

Amazon-Payment für TYPO3

Die Amazon-Payment TYPO3-Erweiterung bietet eine Schnittstelle für Amazon-Pay. Es können direkt ohne weitere Shop-Erweiterung o.ä. Produkte angeboten werden und über einen Amazon-Pay Checkout gekauft werden. Dies können physische oder auch digitale Produkte sein.

Requirements

- jQuery
- Website must be delivered by HTTPS

Links

- Leitfaden für den Bezahl/Login Button: <https://payments.amazon.de/merchant/tools>
- Tutorial zum Einrichten eines Amazon-Accounts: <http://www.alkim.de/Amazon-Payments-Anleitung---Voraussetzungen.html>
- Sourcen und Dokumentation des Amazon-Payment SDK: <https://github.com/amzn/login-and-pay-with-amazon-sdk-php>

Installation und Einrichtung!

Hinweise zum Website-Template

Damit die Widgets fehlerfrei dargestellt werden können, benötigen Sie zum einen den folgenden Meta-Tag:

```
<meta name="viewport" content="width=device-width,initial-scale =1.0, maximum-scale =1.0"/>
```

Des Weiteren sollten Sie das mitgelieferte Default-CSS einbinden, damit die Amazon-Widgets optimal angezeigt werden können. Mehr Informationen dazu finden Sie unter folgendem Link:

<https://payments.amazon.de/developer/documentation/lpwa/201952070>

Das mitgelieferte CSS wird einfach via TypoScript eingebunden:

```
plugin.tx_amazonpayment {
    _CSS_DEFAULT_STYLE (
        .tx-amazonpayment {
            position: relative;
        }
        /*
        ...weiteres CSS...
        */
    )
}
```

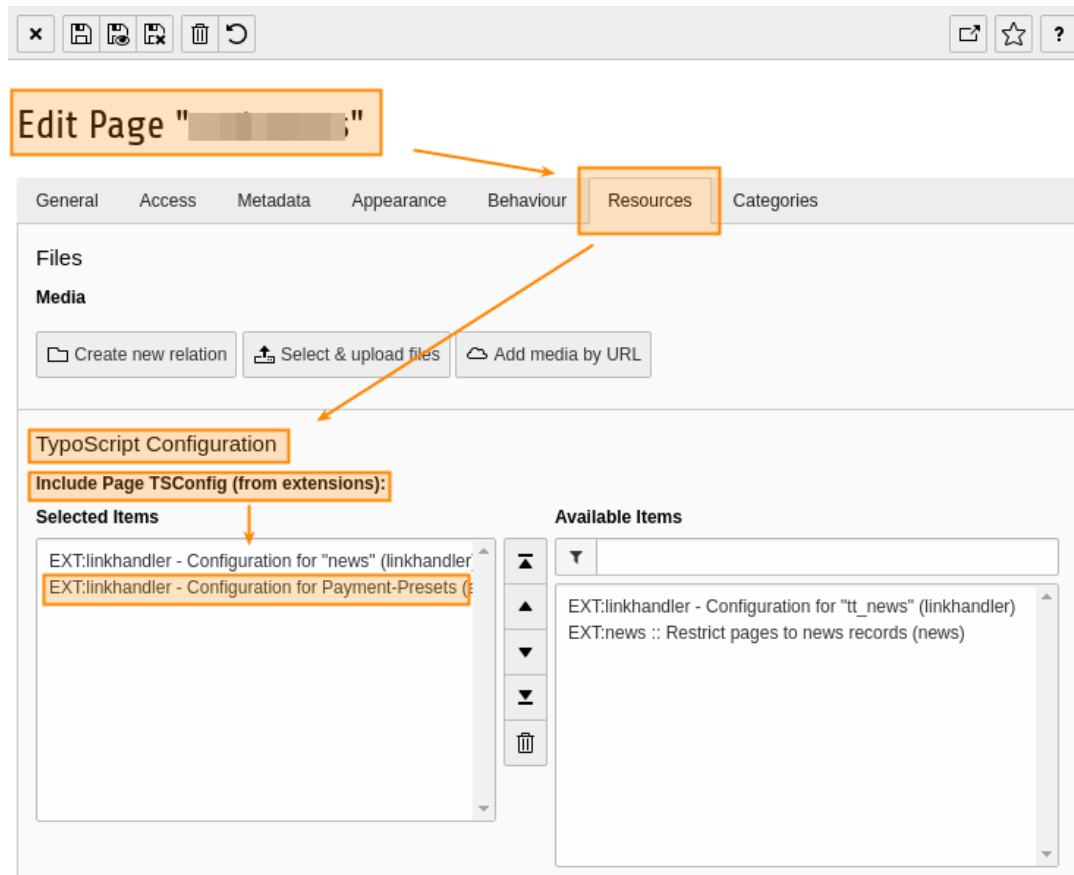
LinkHandler für Verlinkungen aus dem RTE (*optional*)

Wenn Sie Payment-Presets aus dem RTE heraus verlinken möchten, brauchen Sie eine zusätzliche Erweiterung namens `linkhandler`. Diese Erweiterung können Sie hier herunterladen:

- Für TYPO3 7.6.x: <https://github.com/cobwebch/linkhandler>
- Für TYPO3 6.2.x: <https://github.com/Intera/typo3-extension-linkhandler>

In TYPO3 6.2 muss lediglich die *Linkhandler*-Erweiterung installiert werden, danach ist diese Funktion im kompletten System verfügbar. Wenn Sie TYPO3 7.6 oder höher verwenden, müssen Sie im relevanten Seitenbaum die Konfiguration bereitstellen.

Dazu öffnen Sie das Page-Modul, wählen die Wurzelseite Ihrer TYPO3-Website an und bearbeiten die Seiten-Eigenschaften. Hier wechseln Sie dann auf den Tab *Resources* und wählen die zusätzliche TypeScript-Konfiguration an.



Wenn dies geschehen ist, haben Sie im Rich-Text-Editor (RTE) die Möglichkeit Links auf Payment-Presets zu legen.

Konfiguration

Die komplette Einrichtung geschieht via TypoScript-Konstanten. Die wichtigsten Daten, wie z.B. *clientId*, *sellerId*, *accessKey*, *secretKey* können Sie aus Ihrem Account auf <https://sellercentral-europe.amazon.com> auslesen.

TypoScript-Konstante	Description
themes.configuration.container.payment.amazon	Container for all payment records
themes.configuration.payment.amazon.sandbox	Run in sandbox mode
themes.configuration.payment.amazon.url	Amazon payments URL
themes.configuration.payment.amazon.clientId	Client ID
themes.configuration.payment.amazon.sellerId	Seller/Merchant ID
themes.configuration.payment.amazon.accessKey	Access Key ID
themes.configuration.payment.amazon.secretKey	Secret Access Key. Be sure that this key isn't public visible!
themes.configuration.payment.amazon.storeName	Name of the store
themes.configuration.payment.amazon.orderIdPrefix	Prefix for the order id

Hinweis:

Bitte vergessen Sie nicht Ihre Domain in der [Amazon Seller-Central](#) als zulässige Domain anzugeben. Wie das geht, lesen Sie im FAQ-Bereich.

Platzieren des Checkouts

Eine Webseite mit Amazon-Payment hat eine zentrale Checkout-Seite. Legen Sie dafür eine im Menü nicht sichtbare Seite an und erstellen Sie darauf ein Plugin `Amazon - Checkout`. Dieses Plugin benötigt keine weiteren Einstellungen. Wenn Sie

die TypeScript-Konstante `themes.configuration.container.payment.amazon` gesetzt haben, müssen Sie nicht einmal mehr einen Datenspeicher angeben.

Hinweis:

Die Checkout-Seite funktioniert nur, wenn Sie beim Aufruf ein Payment-Preset übergeben.

Verwendung

Allgemeines

Für jede Zahlung wird ein sogenanntes *Payment-Preset* benötigt. Dies kann ein temporär erstelltes (bspw. aus einem Warenkorb) oder auch ein wiederverwendbares (welches mit Hilfe des *Amazon-Payment - Presets* Plugins oder mit Hilfe des Linkhandlers platziert wird) sein.

Dieses *Payment-Preset* wird immer verwendet um einen *Bezahlen mit Amazon* Button bzw. den Checkout zu initiieren. Wenn der Kunde dann eine Zahlung starten möchte und sich einloggt, wird das eigentliche *Payment* erzeugt, welches die Zahlung dokumentiert.

Bei einer Zahlung müssen wir zwischen einem *physischen* und einem *digitalen* Produkt unterscheiden.

- **Ein physisches Produkt** ist ein materielles Produkt, welches real auf dem Postweg zugestellt werden muss. Hier ist somit eine Versandadresse notwendig.
- **Ein digitales Produkt** ist ein immaterielles Produkt (bspw. eine MP3- oder PDF-Datei) die durch einen Download bereitgestellt wird und dadurch vom Rückgaberecht ausgeschlossen ist. Hier wird das Adresswidget für die Versandadresse nicht benötigt.

Entscheidend ist hierbei also die Lieferadresse! Die Lieferadresse ist Teil der Risikobewertung von Amazon-Payment - wenn diese fehlt, greift Amazons Absicherung gegen *Chargebacks* nicht mehr.

Hinweis:

Bei *Chargebacks* mit Lieferadresse reicht ein Nachweis, dass die Bestellung an die im Widget gewählte Adresse stattgefunden hat - dann ist der Händler abgesichert. Beim Fehlen der Adresse ist der Nachweis der Vollfüllung schwieriger ([weitere Informationen zu Amazon Payments User Agreement](#)).

Payment-Presets

Mit den Payment-Preset kann man einfache Bezahl-Produkte erstellen. Diese *Payment-Preset Produkte* haben die folgenden Felder:

- **Seller note:** Dies ist ein Produktname oder Bezeichner für die Zahlung
- **Price:**
 - **Price amount:** Hier geben Sie den Preis des Produktes an. **Achtung:** Dieser verwendet als Euro/Cent-Trenner einen "." ! Somit wäre bspw. 125,99€ mit 125.99 einzutragen.
 - **Currency:** Dieses Feld gibt die Währung an, in welcher Ihr Preis angegeben wurde. Die verfügbaren Werte werden von der Amazon-Pay Schnittstelle vorgegeben.
- **Address required:** Diese Checkbox gibt an, ob es sich um ein *physisches* oder *immaterielles* Produkt handelt. Je nach Auswahl wird beim Checkout ein Adresswidget für die Versandadresse angezeigt oder nicht.
- **Accessible:**
 - **Deactivate:** Ist dieser Haken gesetzt, ist das Produkt deaktiviert und kann somit nicht gekauft werden.
 - **Activate starttime:** Dieses Feld gibt eine Startzeit an, ab welcher das Produkt gekauft werden kann. Möchten Sie bspw. ein besonderes Angebot schalten, welches erst ab einem bestimmten Zeitpunkt verfügbar ist, so verwenden Sie dieses Feld für die Startzeit. **Achtung:** Sie müssen hierfür nicht die *Deaktivieren*-Checkbox setzen, um Ihr Produkt zu deaktivieren. Wenn diese Checkbox gesetzt ist, werden auch die *Start- & Endzeit* ignoriert!
 - **Activate endtime:** Gibt eine Endzeit an, ab welcher Ihr Produkt nicht mehr gekauft werden kann.

Edit Payment-Preset "Rotes Gummiboot, 125.99" on page "Payment"

Seller note

Rotes Gummiboot

Price

Price amount (For example: 125.99)

125.99

Currency

EUR

Address required (Digital products do not need an address)

☐

Accessible

Deactivate

☐

Activate starttime

Activate endtime

 Payment-Preset [1]

Payment-Preset mit dem *Amazon-Payment - Preset Plugin*

Die Amazon-Pay Erweiterung bringt ein kleines Plugin mit sich, mit der Sie ein oder mehrere Produkte auf einer TYPO3-Seite platzieren können. Dazu rufen Sie einfach die entsprechende Seite im TYPO3-Backend auf und platzieren darauf ein *Amazon-Payment - Presets* Plugin. In diesem Plugin haben Sie dann die folgenden Einstellungen:

- **Layout:** Hier können Sie das Layout auswählen, in dem die einzelnen Presets angezeigt werden. Je nach Auswahl wird im `Partials/Amazon/PaymentPreset.html` Fluid-Partial eine andere Section verwendet. Wie Sie hier weitere Optionen hinzufügen können, lesen Sie im FAQ-Bereich.
- **Payment presets:** Hier können Sie auswählen, welche Produkte angezeigt werden. Wenn Sie keine Presets auswählen, so werden alle verfügbaren Presets aus dem zugewiesenen Daten-Container verwendet.

Edit Page Content "Alle aus Storage" on page "Payment-Products"

General

Plugin


Appearance

Access

Categories

Grid Elements

Selected Plugin







 Amazon-Payment - Presets


Plugin Options

Layout

List-Template

Payment presets (empty selection will display the whole presets from selected record storage)



 Payment-Preset

Nun können Sie mit diesem Plugin Produkte als Teaser auf Ihrer Startseite anzeigen, in Ihrer Sidebar oder auf speziell zugeschnittenen Produktseiten.

Payment-Preset mit dem Linkhandler

Dank dem Linkhandler haben Sie die Möglichkeit in jedem RTE ein Payment-Preset Link hinzuzufügen.

The image shows two parts of a software interface. The top part is a 'Modify link' dialog box with tabs for 'Page', 'File', 'Folder', 'External URL', 'Email', 'Payment-Preset', and 'News'. The 'Payment-Preset' tab is selected. It contains fields for 'Target', 'Title' (set to 'Opens internal link in current window'), and 'Style'. Below these is a 'Page tree' and a 'Select Records:' section. The 'Select Records:' section has a search bar and a list of records under the heading 'Payment-Preset (3)'. The records are: 'a. + Rotes Gummiboot, 125.99', 'a. + Drei köpfiger Affe, 2399.99', and 'a. + Halbes Pferd mit Sahne, 75.99'. The third record is highlighted with an orange box. An orange arrow points from this record to the 'Payment-Preset' tab in the dialog. Another orange arrow points from the 'Payment-Preset' tab to the 'Text' editor below. The bottom part is a 'Text' editor with a toolbar containing various icons. The third icon from the right, which represents a link, is highlighted with an orange box. An orange arrow points from this icon to the third line of text in the editor: 'Klicken Sie hier um ein halbes Pferd mit Sahne zu kaufen!'. The first two lines of text are also highlighted with orange boxes: 'Klicken Sie hier um ein knall rotes Gummiboot zu kaufen!' and 'Klicken Sie hier um einen drei köpfigen Affen zu kaufen!'.

Ablauf beim Kaufen

- Der Kunde klickt den *Bezahlen mit Amazon* Button, welcher das *Amazon-Login* Popup öffnet

Anmeldung bei Amazon.de - Chromium

https://eu.account.amazon.com/ap/signin?_encoding=UTF8&openid.mode=checkid_setup&openid.ns=http:

amazon

Melden Sie sich bei  mit Ihrem Amazon-Konto an

Wie lautet Ihre E-Mail-Adresse?

Wie lautet Ihr Passwort?

Weiter (über den Sicherheitsserver)

Durch die Anmeldung bestätigen Sie unsere Nutzungsbedingungen, Datenschutzerklärung sowie Cookies und Internet-Werbung.

[Haben Sie Ihr Passwort vergessen?](#)

[Ein Amazon-Konto erstellen.](#)

Einfacher Login

Verwenden Sie Amazon, um sich bei dieser Website ohne ein weiteres Passwort anzumelden.



Sichere Anmeldung

Amazon gibt Ihr Passwort niemals an OpenImmoTest weiter



[Weitere Informationen](#)

[Unsere AGB](#) [Datenschutzerklärung](#) [Cookies & Internet-Werbung](#) Deutsch ▼

© 1998-2017, Amazon.com, Inc. oder Tochtergesellschaften

- Wenn sich der Kunde erfolgreich eingeloggt hat, schließt sich das Popup wieder. Im Hintergrund wird bei Amazon eine *OrderReferenceId* erstellt, welche für die eindeutige Identifizierung der Zahlung verwendet wird. Diese *OrderReferenceId* wird ebenfalls von der *amazon_payment*-Erweiterung genutzt, um die Zahlung zu verwalten. So wird im TYPO3-Backend ein *Payment*-Datensatz erstellt (`amazonPayment.initializePayment()` sendet AJAX-Request mit *orderReferenceId* an `API::initialize`), welcher alle Informationen in Verbindung zur *OrderReferenceId* zu der Zahlung sammelt. Während dieses Vorgangs wird die *Amazon User Id*, *Amazon Username* und *Amazon Email* abgerufen und in

unserem Payment-Datensatz gespeichert.

Edit Payment "[REDACTED]" on page "Payment"

General

Log entries

Amazon User Id

[REDACTED] x

Amazon Username

[REDACTED] x

Amazon Email

[REDACTED] x

Amazon Order-Reference

ID

[REDACTED] x

State

Draft x

Price

Price amount (For example: 125.99)

125.99 x

Currency

EUR v

 Payment [28]

So finden Sie Payment-Datensatz immer alle relevanten Daten zur Zahlung.

- Wenn die Bezahlung erfolgreich initiiert wurde, werden die benötigten Widgets in die jeweiligen Wrapper injiziert:
 - `<div id="amazon-address-book"></div>` Wrapper for the address book widget
 - `<div id="amazon-wallet"></div>` Wrapper for the the wallet widget

Des Weiteren werden Buttons mit entsprechenden Events verbunden:

- `<div class="amazon-button" id="amazon-back-to-previous-page"></div>` Zurück zum Produkt-Button
- `<div class="amazon-button" id="amazon-logout"></div>` Logout-Button
- `<div class="amazon-button" id="amazon-confirm-payment"></div>` Bestätigungs-Button
- `<div class="amazon-button" id="amazon-modify-payment"></div>` Bearbeitungs-Button
- `<div class="amazon-button" id="amazon-finish-payment"></div>` Zahlungsabschluss-Button

Checkout in Detail

Der Checkout besteht aus mehreren Schritten, welche wie folgt definiert sind:

1. **Vor dem Login (initialize)** Das Plugin wird initialisiert. D.H. der Login-Button wird initialisiert und der Besucher hat die Möglichkeit sich über Amazon einzuloggen. Des Weiteren werden:
 - HTML-Elemente die die CSS-Klasse `amazon-show-on-login` haben, werden sichtbar gemacht, wenn sich der Besucher angemeldet hat.
 - HTML-Elemente die die CSS-Klasse `amazon-hide-on-login` haben, werden unsichtbar gemacht, wenn sich der Besucher abgemeldet hat.
2. **Nach dem Login (modify)** Jetzt wählt der Besucher die gewünscht Lieferadresse (sofern benötigt) und die entsprechende Zahlart aus.
 - HTML-Elemente die die CSS-Klasse `amazon-show-on-modify` haben, werden sichtbar gemacht, wenn dieser Schritt erreicht wird.

- HTML-Elemente die die CSS-Klasse `amazon-hide-on-modify` haben, werden unsichtbar gemacht, wenn dieser Schritt erreicht wird.

3. **Beim Bestätigen (confirm)** In diesem Schritt werden dem Besucher noch einmal alle Daten in der Übersicht angezeigt. Wenn er diese für Richtig erachtet, kann er den *Bezahlung abschließen* Button klicken womit dann die eigentliche Zahlung durchgeführt wird.

- HTML-Elemente die die CSS-Klasse `amazon-show-on-confirm` haben, werden sichtbar gemacht, wenn dieser Schritt erreicht wird.
- HTML-Elemente die die CSS-Klasse `amazon-hide-on-confirm` haben, werden unsichtbar gemacht, wenn dieser Schritt erreicht wird.

Hinweis:

Die Auflistung der Datenzusammenfassung kann über das Fluid-Template `Templates/Amazon/Confirm.html` angepasst werden. Das Resultat dieses Fluid-Templates wird via AJAX an das Frontend geschickt.

4. Nach erfolgreicher Zahlung (finish) Diesem Schritt wird ausgeführt, wenn die Zahlung erfolgreich war.

- HTML-Elemente die die CSS-Klasse `amazon-show-on-finish` haben, werden sichtbar gemacht, wenn dieser Schritt erreicht wird.
- HTML-Elemente die die CSS-Klasse `amazon-hide-on-finish` haben, werden unsichtbar gemacht, wenn dieser Schritt erreicht wird.

Hinweis:

Die Checkout-Ansicht kann über das Fluid-Template `Templates/Amazon/Checkout.html` angepasst werden.

Signals/Slots

Die Amazon-Pay Erweiterung bietet die folgenden Signals/Slots an, womit Sie die Funktion erweitern können:

- **afterAmazonPaymentSuccessful** - Dieses Signal wird gesendet, sobald ein Besucher erfolgreich ein Produkt gekauft hat. Es erhält als Parameter die *OrderReferenceld*, womit Sie dann in der Lage sind das *Payment*-Objekt zu identifizieren und zu laden. So könnten Sie bspw. im Falle eines digitalen Produktes das Signal fangen und eine E-Mail mit dem gekauften Produkt verschicken.

Hier ein kleiner Beispielcode für die Registrierung in der `ext_localconf.php` :

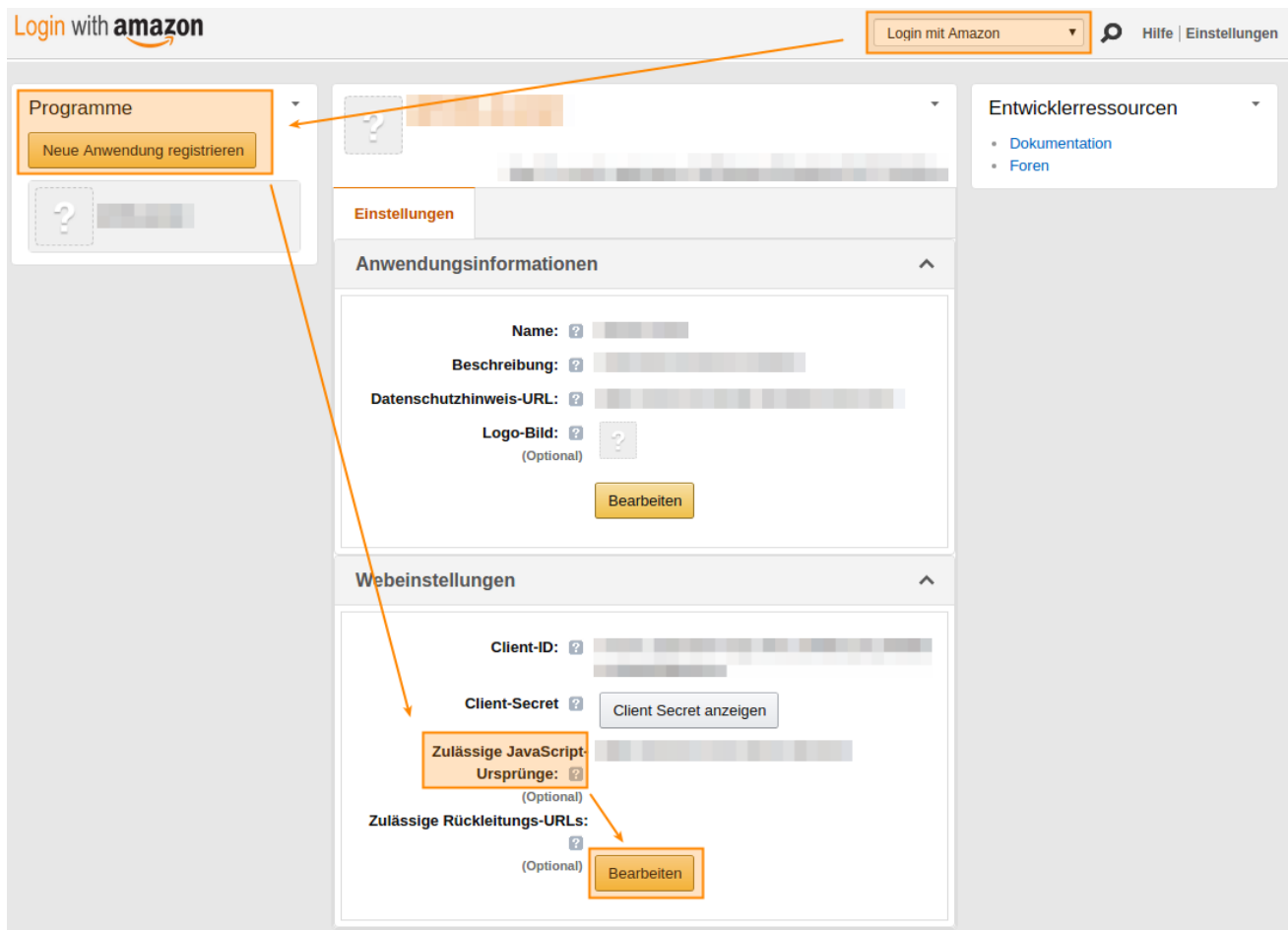
```
/** @var \TYPO3\CMS\Extbase\SignalSlot\Dispatcher $signalSlotDispatcher */
$signalSlotDispatcher = \TYPO3\CMS\Core\Utility\GeneralUtility::makeInstance(\TYPO3\CMS\Extbase\SignalSlot\Dispatcher->connect(
    \CodingMs\AmazonPayment\Controller\AmazonController::class, 'afterAmazonPaymentSuccessful' ,
    \CodingMs\AmazonPayment\Utility\Tools::class, 'logAfterAmazonPaymentSuccessfulSlot'
));
```

Dieses Beispiel erstellt einen Log-Eintrag nach erfolgreicher Bezahlung.

FAQ

Wie gebe ich die Amazon-Payment JavaScript Origin an?

Melden Sie sich in der [Amazon Seller-Central](#) an und wechseln Sie in der oberen Menüleiste auf den Punkt *Login mit Amazon*. Hier müssen Sie nun eine neue Anwendung hinzufügen oder bei einer bestehenden Anwendung die *Zulässige JavaScript Ursprünge* hinzufügen.



Wie kann ich weitere Layouts im *Amazon-Pay - Presets* Plugin verfügbar machen?

Um weitere Layouts für die Anzeige der Presets bereitzustellen, benötigen Sie als erstes ein wenig Seiten-TypoScript (tsconfig). Hiermit fügen Sie einen neuen Wert in der Auswahlbox im Plugin hinzu:

```
TCEFORM.tt_content.pi_flexform.amazonpayment_presets.sDEF {
    settings\templateSection {
        addItem {
            Teaser = Teaser-Layout
        }
    }
}
```

Im nächsten Schritt müssen Sie das `Partials/Amazon/PaymentPreset.html` um eine weitere Section erweitern. Dafür würden Sie bspw. den folgenden Fluid-Code hinzufügen:

```
<f:section name="Teaser">
    <div id="payment-preset-{paymentPreset.uid}" class="payment-preset">
        <h2>{paymentPreset.sellerNote} </h2>
        <span><f:format.number decimals="2" decimalSeparator="," thousandsSeparator=".">{paymentPreset.amo
        <f:link.action action="checkout" pageUid="{settings.amazon.checkout.pageUid}"
            arguments="{paymentPreset: paymentPreset.uid}" class="btn btn-default">
            <f:translate key="tx_amazonpayment_label.payment_preset_buy" />
        </f:link.action>
    </div>
</f:section>
```

Nun sollten Sie eine weitere Layout-Option zur Auswahl haben und zum Rendern wir die gerade gezeigte Fluid-Section verwendet.

Wie kann ich die Amazon-Pay Erweiterung debuggen?

Um Debugging-Ausgaben in der JavaScript-Console zu erhalten, aktivieren Sie einfach die folgende TypoScript-Konstante:

```
themes.configuration.payment.amazon.debug = 1
```

Achtung:

Deaktivieren Sie diese umgehend wieder, wenn Sie Ihr System Live schalten!

Wie kann ich die Sprachvariablen an meine Bedürfnisse anpassen?

Sprachvariablen und Meldungen können Sie einfach mit dem folgenden Setup-TypoScript anpassen:

```
plugin.tx_amazonpayment {
    _LOCAL_LANG {
        default {
            tx_amazonpayment_label.payment_preset_buy = Kaufen
        }
        en {
            tx_amazonpayment_label.payment_preset_buy = Buy
        }
        de {
            tx_amazonpayment_label.payment_preset_buy = Kaufen
        }
    }
}
```

Mit diesem Beispiel wird das Label des Kaufen-Buttons in der Preset-Ansicht angepasst. Welche Translation-Keys verfügbar sind, können Sie in der Datei `amazon_payment/Resources/Private/Language/locallang.xlf` einsehen.