

Installationsanleitung für Dimoco WooCommerce

Diese Anleitung erklärt die Installation und Benutzung der Dimoco Extension für WooCommerce.

Release Datum: Mon, 07 Aug 2023 14:01:06 +0200
Version: 2.0.97

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

E-Mail: info@sellxed.com

Telefon:

CH: +41 (0)44 505 13 60

DE: +49 (0)40 2999 60117

UK: +44 (0)161 850 6890

US: +1 (0)205 557 5869

© copyright by wallee AG Mon, 07 Aug 2023 14:01:06 +0200

Inhaltsverzeichnis

1	Einleitung	5
1.1	Ablauf der Installation	5
1.2	System Anforderungen	6
2	Konfiguration des Zahlungsmoduls und der Zahlungsarten	7
2.1	Live und Test Zugangsdaten	8
2.1.1	Live Zugangsdaten	8
2.1.2	Test Zugangsdaten	8
2.1.3	Zahlungsartenspezifische Einstellungen	9
2.2	Kreditkarten 3D Secure	9
2.3	Integrated Payment Form (Ajax Authorization)	9
2.4	SEPA Direct Debits Mandates	9
3	Modul Installation im WooCommerce Shop	10
3.1	Installation	10
3.2	Updates und Upgrades	10
3.2.1	Update Checklist	10
3.2.2	Update Instruktionen	11
4	Modul Konfiguration im WooCommerce Shop	12
4.1	Konfiguration des Hauptmoduls	12
4.6	Weiterleitungsfehler / Fehlerhaftes Zahlungsformular	14
4.3	Direktes Verbuchen von Bestellungen	13
4.4	Unsicherer Status	13
4.4.1	Setzen von Order-Status	13
4.5	Optional: Validierung	13
5	Einstellungen / Konfiguration von Zahlungsarten	15
5.1	Allgemeine Informationen zu den Zahlungsarten	15
5.2	Informationen zum Zahlungszustand	15
5.2.1	Bestellstatus "ausstehend" / bevorstehende Zahlung (oder ähnlich)	15
5.2.2	Abgebrochene / stornierte Bestellungen	16
6	Das Modul in Aktion	17
6.1	Nützliche Transaktionsinformationen zur Bestellung	17
6.2	Nutzung des Alias Managers / Token Lösung	17
6.3	Verbuchen / Stornieren von Bestellungen	18

6.4	Gutschriften	20
6.5	Transaktionsverlauf	21
6.6	Einrichten Cron Job	22
7	Testing	23
7.1	Testdaten	23
8	Fehler und deren Lösungen	27
8.1	Page Not Found Error vor der Weiterleitung zur Payment Page	27
8.2	Die Referrer URL erscheint in meinem Analytics Tool	27
9	Kompatibilität zu Third-Party Plugins	28
9.1	Gebühren und Rabatte in WooCommerce	28
9.2	Mehrsprachigkeit in WooCommerce	28
9.3	Geburtstag und Geschlecht in WooCommerce	28
10	Error Logging	29
11	Erweiterte Informationen	30
11.1	Transaction Object	30

1 Einleitung

Diese Anleitung beschreibt die Installation, die Konfiguration und die Nutzung des Zahlungsmoduls für WooCommerce und DIMOCO Payment Services (Dimoco).

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

- Dimoco API Benutzername and Passwort
- Zugänge zum Merchant Account von Dimoco - Hier finden Sie alle Zugänge für die Konfiguration des Moduls
- WooCommerce Zahlungsmodul von sellxed.com/shop
- Zugangsdaten zu Ihrem Server und Shop

Sollten Ihnen einige Daten Fehlen nehmen Sie bitte direkt mit Dimoco Kontakt auf. Sollten Sie noch keinen Vertrag mit Dimoco besitzen, bitten wir Sie mit der Sales Abteilung von Dimoco in Kontakt zu treten (payment-merchants@dimoco.eu).

Beachten Sie, dass Sie für unsere Plugins mindestens die PHP Version 5.6 verwenden müssen. PHP 8 oder höher wird derzeit nicht unterstützt.

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.

Die Installation besteht aus folgenden Schritten:

1. Konfiguration der [Grundeinstellungen](#) des Zahlungsmoduls
2. Konfiguration der [Zahlungsart](#) Dimoco
3. Durchführung einer Testbestellung
4. Kontakt aufnehmen mit dem zuständigen Technical Account Manager bei Dimoco. Dieser wird Ihnen helfen Ihr Konto für den Livebetrieb richtig zu konfigurieren.
5. Sofern der Test erfolgreich verlaufen ist, schalten Sie den Betriebsmodus in den Live Modus und nehmen im Anschluss erneut Kontakt mit dem Technical Account Manager auf.

Installationservice

Unsere Zahlungsmodule sind standardmässig so vorkonfiguriert, dass die Installation gleich nach dem Eintragen der benötigten Keys funktionieren sollte. Sollten Sie einen Setting aus der Konfiguration des Moduls nicht kennen, können Sie sich gerne mit unserem Support in Verbindung setzen.

Unser Support Team steht Ihnen während der Geschäftszeiten gerne zur Verfügung: <http://www.sellxed.com/support>. Sie haben zudem jederzeit die Möglichkeit unseren Installationservice zu bestellen. Wir sorgen für eine reibungslose Installation in Ihrem Shop: <http://www.sellxed.com/shop/de/integration-und-installation.html>

.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 Dimoco nicht bis zu Ihrem Shop durchdringt.

1.2 System Anforderungen

Generell hat das Plugin die gleichen Anforderungen an das System wie WooCommerce 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.

2 Konfiguration des Zahlungsmoduls und der Zahlungsarten

Für die Konfiguration des [Hauptmoduls](#) in WooCommerce benötigen Sie folgende Informationen:

- API Benutzername (Erhalten Sie von Dimoco im Setup Email)
- API Passwort (Erhalten Sie von Dimoco im Setup Email)
- API Key (Aus den Einstellungen des Connectors)
- Shared Secret (Aus den Einstellungen des Connectors)

Diese Informationen erhalten Sie direkt von Dimoco oder sind direkt in Ihrem Dimoco Konto im Tab 'Connectors' ersichtlich.

The screenshot shows the Dimoco dashboard with a sidebar on the left containing navigation options: Gateway, Dashboard, Statistics, Connectors, and Transactions. The main content area is titled 'Dashboard' and features a 'Last Transactions' table. The table has columns for Date, TransactionId, Type, Amount, Status, and Connector. Each row includes a 'Details' button. The transactions listed are all 'debit' type, with amounts ranging from 65.00 EUR to 280.00 EUR, and statuses of either 'error' or 'success'. The connector for all transactions is 'testshop Creditcard with PCI'.

Date	TransactionId	Type	Amount	Status	Connector
2018-01-17 12:43	14500039	debit	65.00 EUR	error	testshop Creditcard with PCI
2018-01-17 10:36	14500038	debit	80.00 EUR	success	testshop Creditcard with PCI
2018-01-17 10:33	14500037	debit	65.00 EUR	success	testshop Creditcard with PCI
2018-01-16 15:44	14500036	debit	80.00 EUR	error	testshop Creditcard with PCI
2018-01-16 07:58	14500035	debit	280.00 EUR	success	testshop Creditcard with PCI
2018-01-15 15:38	14500034	debit	65.00 EUR	success	testshop Creditcard with PCI
2018-01-15 13:20	14500033	debit	65.00 EUR	success	testshop Creditcard with PCI
2018-01-15 13:03	14500032	debit	65.00 EUR	success	testshop Creditcard with PCI

Abbildung 2.1: Übersicht des Dashboards (My Payments) bei Dimoco.

Tragen Sie diese Informationen direkt im [Hauptmodul](#) in WooCommerce ein. Für alle weiteren Einstellungsmöglichkeiten Ihres Dimoco Accounts, verweisen wir Sie hiermit auf die offiziellen Handbücher von Dimoco.

2.1 Live und Test Zugangsdaten

Bitte beachten Sie, dass die Konfiguration für Live und Test Modus sich von einander unterscheiden. Die Unterschiede werden in den nächsten zwei Abschnitten beschrieben.

2.1.1 Live Zugangsdaten

Geben Sie den API Username sowie das API Password in die Hauptkonfiguration des Moduls ein. In Ihrem Dimoco Account unter "Connectors" finden Sie einen Connector für jede Zahlart welche von Dimoco eingerichtet wurde.

In diesen Connectors finden Sie die Parameter API Key sowie Shared Secret. Geben Sie nun für jeden Connector den API Key sowie das Shared Secret in die Konfiguration der [entsprechenden Zahlungsart ein](#).

2.1.2 Test Zugangsdaten

Allgemein kann ein Testconnector für alle Zahlarten hinterlegt werden. Spezifisch für Kreditkarten und SEPA ist es möglich einen eigenen Testconnector zu hinterlegen. Ist dieser nicht hinterlegt wird auf den allgemeinen Testconnector zugegriffen. Diese Connectors werden verwendet um unterschiedliche Zahlungsflüsse zu testen.

The screenshot shows the Dimoco payment services interface. On the left is a green sidebar with navigation options: Gateway, Dashboard, Statistics, Connectors, and Transactions. The main content area displays the configuration for a connector named "testshop Credit Card Dummy". The connector is in "TEST" mode and uses the "Creditcard" payment method. The API Key is "APIKEY1234567890" and the Shared Secret is "12345Z9eECn1pzhBtWLIXINftlwApus".

Base information	
Name	testshop Credit Card Dummy
Test/Live Mode	TEST
Payment Method	Creditcard
Disabled	No
Api Key	APIKEY1234567890
Shared Secret	12345Z9eECn1pzhBtWLIXINftlwApus

Abbildung 2.1: Die Einstellungen innerhalb des Connectors bei Dimoco.

Die Connectors für den Live sowie den Testmodus werden von Dimoco eingerichtet. Falls Sie zusätzliche Connectors eingerichtet haben möchten kontaktieren Sie Dimoco

2.1.3 Zahlungsartspezifische Einstellungen

2.2 Kreditkarten 3D Secure

Bei den Einstellungen der Zahlungsart Kreditkarte (Credit Card) besteht die Möglichkeit 3D Secure zu aktivieren oder zu deaktivieren. Sie können eine 3D Secure Authentifizierung zwingend erfordern, als optional belassen oder ganz deaktivieren. Setzen Sie dafür das Setting '3D Secure' auf Off, Optional oder Mandatory.

2.3 Integrated Payment Form (Ajax Authorization)

Falls Sie eine nahtlose Integration der Zahlungsart Kreditkarte wünschen, können Sie die Zahlungsart Kreditkarte Ajax (Credit Card Ajax) aktivieren. Für dies benötigen Sie eine 'Public Integration Key' die von Dimoco zur Verfügung gestellt wird. Sie können das CVV Element mittels CSS anpassen. Das Setting 'CVV Refresh' entscheidet ob Ihre Kunden den CVV Code bei wiederkehrenden Zahlungen (Alias Manager) erneut eingeben müssen.

2.4 SEPA Direct Debits Mandates

Falls Sie die Zahlungsart SEPA Direct Debits aktivieren möchten, müssen Sie unbedingt darauf achten, dass die 'Mandate Creditor ID' und die 'Mandate Merchant Name' konfiguriert wird.

3 Modul Installation im WooCommerce Shop

3.1 Installation

Sie sollten zu diesem Zeitpunkt bereits im Besitz des Moduls sein. Falls nicht erhalten Sie die Moduldateien in Ihrem Benutzerkonto im [sellxed Shop](#) (Menüpunkt "Meine Downloadartikel"). Um das Modul in Ihrem WooCommerce Shop zu installieren, führen Sie bitte folgende Schritte durch:

1. Laden Sie das Plugin herunter. Den Download finden Sie in Ihrem sellxed.com Account unter "Meine Downloadartikel".
2. Entpacken Sie das Archive, welches Sie heruntergeladen haben.
3. Im entpackten Ordner finden Sie einen Ordner "files"
4. Für einige Shops werden verschiedene Versionen des Plugins zur Verfügung gestellt.
5. 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, welcher 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.
6. Falls nicht bereits geschehen, loggen Sie sich bitte in Ihrem Shop ein.

3.2 Updates und Upgrades

Während der Supportdauer, erhalten Sie unbeschränkten Zugriff auf Updates und Upgrades in Ihrem Kundenkonto. Um über solche Updates informiert zu werden abonnieren Sie bitte den RSS-Feed. Mehr Informationen hierzu finden Sie hier: http://www.sellxed.com/de/updates_upgrades.

3.2.1 Update Checklist

Bitte führen Sie vor jedem Update 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, das Update zu einem Zeitpunkt durchzuführen, an dem nicht zu viele Kunden auf Ihrem Shop zugreifen.

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. Bei Updates direkt im Live-System kann es jedoch zu Unterbrüchen kommen, welche je nach Support Ressourcen bis zu 2 Tage dauern können.

3.2.2 Update Instruktionen

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

4 Modul Konfiguration im WooCommerce 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.

4.1 Konfiguration des Hauptmoduls

Sie aktivieren das Plugin indem Sie im Menü unter **Plugins** das Modul WooCommerce Dimoco aktivieren.

Sie finden die Einstellungen des Hauptmoduls nach der Installation des Moduls unter dem Menüpunkt **Dimoco**. Tragen Sie hier die Daten in die einzelnen Felder ein. Die benötigten Daten haben Sie entweder bereits im Backend von Dimoco eingetragen oder von Dimoco in vorheriger Korrespondenz erhalten. Jede Option wird zudem mit einem kurzen Info Text im Shop nochamls erläutert.

4.2 Konfiguration der Zahlungsart

Nachdem das Hauptmodul erfolgreich konfiguriert wurde, finden Sie unter **WooCommerce > Settings > Checkout** (In älteren Version lautet der Pfad: **WooCommerce > Settings > Payment Gateways**) die Einstellungen zu den einzelnen Zahlungsarten in Ihrem Shop. Jede Zahlungsart 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 **Enable** wird eine Zahlungsmethode in Ihrem Shop aktiviert. Für die detaillierten Einstellungen sei an dieser Stelle auf die Informationen zu den einzelnen Einstellungen direkt im Modul verwiesen.

4.3 Direktes Verbuchen von Bestellungen

Mit der Option "Buchung" legen Sie fest, ob Zahlungen direkt verbucht, oder zuerst autorisiert werden. 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 Dimoco - Backend und Modul

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

4.4 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.

4.4.1 Setzen von Order-Status

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

4.5 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 erfolgen 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. Damit die Bonitätsprüfung beziehungsweise Adressvalidierung auch mit europäischen Zeichen funktioniert, muss bei gewissen PSP-Einstellungen das Charset des "Blowfish mode" auf "UTF-8" gesetzt werden.

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

- **Validierung vor Auswahl der Zahlungsmethode:** Es erfolgt vor der Auswahl der Methode eine 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 Dimoco während dem Autorisierungsprozess. Die Zahlungsmethode wird auf jeden Fall angezeigt.

4.6 Weiterleitungsfehler / Fehlerhaftes Zahlungsformular

Falls die erfolgreiche Weiterleitung zur Zahlungsseite beziehungsweise das Anzeigen des Zahlungsformulars nicht mehr funktioniert (es wird zum Beispiel nur Header und Footer geladen oder ein 404 Fehler erscheint), ist Ihre WooCommerce Installation sehr wahrscheinlich von diesem Fehler betroffen. In einem ersten Schritt sollten Sie das Zahlungsmodul aktualisieren, das Plugin im Backend von WooCommerce deaktivieren und anschliessend wieder aktivieren.

Sollten Sie **WPML** verwenden kann es sein, dass die Pages für die Darstellung der Zahlungsarten oder für die Weiterleitung zu Dimoco in einem 404 Page Not Found Error enden. In diesem Fall müssen Sie unsere Pages noch in die entsprechenden Sprachen richtig duplizieren. Dafür führen Sie bitte folgende Schritte durch:

1. Im WooCommerce Backend sollten Sie unter Pages > All Pages die Seite "Dimoco Checkout" finden.
2. Löschen Sie in jeder installierten Sprache die Seite "Dimoco Checkout".
3. In der Pluginansicht, müssen Sie als nächstes das WooCommerce Dimoco Modul deaktivieren. Gleich im Anschluss aktivieren Sie das Modul erneut. Dies sorgt dafür, dass die Weiterleitungsseite erneut generiert wird.
4. Kehren Sie zur Ansicht Pages > All Pages zurück und öffnen Sie die Seite "Dimoco Checkout".
5. Auf der rechten Seite unter Languages können Sie den Inhalt dieser Seite nun für alle von Ihnen installierten Sprachen duplizieren. Nach dem Speichern ist diese Seite für alle Ihre Sprachen aktiv und der Fehler sollte verschwunden sein.

5 Einstellungen / Konfiguration von Zahlungsarten

5.1 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 mit uns in Verbindung zu setzen.

Für die Nutzung einer Zahlungsmethode muss die Zahlungsart in Ihrem Konto bei Dimoco 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.

5.2 Informationen zum Zahlungszustand

Sie können für jede Zahlungsart den 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, welcher 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 Dimoco** oder einen ähnlich lautenden pending status, der durch das Modul eingeführt wird.

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

Bestellungen mit dem Status 'pending Dimoco' 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 Zahlungsart 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 Zahlungsart 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 Dimoco blockiert werden. In diesem Fall überprüfen Sie die Einstellungen Ihrer Firewall und bitten den Hoster die IPs und User Agents von Dimoco zu aktivieren.

5.2.2 Abgebrochene / stornierte Bestellungen

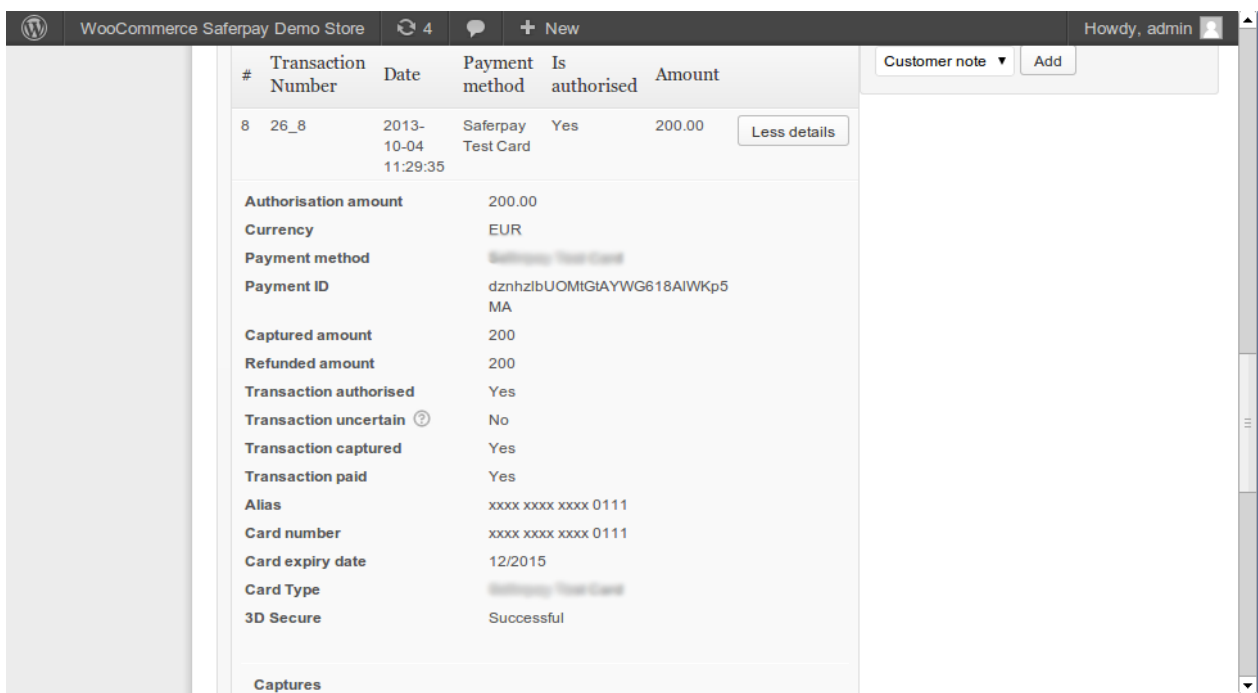
Bestellungen, die als abgebrochen markiert werden, wurden wie oben beschrieben automatisch in diesen Status aufgrund des Timeouts versetzt. Abgebrochen werden aber auch alle Bestellungen, welche von den Kunden aktiv abgebrochen wurden.

6 Das Modul in Aktion

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

6.1 Nützliche Transaktionsinformationen zur Bestellung

Sie finden in jeder Bestellung, die 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 Dimoco im Backend sehen.



#	Transaction Number	Date	Payment method	Is authorised	Amount	Customer note	Add
8	26_8	2013-10-04 11:29:35	Saferpay Test Card	Yes	200.00		Less details
Authorisation amount		200.00					
Currency		EUR					
Payment method		Saferpay Test Card					
Payment ID		dzhzlbUOMIGAYWG618AIWKp5MA					
Captured amount		200					
Refunded amount		200					
Transaction authorised		Yes					
Transaction uncertain ?		No					
Transaction captured		Yes					
Transaction paid		Yes					
Alias		xxxx xxxx xxxx 0111					
Card number		xxxx xxxx xxxx 0111					
Card expiry date		12/2015					
Card Type		Saferpay Test Card					
3D Secure		Successful					
Captures							

Abbildung 6.1: Transaktionsinformationen in WooCommerce.

6.2 Nutzung des Alias Managers / Token Lösung

Mit dem Alias Manager können Kreditkarten Ihrer Kunden für spätere Einkäufe sicher bei Dimoco gespeichert werden. Die Nutzung wird aktiviert indem Sie die Option "Alias Manager" in der [Konfiguration der Zahlungsart](#) aktivieren. Für spätere Einkäufe kann der Kunde zwischen den bereits gespeicherten Karten wählen und muss so nicht die gesamte Karte nochmals erfassen.

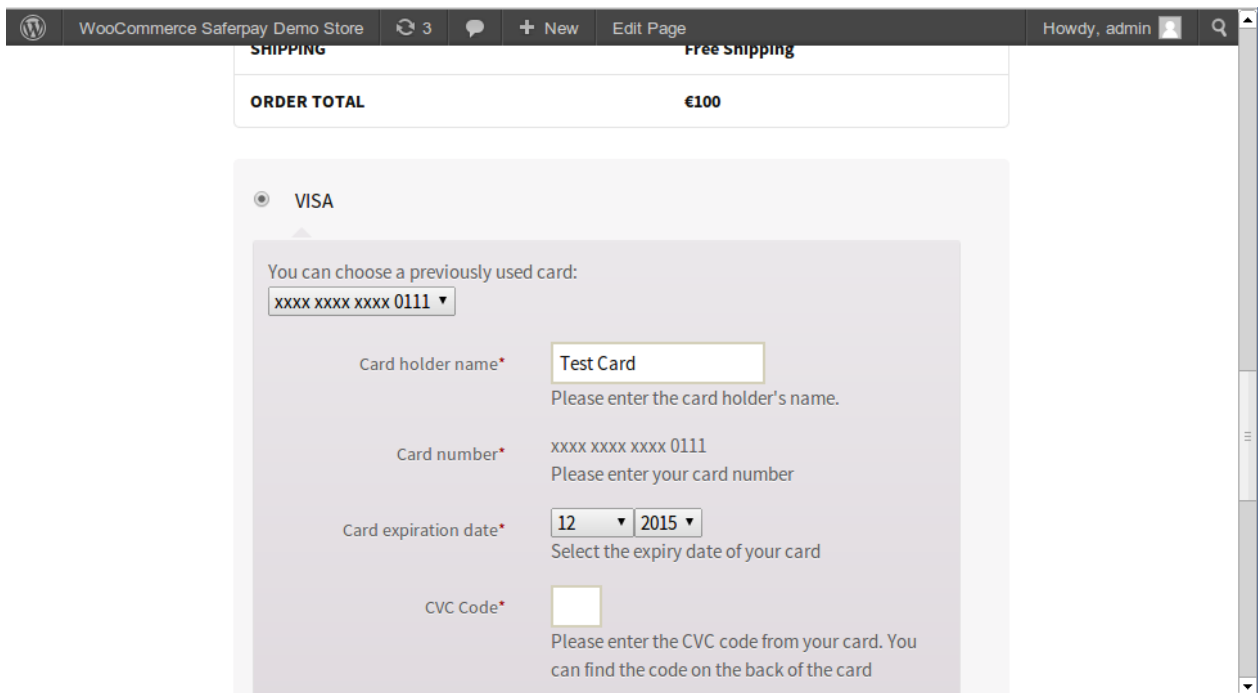


Abbildung 6.1: Alias Manager in WooCommerce

Alias Manager Optionen

Die Nutzung des Alias Managers benötigt die Freischaltung der entsprechenden Option bei Dimoco. Nehmen Sie dafür bitte direkt mit dem Support kontakt auf.

6.3 Verbuchen / Stornieren von Bestellungen

Beachten Sie

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

Damit Sie Bestellungen manuell verbuchen können, müssen Sie bitte sicherstellen, dass Sie in den Einstellungen in der [Zahlungsart](#) die Verbuchung auf **"verzögert"** eingestellt haben.

Öffnen Sie die Bestellung und suchen Sie den Tab Dimoco Transaktionen am unteren Ende. Ein Capturing via WooCommerce nativ Integration ist nicht implementiert.

Customweb Transactions

#	Transaction Number	Date	Payment Method	Authorised	Amount
6	16	2017-01-06 10:23:49	MasterCard	Yes	10.00

[Less Details](#)

Authorisation Amount	10	Currency	CHF
Payment Method	MasterCard	Payment ID	452Wtjb3E7d7tASKGxK1bCEWxhdA
Test Transaction	Yes	Transaction authorised	Yes
Transaction uncertain	No	Transaction paid	Yes
Payment Method Name	MasterCard	Payment Details	xxxx xxxx xxxx 0000
Card holder name	Customweb	Card Expiry Date	12 / 20
Card Country Origin	DE	Acquirer Reference	000000
3ds Authentication	Yes	3ds Liability Shift	Yes
Authorisation Method	PaymentPage		

Previous actions

Date	Action	Message
2017-01-06 10:24:21	authorization	The amount of 10.00 is authorized.

Abbildung 6.1: Transaktionsansicht im Transaktionstab nachdem die Bestellung geöffnet wurde.

Hier können Sie den Button "More Details" anklicken um die Transaktionsdetails einzusehen sowie den Button **Capturing** anklicken um in folgenden Kontext zu gelangen.

Name	SKU	Type	Tax	Quantity	Total Amount (excl. Tax)	Total Amount (incl. Tax)
Testproduct	Testproduct	product	0 %	1	10.00	10.00

Total captured amount: 10.00CHF

Abbildung 6.1: Verzögerte Verbuchung von Transaktionen in WooCommerce

Tragen Sie die Beträge für die entsprechenden Produkte ein, die Sie dem Kunden belasten möchten. Mit dem Klick auf **Capturing** erfolgt eine direkte Verbuchung der Bestellung bei Dimoco.

Bitte beachten Sie, dass die Stornierung der Bestellung keine Statusänderung auslöst.

6.4 Gutschriften

Für belastete Transaktionen können Sie Gutschriften direkt im Shop erstellen und automatisch an Dimoco übermitteln.

Beachten Sie

Das die WooCommerce shopinterne Funktion, um Gutschriften durchführen zu können, nicht unterstützt wird.

Öffnen Sie die Bestellung und suchen Sie den Tab Dimoco Transaktionen am unteren Ende.

Hier können Sie den Button "More Details" anklicken um die Transaktionsdetails einzusehen sowie den Button **Refund Transaction** anklicken.

Customweb Transactions ▲

#	Transaction Number	Date	Payment Method	Authorised	Amount
4	13	2016-11-04 14:50:07	MasterCard	Yes	20.00

[Less Details](#)

Authorisation Amount	20	Currency	EUR
Payment Method	MasterCard	Payment ID	b3YMh8bj8d4pvAnj0ddGAGvj3t9A
Captured amount	20	Test Transaction [?]	Yes
Transaction authorised	Yes	Transaction uncertain [?]	No
Transaction captured	Yes	Transaction paid	Yes
Payment Method Name	MasterCard	Payment Details	xxxx xxxx xxxx 0000
Card holder name	Customweb	Card Expiry Date	12 / 20
Card Country Origin	DE	Acquirer Reference	000000
3ds Authentication	Yes	3ds Liability Shift	Yes
Authorisation Method	PaymentPage		

[Refund](#)

Captures

Date	Amount	State
2016-11-04 14:51:34	20	succeed

Previous actions

Date	Action	Message
2016-11-04 14:51:33	authorization	The amount of 20.00 is authorized.
2016-11-04 14:51:34	capturing	The amount of 20.00 is captured.

Abbildung 6.1: Transaktionsansicht im Transaktionstab nachdem die Bestellung geöffnet wurde.

Klicken Sie den Button **Refund** um in folgenden Kontext zu gelangen.

Name	SKU	Type	Tax	Quantity	Total Amount (excl. Tax)	Total Amount (incl. Tax)
Testproduct	Testproduct	product	0 %	2	20.00	20.00

Total refunded amount: 20.00CHF

Close transaction for further refunds

Abbildung 6.1: Rückerstattung von Transaktionen in WooCommerce

Tragen Sie die Beträge für die entsprechenden Produkte ein, die Sie dem Kunden gutschreiben möchten. Mit dem Klick auf **Gutschrift** wird die Gutschriftsanfrage direkt an Dimoco übermittelt und die Kreditkarte Ihres Kunden um den eingegebenen Betrag entlastet.

Bitte beachten Sie, dass Sie maximal 100% des ursprünglich gebuchten Betrags rückvergüten können.

Bitte beachten Sie weiter, dass die Rückerstattung für die Bestellung keine Statusänderung auslöst.

6.5 Transaktionsverlauf

Jede Aktion im Rahmen des Transaktionsmanagements wird durch das Modul geloggt.

The screenshot shows the 'WooCommerce Saferpay Demo Store' interface. At the top, there's a navigation bar with 'WooCommerce Saferpay Demo Store', a refresh icon, a chat icon, and a '+ New' button. The user is logged in as 'Howdy, admin'. The main content area displays a transaction summary for '2013-11-28 14:07:56' with an amount of '100'. Below this, there's a 'Refund transaction' section with 'Amount to refund: 100' and 'Maximal refundable amount: 100'. There's a checkbox for 'Close transaction for further refunds' and a 'Refund' button. A 'Previous Actions' section follows, containing a table with columns 'Date', 'Action', and 'Message':

Date	Action	Message
2013-11-28 13:55:25	authorization	The amount of 100.00 is authorized.
2013-11-28 14:07:56	capturing	The amount of 100.00 is captured.

At the bottom, there's a 'Downloadable Product Permissions [?]' section with a dropdown menu 'Choose a downloadable product...' and a 'Grant Access' button.

Abbildung 6.1: Transaktionen in der Übersicht im Transaktionsverlauf

6.6 Einrichten Cron Job

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

Dass Modul verwendet den Standard WooCommerce Cron. Informationen zur Einrichtung finden Sie [hier](#).

7 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.

7.1 Testdaten

Nachfolgend finden Sie die Testdaten für die Zahlungsmethoden:

Credit Card

Kartennummer	4111 1111 1111 1111	Visa Success
Ablaufdatum	12/2020	
CVC	123	
Kartennummer	4242 4242 4242 4242	Visa Failure
Ablaufdatum	12/2020	
CVC	123	
Kartennummer	5555 5555 5555 4444	MasterCard Success
Ablaufdatum	12/2020	
CVC	123	
MasterCard (Failure)	5105 1051 0510 5100	MasterCard Failure
Ablaufdatum	12/2020	
CVC	123	

SOFORT

Instructions	Set the last name of the customer to the given value to simulate an outcome.	Simulation control
ignore	Do not simulate any behaviour	
success	The transaction succeeds immediately	
failed	The transaction fails immediately	
redirectSuccess	A redirection occurs, and then the transaction succeeds	
redirectError	A redirection occurs, and then the transaction fails	
pending	The transaction stays pending	
html	Submit a given form, then the transaction succeeds	
htmlError	Submit a given form, then the transaction fails	

giropay

	Set the last name of the customer to the given value to simulate an outcome.	Simulation control
Instructions		
ignore	Do not simulate any behaviour	
success	The transaction succeeds immediately	
failed	The transaction fails immediately	
redirectSuccess	A redirection occurs, and then the transaction succeeds	
redirectError	A redirection occurs, and then the transaction fails	
pending	The transaction stays pending	
html	Submit a given form, then the transaction succeeds	
htmlError	Submit a given form, then the transaction fails	

EPS

	Set the last name of the customer to the given value to simulate an outcome.	Simulation control
Instructions		
ignore	Do not simulate any behaviour	
success	The transaction succeeds immediately	
failed	The transaction fails immediately	
redirectSuccess	A redirection occurs, and then the transaction succeeds	
redirectError	A redirection occurs, and then the transaction fails	
pending	The transaction stays pending	
html	Submit a given form, then the transaction succeeds	
htmlError	Submit a given form, then the transaction fails	

paysafecard

	Set the last name of the customer to the given value to simulate an outcome.	Simulation control
Instructions		
ignore	Do not simulate any behaviour	
success	The transaction succeeds immediately	
failed	The transaction fails immediately	
redirectSuccess	A redirection occurs, and then the transaction succeeds	

	A redirection occurs, and then the
redirectError	transaction fails
pending	The transaction stays pending
html	Submit a given form, then the transaction succeeds
htmlError	Submit a given form, then the transaction fails

Przelewy24

	Set the last name of the customer to the given value to simulate an outcome.	Simulation control
Instructions	Do not simulate any behaviour	
ignore	The transaction succeeds immediately	
success	The transaction fails immediately	
failed	A redirection occurs, and then the transaction succeeds	
redirectSuccess	A redirection occurs, and then the transaction fails	
redirectError	The transaction stays pending	
pending	Submit a given form, then the transaction succeeds	
html	Submit a given form, then the transaction fails	
htmlError		

IDEAL

	Set the last name of the customer to the given value to simulate an outcome.	Simulation control
Instructions	Do not simulate any behaviour	
ignore	The transaction succeeds immediately	
success	The transaction fails immediately	
failed	A redirection occurs, and then the transaction succeeds	
redirectSuccess	A redirection occurs, and then the transaction fails	
redirectError	The transaction stays pending	
pending	Submit a given form, then the transaction succeeds	
html	Submit a given form, then the transaction fails	
htmlError		

Sepa Direct Debits

Simulation control

	Set the last name of the customer to the given value to simulate an
Instructions	outcome.
ignore	Do not simulate any behaviour
success	The transaction succeeds
failed	immediately
redirectSuccess	The transaction fails immediately
redirectError	A redirection occurs, and then the transaction succeeds
pending	A redirection occurs, and then the transaction fails
html	The transaction stays pending
htmlError	Submit a given form, then the transaction succeeds
	Submit a given form, then the transaction fails

8 Fehler und deren Lösungen

Sie finden ausführliche Informationen unter <http://www.sellxed.com/de/fag>. Sollte sich Ihr Problem mit den zur Verfügung gestellten Informationen nicht lösen lassen, melden Sie sich bitte direkt bei uns: <http://www.sellxed.com/de/support>

8.1 Page Not Found Error vor der Weiterleitung zur Payment Page

Sollten Sie WPML verwenden kann es sein, dass die Pages für die Darstellung der Zahlungsarten oder für die Weiterleitung zu Dimoco in einem Page Not Found Error enden.

In diesem Fall müssen Sie unsere Pages noch in die entsprechenden Sprachen duplizieren. Dafür führen Sie bitte folgende Schritte durch:

1. Wählen Sie im Hauptmenü **Pages > All Pages** aus. Dort finden sollten Sie eine Page namens Dimoco Checkout finden.
2. Öffnen Sie die Konfiguration.
3. Auf der rechten Seite unter Languages können Sie den Inhalt dieser Seite nun für alle von Ihnen installierten Sprachen duplizieren. Nach dem Speichern ist diese Seite für alle Ihre Sprachen aktiv und der Fehler sollte verschwunden sein.

8.2 Die Referrer URL erscheint in meinem Analytics Tool

Unter Umständen kann die Referrer URL von Dimoco fälschlicherweise im Analytics Tool erscheinen. Dies passiert, wenn der Kunde mit einer HTTP Header Weiterleitung zurück in den Shop geleitet wird. Die meisten Analytic Tools können dieses Problem jedoch minimieren, indem die entsprechende URL aus dem Report ausgeschlossen wird.

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

9 Kompatibilität zu Third-Party Plugins

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

9.1 Gebühren und Rabatte in WooCommerce

Um Gebühren und Rabatte basierend auf den Zahlungsarten der Dimoco Zahlungsschnittstelle zu konfigurieren, ist es Ihnen möglich folgendes 3rd-Party Plugin zu verwenden.

- [Gebühren Plugin](#)

9.2 Mehrsprachigkeit in WooCommerce

Um mehrsprachige WooCommerce-Websites zu erstellen, benötigen Sie bitte das Folgende 3rd-Party Plugin.

- [WPML für WooCommerce](#)

9.3 Geburtstag und Geschlecht in WooCommerce

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

Geburtstag und Geschlecht Abfragen in Woocommerce einfügen

1. Fügen Sie entweder manuell "custom checkoutfields" in ihren Checkout ein oder laden Sie das Plugin [Flexible Checkout Fields](#) herunter um dies zu 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 "wp-plugins/woocommerce_dimococw/classes /DimocoCw/AbstractOrderContext.php".

10 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 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.

11.1 Transaction Object

In diesem Abschnitt finden Sie Informationen wie Sie Daten zu einer Transaktion 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.

Initialisieren Sie WooCommerce in Ihrem Script, dies ist nötig um die Verbindung zur Datenbank herzustellen.

```
require_once( dirname(__FILE__) . '/wp-load.php' );
```

Intialisieren Sie die Util Klasse des Moduls.

```
require_once ABSPATH.'wp-content/plugins/woocommerce_dimococw/classes  
/DimocoCw/Util.php' ;
```

Nun können Sie die Transaktion laden und das transactionObject extrahieren.

Laden der Transaktion per Id:

```
$transactionById = DimocoCw_Util::getTransactionById($transactionId);  
$transactionObject = $transactionById->getTransactionObject();
```

Laden der Transaktion per external Id:

```
$transactionByExternal = DimocoCw_Util::getTransactionByTransactionNumber  
($externalId);  
$transactionObject = $transactionByExternal->getTransactionObject();
```

Laden der Transaktion per Post ID, dies ist die interne WooCommerce order id:

```
$transactionsByPostId = DimocoCw_Util::getTransactionsByPostId($orderId);  
foreach($transactionsByPostId as $transaction){  
    $transactionObject = $transaction->getTransactionObject();  
    //Do something with each object  
}
```