findest Du diese Option beispielsweise im Backend unter **Kunde > Zahlungen > PayPal > Einstellungen > Grundeinstellungen**. Dort kann die Sandbox Option im Bereich **API-Einstellungen** deaktiviert werden.



Wie der Sandbox-Modus für die jeweiligen Zahlungsanbieter zu deaktivieren ist, kann sich von Zahlungsart zu Zahlungsart unterscheiden. Wie Du den Sandbox-Modus jeweils deaktivierst, erfährst Du in der Dokumentation des jeweiligen Anbieters.

Bestellprozess prüfen

Vor dem Go-Live solltest Du mindestens eine Testbestellung komplett durchführen. Nur so kannst Du erkennen, an welcher Stelle eventuell noch etwas nicht rund läuft. Besser ist es noch mehrere verschiedene Bestellungen durchzuführen, sodass möglichst viele verschiedene Fälle überprüft werden können.

Werden alle gewünschten Zahlungsarten angezeigt? Stehen die gewünschten Versandmethoden zur Verfügung? Funktioniert der Checkout Prozess ohne Probleme? Wird die Bestellbestätigung ordnungsgemäß versendet?

E-Mail Vorlagen und Mailer prüfen

Vor dem Go-Live ist es wichtig, Deine E-Mail Vorlagen sowie den Mailer zu prüfen. Führe dazu eine komplette Bestellung durch und überprüfe, ob die Bestellbestätigung ohne Probleme bei Dir ankommt. Wenn keine Mail versendet wird, überprüfe Deine [Mailer Einstellungen](#).

Wenn Deine Mail ankommt, überprüfe, ob alle Informationen in der Mail vorhanden sind.

- Wird das Logo korrekt angezeigt?
- Werden die Artikel korrekt angezeigt?
- Werden alle Variablen wie Name, Anrede und Adresse des Kunden richtig gefüllt?
- Hast Du die Bankverbindung angegeben?

Sollte noch etwas nicht passen, findest Du [hier](#) alle Informationen, um Anpassungen in der E-Mail Vorlage vorzunehmen.

Nummernkreise ggf. zurücksetzen

Während Deiner Tests wirst Du vermutlich Bestellungen ausgeführt, Kunden angelegt und eventuell sogar Rechnungen erstellt haben. Die Nummernkreise kannst Du bei Bedarf vor dem Go-Live wieder auf einen Startwert zurücksetzen. Dies kannst Du im Backend unter **Einstellungen > Grundeinstellungen > Shopeinstellungen > Nummernkreise** tun.

Der jeweilige Nummernkreis sollte nach dem Go-Live nicht mehr verändert werden. Zum einen fordert die aktuelle Gesetzeslage fortlaufende Rechnungsnummern, zum anderen kann es bei Anbindung eines Warenwirtschafts- oder CRM-Systems zu Abstimmungsproblemen zwischen den Programmen kommen.

Wenn Du Warenwirtschaft- oder CRM-Systeme nutzt, solltest Du Dich am besten vorher beim Hersteller informieren, ob das Ändern der Nummernkreise jetzt noch unproblematisch ist.

Shopware Account Lizenzumstellungen

Beim Anlegen des Shopware Accounts ist es notwendig eine Lizenzdomain anzugeben. Wenn Du auf Deiner Hauptdomain aber noch einen alten Shop oder eine Webseite betreibst, hast Du unter Umständen temporär eine Subdomain angegeben. Für eine korrekte Lizenzierung ist es notwendig, vor dem Go-Live diese temporäre Domain durch die finale Lizenzdomain zu ersetzen, unter der der Shop in Zukunft zu erreichen ist.

In Deinem Shopware Account unter **account.shopware.com** klickst Du auf Shops und dann auf Deine entsprechende **Shopdomain**, die Du ändern möchtest.



Im Bereich Shopinformationen findest Du nun den Button **Shop-Domain umbenennen (1)**, um Deine Domain umzubenennen.

Beachte, dass Du die Shop-Domain nur einmal umbenennen kannst. Du solltest Dir also sicher sein, dass die neue Domain Deine finale Domain ist.

Anschließend ist es noch notwendig den **Verwendungstyp (2)** der Domain auf Produktivumgebung zu ändern.

Domain beim Hoster anpassen

Wenn Du alle Punkte kontrolliert hast und Dein Shop soweit funktioniert, ist der letzte Schritt die Domain des Hosters auf das Hauptverzeichnis Deines Shopware Shops zu routen, sodass Dein Shop nun über die finale Domain erreichbar ist.

Wie Du genau das Routing Deiner Domain anpassen kannst, erfährst Du bei Deinem Hoster. In der Regel kann dies über eine entsprechende Admin Oberfläche selber konfiguriert werden.