

Installationsanleitung für Unser OpenCart

Diese Anleitung erklärt die Installation und Benutzung der Unser Extension für OpenCart.

Release Datum: Thu, 04 Mar 2021 14:16:25 +0100
Version: 1.0.54

wallee AG
General-Guisan-Strasse 47
CH-8400 Winterthur

© copyright by wallee AG Thu, 04 Mar 2021 14:16:25 +0100

Inhaltsverzeichnis

1	Einleitung	5
1.1	Ablauf der Installation	5
1.2	System Anforderungen	5
2	Konfiguration	7
2.1	Live Umgebung	7
2.2	Test Umgebung	7
3	Modul Installation im OpenCart Shop	8
3.1	Installation	8
3.2	Updates und Upgrades	8
3.2.1	Update Checklist	8
3.2.2	Update Instruktionen	8
4	Modul Konfiguration im OpenCart Shop	9
5	Besonderheit OpenCart 1.5.x / 2.0 / 3.0 Installation	10
5.1	Konfiguration des Hauptmoduls	10
5.2	OpenCart 2.0 / 3.0 - Refresh Modifications	10
5.3	Besonderheit OpenCart 1.0 Installation	10
5.4	Konfiguration der Zahlungsart	10
5.5	Shop ID	11
5.6	Bestell Präfix	11
5.7	Wahl der Autorisationsmethode	12
5.8	Direktes Verbuchen von Bestellungen	12
5.9	Unsicherer Status	12
5.9.1	Setzen von Order-Status	12
5.10	Optional: Validierung	12
5.11	Multishop Set UP	13
6	Einstellungen / Konfiguration von Zahlungsarten	14
6.1	Allgemeine Informationen zu den Zahlungsarten	14
6.2	Informationen zum Zahlungszustand	14
6.2.1	Bestellstatus "ausstehend" / bevorstehende Zahlung (oder ähnlich)	14
6.2.2	Abgebrochen / stornierte Bestellungen	15
6.3	Rechnung	15

6.4	Prepayment	15
6.5	Ratenzahlung	15
6.6	Gesicherte Rechnung	16
7	Das Modul in Aktion	17
7.1	Empfohlene Arbeitsweise	17
7.2	Nützliche Transaktionsinformationen zur Bestellung	17
7.3	Einpflegen von Zahlungsinformationen eines Verarbeiters	17
7.3.1	OpenCart Bestellbestätigung (E-Mail)	18
7.3.2	OpenCart Rechnung (PDF)	18
7.3.3	OpenCart-Backend (Transaktionsdetails)	18
7.3.4	OpenCart Success-Page	18
7.4	Verbuchen / Stornieren von Bestellungen	19
7.4.1	Verbuchen von Bestellungen	19
7.4.2	Bestellungen stornieren	20
7.5	Rückvergüten von Bestellungen	20
7.6	Einrichten Cron Job	21
7.7	Teilstornierung	21
8	Testing	22
8.1	Testdaten	22
9	Fehler und deren Lösungen	24
9.1	Die Referrer URL erscheint in meinem Analytics Tool	24
10	Kompatibilität zu Third-Party Plugins	25
10.1	Geburtstag und Geschlecht in OpenCart	25
11	Error Logging	26
11.1	Log Levels	26
11.2	Pfad der Log Datei	26
12	Erweiterte Informationen	27
12.1	Transaction Object	27

1 Einleitung

Diese Anleitung beschreibt die Installation, die Konfiguration und die Nutzung der Zahlungsmoduls für OpenCart und Unzer.

Bevor Sie mit der Installation beginnen können, stellen Sie sicher, dass Sie im Besitz aller notwendigen Daten sind:

- Bentzernamen und Passwort zum Login des Backends von Unzer
- OpenCart Zahlungsmodul als ZIP Datei.
- Zugangsdaten zu Ihrem Server und Shop

Teilstornierungen sind nur bedingt durch das Backend des Shops auslösbar. Hierfür können Sie während der Verbuchung die Option "Bestellung schliessen" anwählen, dann werden alle restlichen Positionen storniert. Reine Teilstornierungen müssen über das hIP von Unzer ausgelöst werden.

Der Kassenvorgang muss via https geladen werden damit die Zahlarten für Kunden verfügbar sind.

1.1 Ablauf der Installation

In diesem Dokument erhalten Sie alle Informationen, welche für die Installation des Moduls notwendig sind. Für die Inbetriebnahme des Moduls sollten Sie die einzelnen Punkte der Checkliste strikt befolgen. Nur so kann ein sicherer Einsatz in Übereinstimmung mit allen Sicherheitsregularien gewährleistet werden.

1. Konfiguration der Testoberfläche anhand der Integrationsdaten von Unzer.
2. Konfiguration der Grundeinstellungen des Zahlungsmoduls
3. Konfiguration der Zahlungsarten

.htaccess Verzeichnisschutz

Für den Test des Zahlungsmoduls muss jeglicher Verzeichnisschutz oder IP Sperre auf Ihrem Server zwingend ausgeschaltet sein, sonst kann es sein, dass das Zahlungsfeedback von Unzer nicht bis zu Ihrem Shop durchdringt.

1.2 System Anforderungen

Generell hat das Plugin die gleichen Anforderungen an das System wie OpenCart selber. Nachfolgend finden Sie die wichtigsten Anforderungen des Plugins:

- PHP Version: 5.4.x oder höher
- OpenSSL: Aktuelle Version mit Unterstützung von TLS 1.2 oder höher.
- fsockopen: Die PHP Funktion fsockopen muss zur Verfügung stehen. Das Plugin muss Verbindungen zu anderen Servern öffnen können.

- PHP Funktionen: Im weiteren müssen alle gängigen PHP Funktionen zur Verfügung stehen.

Verwenden Sie OpenCart in der Version 3.0.3.5 oder 3.0.3.6, müssen Sie ihren Store patchen, um das Twig-Extension-Problem zu beheben. Dies kann bspw. mit der Installation der folgenden Extension gelöst werden: https://www.opencart.com/index.php?route=marketplace/extension/info&extension_id=40469.

2 Konfiguration

2.1 Live Umgebung

Die Zugangsdaten für (<https://insights.unzer.com/>) finden Sie in Ihrem Unzer Account. Diese sind:

- API Private Key
- API Public Key

Loggen Sie sich in Ihren Unzer Account ein und navigieren Sie zu "Einstellungen > Konfiguration". Dort finden Sie die nötigen Parameter.

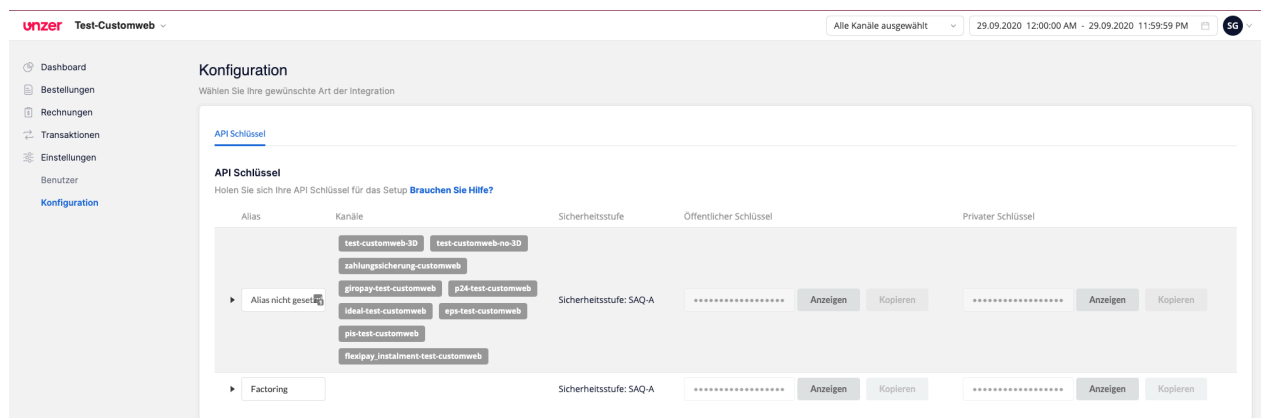


Abbildung 2.1: API Schlüssel

Tragen Sie nun diese beiden Schlüssel in der Hauptkonfiguration des Plughins in Ihrem Shop ein.

2.2 Test Umgebung

Um sich in die Testumgebung einzuloggen verwenden Sie folgende URL: <https://sbx-insights.unzer.com/>

3 Modul Installation im OpenCart Shop

3.1 Installation

Sie sollten zu diesem Zeitpunkt eigentlich bereits im Besitz des Moduls sein. Um das Modul in Ihrem OpenCart Shop zu installieren, führen Sie bitte folgende Schritte durch:

1. Entpacken Sie das Archive welches Sie heruntergeladen haben.
2. Im entpackten Ordner finden Sie einen Ordner "files"
3. Für einige Shops werden verschiedene Versionen des Plugins zur Verfügung gestellt.
4. Laden Sie nun mit Ihrem preferierten FTP Programm **den ganzen Inhalt** dieses Ordners ins Stammverzeichnis Ihres Shops. Bei gewissen Shops gibt es einen spezifischen Ordner welche die Plugins enthält. Sollte dies der Fall sein laden Sie das Plugin in diesen Ordner hoch. Stellen Sie sicher, dass die Ordner nur zusammengeführt werden und nicht ersetzt.
5. Falls nicht bereits geschehen, loggen Sie sich bitte in Ihrem Shop ein.

3.2 Updates und Upgrades

3.2.1 Update Checklist

Bitte führen Sie vor jedem Update bitte folgende Schritte durch:

1. Erstellen Sie immer ein Backup der Files und der Datenbank, bevor Sie mit dem Update beginnen
2. Testen Sie das Update auf einem **Testsystem**. Wir testen die Module intern, es kann aber sein, dass im Zusammenspiel mit anderen Plugins weitere Fehler auftreten.
3. Warten Sie bis die Files vollständig auf den Server gespielt sind und besuchen Sie dann die Konfigurationsseite des Hauptmoduls. Falls vorhanden leeren Sie bitte auch noch alle Caches des Shopssystems

Je nach Update kann es sein, dass eine Datenbankmigration durchgeführt werden muss. Wir empfehlen Ihnen daher, dass Update zu einem Zeitpunkt durchzuführen, indem nicht zu viele Kunden auf Ihrem Shop surfen.

Vorsicht bei Updates im Live Shop

Besondere Vorsicht ist bei Updates im Live Shop geboten. Testen Sie ein Update immer zuerst in einem Testsystem. Unser Support Team hilft Ihnen gerne bei Problemen.

3.2.2 Update Instruktionen

Bevor Sie das Update aufspielen, lesen Sie bitte die Update Instruktionen. Diese finden am unteren Ende der Pluginseite unter Changelog. Sofern nichts weiteres in den Update Instruktionen steht, können die Files einfach überschrieben werden.

4 Modul Konfiguration im OpenCart Shop

Die Konfiguration teilt sich in zwei Stufen. Der erste Schritt umfasst die Konfiguration des Hauptmoduls mit den Grundeinstellungen (vgl. [Konfiguration des Hauptmoduls](#)). In einem zweiten Schritt können für jede [Zahlungsart](#) individuelle Konfigurationen vorgenommen werden. Dies bringt Ihnen volle Flexibilität und grösste Anpassung an Ihre Prozesse.

Erstellen Sie Backups!

Bitte erstellen Sie auf jeden Fall ein Backup des Hauptverzeichnisses Ihres Shops. Bei Problemen können Sie so jederzeit wieder zum Ursprungszustand zurückkehren.

Im Weiteren empfehlen wir Ihnen die Integration zuerst auf einem Testsystem durchzuführen. Es kann immer wieder zu Komplikationen mit von Ihnen installierten Drittmodulen kommen. Bei Fragen steht Ihnen unser kompetenter Support gerne zur Verfügung.

5 Besonderheit OpenCart 1.5.x / 2.0 / 3.0 Installation

Das Modul ist mit OpenCart 2.x und 1.5.x kompatibel. Achten Sie darauf, dass Sie die Dateien des Ordners files_2.x verwenden, sollten Sie mit OpenCart 2 / MijoShop / AceShop arbeiten. Verwenden Sie den Ordner files_1.x, wenn Sie OpenCart 1.5 als Basis nutzen.

Um einen reibungslosen Einsatz zu garantieren, müssen Sie einige Besonderheiten bei den entsprechenden Versionen beachten.

5.1 Konfiguration des Hauptmoduls

Sie finden die Einstellungen des Hauptmoduls unter "**Extension > Modules > Unser Base Module**". Installieren Sie das Modul indem Sie auf **Install** klicken.

Mit einem Klick auf **Edit** gelangen Sie in die Konfiguration des Hauptmoduls. Tragen Sie hier die Daten in die einzelnen Felder ein. Die benötigten Daten haben Sie entweder bereits im Backend von Unser eingetragen oder von Unser in vorheriger Korrespondenz erhalten. Jede Option wird zudem mit einem kurzen Info Text im Shop nochamls erläutert.

5.2 OpenCart 2.0 / 3.0 - Refresh Modifications

Sollten Sie OpenCart 2.0 verwenden ist es wichtig, dass Sie die Modifications refreshen. Dazu gehen nach **Module > Modifications**. Oben Rechts finden Sie den Button refresh. Mit dem Klick darauf können Sie die Modifikationen neuladen. Stellen Sie sicher, dass der Ordner system > modifications Schreibrechte besitzt.

5.3 Besonderheit OpenCart 1.0 Installation

Sollten Sie OpenCart 1.0 verwenden, dann benötigen Sie die Extension VQMOD. Weitere Informationen zur Installation und zum Projekt finden Sie unter <https://github.com/vqmod/vqmod/wiki/Installing-vQmod-on-OpenCart>

5.4 Konfiguration der Zahlungsart

Nachdem das Hauptmodul erfolgreich konfiguriert wurde, finden Sie unter **Extensions > Payments** die Einstellungen zu den einzelnen Zahlungsarten in Ihrem Shop. Jede Zahlart wird Ihnen einzeln aufgelistet. Installieren Sie jene Zahlungsmethoden, welche Sie Ihren Kunden anbieten möchten. Sie können für jede Zahlungsart individuelle Einstellungen vornehmen und so die Zahlung optimal an Ihre Prozesse anpassen. Die zentralen Optionen werden in der Folge genauer erläutert.

Mit dem Klick auf **Install** wird eine Zahlungsmethode in Ihrem Shop installiert. Klicken Sie auf **Edit** um die Konfiguration der Zahlungsart anzupassen.

5.5 Shop ID

Im Falle eines Multishop Set-Ups (vgl. [Kapitel Multishop](#)) identifiziert der Parameter Shop ID die Anfrage des Shops und leitet, somit abhängig von der Shop ID, das Feedback von Unser entsprechend weiter.

5.6 Bestell Präfix

Mit der Option Bestell Präfix können Sie ihr eigenes Bestellschema für die Übermittlung an Unser definieren. Diese Option hilft Ihnen vor allem in einem Multishop Set up, um auch im Backend von Unser direkt zu erkennen, von welchem Shop die Transaktion stammt. Der Term "{id}" wird durch das Modul automatisch mit der fortlaufenden Bestellnummer ersetzt.

5.7 Wahl der Autorisationsmethode

Sie haben die Wahl zwischen verschiedenen Darstellungsweisen, wie Sie Kartenzahlungen autorisieren möchten. Über die Option Autorisierungsmethode legen Sie fest, wie Sie die einzelne Zahlungsart verarbeiten möchten.

Zahlungsverarbeitung

Bitte beachten Sie, dass zur Nutzung einiger Methoden der Zahlungsverarbeitung zusätzliche Optionen bei Unzer freigeschaltet werden müssen.

Es kann sein, dass gewisse Autorisierungsmethode nicht für jede Zahlungsart verfügbar ist.

Mit der Einführung von PCI 3 ergeben sich insbesondere für die Autorisierungsart Hidden und Ajax zusätzliche Anforderungen bei der Verarbeitung von Kreditkarten. für weitere Informationen hierzu kontaktieren Sie am besten direkt Unzer.

5.8 Direktes Verbuchen von Bestellungen

Mit der Option "Buchung" legen Sie fest, ob Zahlungen direkt verbucht, oder zuerst autorisiert werden sollen. Sofern Sie in einem ersten Schritt nur autorisieren möchten, stellen Sie sicher, dass die Zahlungen später verbucht werden.

Eine Reservation ist, abhängig von Ihrem Acquiring Vertrag, nur eine gewisse Zeit garantiert. Sollten Sie in dieser Zeit nicht verbuchen, kann es sein, dass die Autorisierung nicht mehr garantiert wird. Weitere Informationen zum Prozess der Verbuchung finden Sie weiter unten.

Unterschiedliche Einstellungen zwischen Unzer - Backend und Modul

Es kann sein, dass die Einstellungen im Modul für die Zahlungsweise Ihre Einstellungen im Backend von Unzer einfach überschreiben.

5.9 Unsicherer Status

Sie können Bestellungen, für welche die Zahlung noch nicht garantiert werden kann, speziell markieren. Dies erlaubt Ihnen diese Bestellung vor dem Versand noch einmal manuell zu prüfen.

5.9.1 Setzen von Order-Status

Sie können für jede Zahlungsart festlegen in welchen Status die Bestellung je nach Buchungstatus verschoben werden soll. Es handelt sich hier um den initialen Status der Bestellung.

5.10 Optional: Validierung

Mit der Validierung steuern Sie den Zeitpunkt wann eine Zahlungsart im Checkout angezeigt werden soll. Diese Einstellung ist für Module relevant bei denen die Benutzung von Voraussetzungen Ihres Kunden abhängig ist. Beispielsweise wenn noch eine Bonitätsprüfung erfolgt soll, oder die Zahlungsart nur in gewissen Ländern verfügbar ist. Es kann sein, dass diese Einstellung bei Ihnen nicht ersichtlich ist. Dann überspringen Sie diesen Punkt.

Sollte die Einstellung ersichtlich sein, haben Sie die Wahl zwischen folgenden Optionen:

- **Validierung vor Auswahl der Zahlungsmethode:** Es erfolgt vor der Auswahl der Methode ein Validierungsprüfung. Sofern die Voraussetzungen nicht erfüllt sind, wird die Zahlungsart nicht angezeigt.
- **Validierung nach der Auswahl der Zahlungsmethode:** Die Prüfung der Voraussetzung erfolgt nach der Wahl und vor der Bestätigung der Bestellung.
- **Während der Autorisierung:** Die Validierungsprüfung erfolgt durch Unzer während dem Autorisationsprozess. Die Zahlmethode wird auf jeden Fall angezeigt.

5.11 Multishop Set UP

Das Zahlungsmodul ist für die Nutzung eines Vertrags mit Unzer mit mehreren Shops ausgelegt. Es gilt dabei den Fall der integrierten Multishop Funktionalität vom Fall der unabhängigen Shops zu unterscheiden:

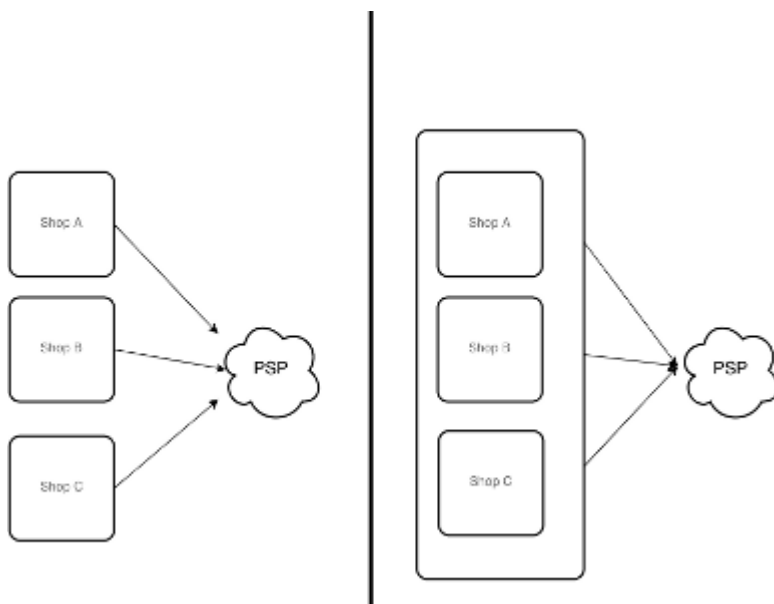


Abbildung 5.1: Multishop set UP

Lizenzierung

Für den Fall der unabhängigen Shops benötigen Sie für jeden Shop eine Lizenz für das Zahlungsmodul. Für Staffelpreise treten Sie bitte mit uns in Verbindung.

6 Einstellungen / Konfiguration von Zahlungsarten

6.1 Allgemeine Informationen zu den Zahlungsarten

Das Plugin wird mit einer grossen Auswahl von Zahlungsarten ausgeliefert. Sollte eine Zahlungsart nicht verfügbar sein, bitten wir Sie sich bitte mit uns in Verbindung zu setzen.

Für die Nutzung eines Zahlungsmittels muss die Zahlungsart in Ihrem Konto bei Unzer und im Shop aktiviert sein. Informationen zur Konfiguration des Zahlungsmoduls finden Sie weiter oben.

In den folgenden Punkten finden Sie wichtige Informationen zu bestimmten Zahlungsarten, die sich vom Standard unterscheiden.

6.2 Informationen zum Zahlungszustand

Sie können für jede Zahlungsart die initialen Zahlungsstatus individuell definieren (Status für autorisierte Zahlungen etc). Dabei definieren Sie den Zahlungsstatus für die einzelnen Zustände abhängig von der Verarbeitungsart der Bestellung (verbucht, autorisiert, etc.). Es handelt sich dabei um den initialen Status, welchen die Bestellung annimmt. Abhängig von der durch Sie durchgeführten Mutationen kann sich der Status ändern.

Wichtige Info bezüglich dem Bestellstatus

Setzen Sie den Status nie auf **Pending Unzer** oder einen ähnlich lautenden pending status, der durch das Modul eingeführt wird.

6.2.1 Bestellstatus "ausstehend" / bevorstehende Zahlung (oder ähnlich)

Bestellungen mit dem Status 'pending Unzer' sind pendente Bestellungen. Der Status wird gesetzt, wenn der Kunde für die Zahlung umgeleitet wird aber nicht erfolgreich zurückkehrt oder das Feedback nicht bis zu Ihrem Shop gekommen ist (Kunde hat das Fenster auf der Payment Page geschlossen und die Zahlung nicht abgeschlossen). Abhängig von der Zahlart werden diese Bestellungen automatisch in abgebrochene Bestellungen umgewandelt und der Lagerbestand wieder freigegeben (sofern der Cronjob Aktiviert ist). Diese Zeitspanne hängt von den Eigenheiten der Zahlart ab, und kann nicht konfiguriert werden.

Wenn Sie eine Vielzahl ausstehender Bestellungen haben, deutet dies in der Regel darauf hin, dass die Notifikationen von Ihrem Webserver zu Unzer blockiert werden. In diesem Fall überprüfen Sie die Einstellungen Ihrer Firewall und bitten den Hoster die IPs und User Agents von Unzer zu aktivieren.

6.2.2 Abgebrochen / stornierte Bestellungen

Bestellungen, die als abgebrochen markiert werden, wurden entweder wie oben beschrieben automatisch in diesen Status aufgrund des Timeouts versetzt. Abgebrochen werden aber auch alle Bestellungen, welche von den Kunden aktiv abgebrochen wurden.

6.3 Rechnung

Das grundlegende Konzept der Rechnung ist, dass die Ware versandt wird, bevor der Käufer den ausstehenden Betrag bezahlt. Das bedeutet, dass der Käufer eine begrenzte Zeit hat, um die Zahlung auf das Bankkonto des Händlers vorzunehmen.

Für den Fall, dass der Käufer einige der Artikel zurückgibt, sollten Sie eine Rückerstattung erstellen. Dies funktioniert auch, wenn der Käufer die Rechnung noch nicht bezahlt hat. In diesem Fall wird der ausstehende Betrag entsprechend reduziert.

6.4 Prepayment

Bei Vorkasse hat der Käufer die Rechnung zu bezahlen, bevor der Händler die Ware versendet.

Um diesen Prozess zu realisieren, muss der Shop über Zahlungseingänge informiert werden. Dazu wird die Bestellung als unsicher markiert, bis Unser den Shop über eine Hintergrundabfrage über den erfolgreichen Zahlungseingang benachrichtigt. Dies führt dazu, dass der unsichere Zustand aufgehoben wird und ist der Auslöser, den Versandprozess zu starten.

Für den Fall, dass der Kunde sich entscheidet, nicht alle Artikel zu bestellen, können Sie eine Rückerstattung erstellen. Dies ist möglich, noch bevor der Käufer den Betrag bezahlt hat. In diesem Fall wird der ausstehende Betrag reduziert, so dass die geringere Zahlung des Käufers dennoch den Versandprozess durch Aufhebung des unsicheren Zustands auslöst.

6.5 Ratenzahlung

Ratenzahlung als Zahlungsmethode ermöglicht es dem Käufer, den ausstehenden Betrag in mehreren Tranchen zu bezahlen.

Ähnlich wie bei der Rechnung stützt sich die Ratenzahlungsmethode auf die Liefermeldung, um die Fälligkeitsdaten der verschiedenen Zahlungen zu bestimmen. Sie können die aufgeschobene Capture-Operation verwenden, um Unser wissen zu lassen, wann die Sendung fertig ist. Falls Sie den Betrag reduzieren möchten, können Sie dies als Teil der Erfassung tun. Das Plugin führt entsprechende Stornoanfragen aus und sendet die Liefermeldung an Unser.

Obwohl es möglich ist, die Lieferbenachrichtigung, genannt Finalize, in Unser's Insights auszulösen, empfehlen wir Ihnen dringend, dies im Backend des Shops zu tun.

Falls Sie zu einem späteren Zeitpunkt den Betrag der Teilzahlungen reduzieren müssen, können Sie Rückzahlungen vornehmen.

6.6 Gesicherte Rechnung

Das grundlegende Konzept der Rechnung ist, dass die Ware versandt wird, bevor der Käufer den ausstehenden Betrag bezahlt. Das bedeutet, dass der Käufer eine begrenzte Zeit hat, um die Zahlung auf das Bankkonto des Händlers vorzunehmen.

Dieser Vorgang hat Auswirkungen auf den Versand. Falls die Ware nicht auf Lager ist, ist es wichtig, den Betrag nicht direkt zu erfassen, da mit der Erfassung das Lieferavis versendet wird und das Fälligkeitsdatum relativ zu diesem Zeitpunkt definiert ist. Mit anderen Worten, Sie sollten die Zahlungsmethode so konfigurieren, dass die Zahlung nur autorisiert und der Betrag erfasst wird, sobald die Waren zum Versand bereit sind. Dieses Verfahren wird auch für andere Zahlungsmethoden empfohlen, wenn die Ware nicht auf Lager sind. Im Unzer's Insights wird die Ausführung der Versandbenachrichtigung als finalize bezeichnet. Der Vorgang finalize wird mit der Erfassung ausgelöst. Wenn nur ein Teil des Betrags erfasst wird, wird die wird der entsprechende Abschlag automatisch angewendet.

Für den Fall, dass der Käufer einige der Artikel zurückgibt, sollten Sie eine Rückerstattung erstellen. Dies funktioniert auch, wenn der Käufer die Rechnung noch nicht bezahlt hat. In diesem Fall wird der ausstehende Betrag entsprechend reduziert.

7 Das Modul in Aktion

In der Folge finden Sie eine Übersicht über die wichtigsten Funktionen im täglichen Gebrauch des Unzer Moduls.

7.1 Empfohlene Arbeitsweise

Grundsätzlich können die wichtigsten Aktionen im Shop Backend oder auch im Unzer Insights ausgelöst werden. Nicht alle Aktionen, die im Insights ausgelöst werden, können sauber übertragen werden in den Shop.

Daher empfehlen wir die Aktionen, wie Gutschriften auslösen etc., immer im Shop auszulösen. Dadurch ist sichergestellt, dass der Shop und Unzer übereinstimmen.

7.2 Nützliche Transaktionsinformationen zur Bestellung

Sie finden in jeder Bestellung, welche mit unserem Modul abgewickelt wurde, eine Übersicht über die wichtigsten Informationen zur Transaktion. Unter anderem erlauben diese Informationen die eindeutige Zuordnung der Bestellungen zu den Transaktionen, welche Sie bei Unzer im Backend sehen.

Orders	
Order Details	Order ID: #2
Payment Details	Invoice No.: [Generate]
Shipping Details	Store Name: Your Store
Products	Store Url: http://demo.sellxed.com/business/
History	Customer: Sascha Krügel
	Customer Group: Default
	E-Mail: info@customweb.ch
	Telephone: 33123456798
	Total: 78.90€
	Order Status: Pending
	IP Address: 160.85.155.47
	User Agent: Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Ubuntu Chromium/30.0.1599.114 Chrome/30.0.1599.114 Safari/537.36
	Accept Language: de,en-US;q=0.8,en;q=0.6
	Date Added: 13/11/2013
	Date Modified: 13/11/2013

Abbildung 7.1: Transaktionsinformationen in OpenCart.

7.3 Einpflegen von Zahlungsinformationen eines Verarbeiters

An folgenden Stellen können Sie die Zahlungsinformationen einer "Bestellung auf Rechnung" einsehen bzw. manuell einbetten.

7.3.1 OpenCart Bestellbestätigung (E-Mail)

Die Zahlungsinformationen werden in der Standard Bestellbestätigungsmail von OpenCart dargestellt.

7.3.2 OpenCart Rechnung (PDF)

Dieses Feature ist bei OpenCart standardmässig nicht dabei und kann somit von der Zahlungsschnittstelle nicht bedient werden.

7.3.3 OpenCart-Backend (Transaktionsdetails)

Die Transaktionsdetails können Sie im OpenCart unter **Sales > Unser Transactions** öffnen. Sollten Sie diesen Menüpunkt nicht finden können, müssten Sie bitte unter **Erweiterungen > Modifikationen >** die Modifikationen "bereinigen".

Authorization Type	PaymentPage	New Authorization
Order Status	authorized	
Recurring	No	
Created At	2017-03-06 15:22:26	
Updated At	2017-03-06 15:22:45	
Alias Active	Yes	
Authorisation Amount	20.3	
Currency	EUR	
Payment Method	Open Invoice	
Payment ID	3571832	
Test Transaction	Yes	
Transaction authorised	Yes	
Transaction uncertain	No	
Transaction paid	Yes	

Transaction History

Date	Action	Message
2017-03-06 15:22:45	authorization	The amount of 20.30 is authorized.

Payment Information ⓘ

HTML	Bank: Customweb Test Bank Account Holder: Customweb GmbH IBAN: DE2501200000TEST000000000003 BIC: TESTBIC0003 Reference Number: BP3571832/2749
TEXT	Bank: Customweb Test Bank Account Holder: Customweb GmbH IBAN: DE2501200000TEST000000000003 BIC: TESTBIC0003 Reference Number: BP3571832/2749

Abbildung 7.1: Transaktionsinformationen innerhalb OpenCart.

7.3.4 OpenCart Success-Page

Aufgrund technischer Einschränkungen ist es momentan nicht möglich die Zahlungsinformationen hier abzubilden.

7.4 Verbuchen / Stornieren von Bestellungen

7.4.1 Verbuchen von Bestellungen

Um Bestellungen zu verbuchen, öffnen Sie den Transaktionsmanager unter **Sales > Unzer Transactions**. Hier finden Sie eine Übersicht über alle Transaktionen. Suchen Sie Die Bestellung im Feld der Bestellnummern. Mit "view" öffnen Sie die Transaktion zur Bestellung.

7.4.1.1 Verbuchen kompletter Bestellungen oder Teilverbuchung (partial Capturing)

Durch den Klick auf den Button "Capture Transaction" öffnet sich ein neues Eingabefeld. Sie können nun jeden Artikel einzeln verbuchen, sofern Sie nicht alle Artikel auf einmal versenden können. Die Anzahl der Verbuchung von Teilbeträgen hängt von Ihrem Vertrag ab. Erkundigen Sie sich diesbezüglich direkt bei Unzer.

Verbuchung von Bestellungen im Backend von Unzer

Das Transaktionsmanagement zwischen Unzer und Ihrem Shop ist nicht synchronisiert. Wenn Sie Zahlungen im Backend von Unzer verbuchen, wird der Status im Shop nicht angepasst und ein zweites Verbuchen im Shop ist nicht möglich.

Falls Sie nicht alle Artikel der Bestellung verbuchen wollen, können Sie die Transaktion durch anklicken des Kästchens auch für spätere Verbuchungen schliessen.

PARTIAL CAPTURE

With the following form you can perform a partial capture.

Name	SKU	Type	Tax Rate	Quantity	Total Amount (excl. Tax)	Total Amount (incl. Tax)
iPhone	product 11	product	0 %	<input type="text" value="1"/>	<input type="text" value="74.24"/>	<input type="text" value="74.24"/>
Flat Shipping Rate	shipping	shipping	0 %	<input type="text" value="1"/>	<input type="text" value="3.68"/>	<input type="text" value="3.68"/>
Total Capture Amount:						77.91 EUR

Close transaction for further captures ☐

[Capture](#)

Abbildung 7.1: Verbuchen von Bestellungen.

Partial Capturing

Erkundigen Sie sich, ob mehrere Verbuchungen von Unser in Ihrem Vertrag unterstützt werden. Sonst kann es vorkommen, dass die Transaktion bei einer Teilverbuchung für spätere Verbuchungen geschlossen bleibt.

7.4.2 Bestellungen stornieren

Mit dem Klick auf "Cancel Transaction" erfolgt ein **Storno** der Transaktion und der reservierte Betrag wird auf der Karte Ihres Kunden sofort freigegeben.

[Capture Transaction](#)
[Refund Transaction](#)

Abbildung 7.1: Capture oder Cancel in OpenCart.

7.5 Rückvergüten von Bestellungen

Um Bestellungen rückzuvergüten, öffnen Sie die Transaktionsinformationen (vgl. oben).

Sie können jeden einzelnen Artikel oder einen beliebigen Betrag rückerstatten indem Sie die Totalbeträge oder die Anzahl Artikel anpassen.

PARTIAL REFUND

With the following form you can perform a partial refund.

Name	SKU	Type	Tax Rate	Quantity	Total Amount (excl. Tax)	Total Amount (incl. Tax)
iPhone	product 11	product	0 %	<input type="text" value="1"/>	<input type="text" value="74.24"/>	<input type="text" value="74.24"/>
Flat Shipping Rate	shipping	shipping	0 %	<input type="text" value="1"/>	<input type="text" value="3.68"/>	<input type="text" value="3.68"/>
Total Refund Amount:						77.91 EUR

Close transaction for further refunds ☐

Refund

Abbildung 7.1: Refunds in OpenCart für Unzer.

Maximaler Refund

Mit unserem Zahlungsmodul können Sie nie mehr als 100% des ursprünglich autorisierten Betrages zurückerstatten.

7.6 Einrichten Cron Job

Um die zeitgesteuerten Funktionen des Plugins zu aktivieren (bspw. Update Service, Löschen pendenteter Bestellungen, etc.) müssen Sie in OpenCart Cron Jobs einrichten. Insbesondere die Update funktion ermöglicht Ihnen nachträgliche Änderungen der Transaktion bei Unzer über die API abzuholen (Bitte beachten Sie es kann sein, dass hier zusätzliche Optionen freigeschaltet werden müssen).

Damit die zeitgesteuerten Aktionen durchgeführt werden richten Sie bitte einen Cron Job in Ihrem Server auf folgenden Controller ein:

<https://www.your-shop.com/index.php?route=unzercw/cron/cron>

Wir empfehlen Ihnen hier eine Cron Engine wie beispielsweise [EasyCron](#). Damit können Sie die Datei (URL) mit einem externen Service aufrufen

7.7 Teilstornierung

Wird das Häkchen während der manuellen Verbuchung bei "Transaktion Schliessen" gesetzt, so wird der Restbetrag storniert.

Möchten Sie einen Teilbetrag stornieren, aber die Transaktion für weitere Buchungen offen lassen, so müssen die Teilstornos in dem Backend von Unzer erstellt werden.

Gutschriften können nur ausgelöst werden, wenn die Transaktion für weitere Teilbuchungen geschlossen wurde.

8 Testing

Bevor Sie vom Test Modus in den Live Modus wechseln, sollten Sie das Modul ausführlich testen.

Testen

Vergessen Sie nicht, nach erfolgreichem Testing den Betriebsmodus von Test auf Live umzustellen.

8.1 Testdaten

Nachfolgend finden Sie die Testdaten für die Zahlungsmethoden:

Credit / Debit Card

Kartenummer	4711 1000 0000 0000	Visa Success
Ablaufdatum	12/2030	
CVC	123	
3D PW	secret!33	
Kartenummer	5453 0100 0005 9543	Mastercard Success
Ablaufdatum	12/2030	
CVC	123	
3D PW	secret3	

WeChat Pay

Username	keychain
Password	123

Alipay

Username	keychain
Password	123

EPS

Bank / Issuer	Stuzza Bank	EPS Simulation
Username	1003993	
Password	rX/'PvZzIW?&	

iDEAL

Account Bankname	ING_TEST
------------------	----------

giropay

Bank number	12345679
Account number	0000000300
IBAN	DE46940594210000012345
BIC	TESTDETT421
User	chiptanscatest2
User PIN	12345

User TAN	123456
TAN mechanism	optical Chip-TAN

SOFORT

Bank number	00000
Account number	123456
User PIN	123456
User TAN	12345

9 Fehler und deren Lösungen

9.1 Die Referrer URL erscheint in meinem Analytics Tool

Unter Umständen kann die Referrer URL von Unser fälschlicherweise im Analytics Tool erscheinen. Dies passiert, wenn der Kunde und die anschließende Notification via Header Redirection weitergeleitet werden. Die meisten Analytic Tools können dieses Problem jedoch minimieren, indem die entsprechende URL aus dem Report ausgeschlossen wird.

Für Google Analytics hilft dieses Stept-by-Step Guide am Beispiel von PayPal: [Weitere Infos sind unter Punkt 4 beschrieben.](#)

10 Kompatibilität zu Third-Party Plugins

Unten aufgeführte Plugins sind kompatibel mit unseren Zahlungsmodulen und erleichtern Ihnen in gewissen Bereichen die Arbeit.

10.1 Geburtstag und Geschlecht in OpenCart

Für gewisse Zahlungsanbieter ist es nötig den Geburtstag sowie das Geschlecht des Kunden zu prüfen. OpenCart tut dies standartmässig nicht.

Geburtstag und Geschlecht Abfragen in ihren Checkout einfügen

1. Fügen Sie dem Checkout Ihres Shops zwei neue Custom Fields hinzu. Dies können Sie im Backend Ihres Shops unter "Customers > Custom Fields" tun.
2. Modifizieren Sie dann die "order context getters" so, dass diese den Wert der Order / Session (oder wo immer der vorgängige Schritt die Daten speichert) zurück geben.

Order Context Getters

- AbstractOrderContext
- getBillingDateOfBirth()
- getBillingGender()

Diese Funktionen finden Sie in "system/library/cw/Unzer/AbstractOrderContext.php".

11 Error Logging

Das Modul logged verschiedene unerwartete Fehler und sonstige Fehler abhängig vom konfigurierten Logging Level. Sollte es ein Problem mit dem Modul geben kann das Log Hinweise auf dessen Ursache liefern.

11.1 Log Levels

Sie können das Log Level in den Einstellungen von Unser konfigurieren.

- Error: Logt nur unerwartete Fehler. (Standard)
- Info: Logt zusätzliche Informationen.
- Debug: Logt debugging relevante Informationen.

11.2 Pfad der Log Datei

Die Log Datei wird standardmässig im Log Ordner von OpenCart gespeichert. Der Pfad wird in der config.php Ihres Shops gespeichert. (Standard Pfad: {shopRootDirectory}/system/logs oder {shopRootDirectory}/system/storage/logs)

12 Erweiterte Informationen

Dieser Abschnitt des Manuals beinhaltet Informationen für Anwender mit speziellen Anforderungen. Sämtliche hier aufgeführte Informationen und Konfigurationen sind optional und für die normale Verwendung des Plugins nicht erforderlich

12.1 Transaction Object

In diesem Abschnitt finden Sie Informationen wie Sie Daten zu einer Transaktionen aus der Datenbank auslesen können um diese, z.B. in einem ERP System, weiter zu verarbeiten.

Für die Code Snippets in diesem Abschnitt wird angenommen, dass sich Ihr Script im Stammordner des Shops, in einer standard Ordnerstruktur befindet.

In your script initialize the base of OpenCart.

Opencart 1.x

```
require_once('config.php');

require_once(DIR_SYSTEM . 'startup.php');

require_once(DIR_SYSTEM . 'library/customer.php');
require_once(DIR_SYSTEM . 'library/affiliate.php');
require_once(DIR_SYSTEM . 'library/currency.php');
require_once(DIR_SYSTEM . 'library/tax.php');
require_once(DIR_SYSTEM . 'library/weight.php');
require_once(DIR_SYSTEM . 'library/length.php');
require_once(DIR_SYSTEM . 'library/cart.php');

$registry = new Registry();
$loader = new Loader($registry);
$registry->set('load', $loader);
$config = new Config();
$registry->set('config', $config);
$db = new DB(DB_DRIVER, DB_HOSTNAME, DB_USERNAME, DB_PASSWORD,
DB_DATABASE);
$registry->set('db', $db);
```

Opencart 2.x

```
require_once('config.php');
require_once(DIR_SYSTEM . 'startup.php');
// Registry
$registry = new Registry();

// Config
$config = new Config();
```

```
$config->load('default');
$config->load('catalog');
$registry->set('config', $config);
$loader = new Loader($registry);
$registry->set('load', $loader);
$registry->set('db', new DB($config->get('db_engine'), $config->get(
('db_hostname'), $config->get('db_username'), $config->get
('db_password'), $config->get('db_database'), $config->get('db_port'))));
```

Opencart 3.x

```
require_once('config.php');
require_once(DIR_SYSTEM . 'startup.php');
// Registry
$registry = new Registry();

// Config
$config = new Config();
$config->load('default');
$config->load('catalog');
$registry->set('config', $config);
$loader = new Loader($registry);
$registry->set('load', $loader);
$registry->set('db', new DB($config->get('db_type'), $config->get
('db_hostname'), $config->get('db_username'), $config->get
('db_password'), $config->get('db_database'), $config->get('db_port')));
```

Include the module specific files and set registry.

```
require_once DIR_SYSTEM.'library/cw/init.php';
require_once DIR_SYSTEM.'library/cw/UnzerCw/Util.php';
require_once DIR_SYSTEM.'library/cw/UnzerCw/Entity/Transaction.php';
UnzerCw_Util::setRegistry($registry);
```

Now you can load the transaction and then extract the transactionObject.

Load the transaction by Id:

```
$transactionById = UnzerCw_Entity_Transaction::loadById($transactionId);
$transactionObject = $transactionById->getTransactionObject();
```

Load transactions by Order ID:

```
$transactionsByOrderId = UnzerCw_Entity_Transaction::
getTransactionsByOrderId($orderId);
foreach($transactionsByOrderId as $transaction){
    $transactionObject = $transaction->getTransactionObject();
    //Do something with each object
}
```