



Der Weg durch die Technik

Technische Modul-Dokumentation

Der wohl größte Vorteil der eCommerce-Plattform Magento liegt in seiner offenen und erweiterbaren Struktur. Module der Modulwerft können und sollen nicht alle erdenklichen Funktionen abdecken, aber sie sollen einfach und ohne Probleme für Magento-Entwickler erweiterbar sein.

Mit Hilfe dieser technischen Modul-Dokumentation sollen erfahrene Magento-Entwickler weitere Funktionalitäten einfach und schnell hinzufügen können, ohne die Update-Fähigkeit des Moduls zu beeinträchtigen.

Kompetenz, Leidenschaft, Zuverlässigkeit – gemeinsam über die Weltmeere!

Geschenkgutscheine

Einleitung

Diese technische Modul-Dokumentation wendet sich an Magento-Entwickler, die mit den gängigen Programmier-Techniken in Magento vertraut sind. Hierzu zählen allgemein die Anpassung des Systems in Form von Modulen, der Magento-Rewrite-Mechanismus sowie das Event-Observer-Muster.

Die vorliegende Dokumentation beschreibt, wie das vorliegende Modul auf korrekte Art und Weise erweitert werden kann.

Release Notes

Datum	Modul-Version	Änderungen
08.12.2014	1.0.0	<ul style="list-style-type: none">+ Erste Version dieser Dokumentation+ Absatz „Allgemeiner Modulaufbau“+ Absatz „Datenbank-Struktur“+ Absatz „Neue Events“+ Absatz „Benutzte Events“+ Absatz „Frontend-Integration“+ Absatz „Erweiterbarkeit“+ Absatz „Unterstützung durch die Modulwerft“
06.02.2015	1.1.0	<ul style="list-style-type: none">• Aktualisierung des Modul-Aufbaus• Aktualisierung der Datenbank-Struktur• Aktualisierung des Absatzes „Benutzte Events“• Neuer Absatz „Cronjob“

Allgemeiner Modul-Aufbau

Das Modul integriert seine Funktionalität vollständig über Events und einen Cronjob in das Magento-Grundsystem. Auf Rewrites von Core-Klassen wurde zugunsten der Kompatibilität und Updatefähigkeit verzichtet. Die benutzten Events sowie der Cronjob werden in einem der nachfolgenden Absätze genauer beschrieben.

Vom Modul gespeicherte Daten werden in der Datenbank abgelegt. Logging-Informationen werden in der Datei `/var/log/Modulwerft_GiftCard.log` hinterlegt.

Das Modul ist ausführlich über das Magento-Backend konfigurierbar. Details hierzu finden Sie in der Konfigurationsanleitung.

Aufgabe des Moduls ist es, Gutscheine in Form von Produkten verkaufbar zu machen und die Abwicklung vollständig zu automatisieren. Der Ablauf basiert auf einem dreistufigen System:

1. **Registrierung eines Gutscheinverkaufs**

Sobald ein Gutschein verkauft wird, wird dies in der Datenbank-Tabelle

modulwerft_giftcard_giftcard_ordered zusammen mit dem zu dem Zeitpunkt im Produkt konfigurierten Gutschein-Design vermerkt.

2. **Erstellung des Gutschein-Objektes**

Sobald eine Rechnung gespeichert wird, wird per Observer eingegriffen und geprüft, ob sich unter den verkauften und bezahlten Artikeln Gutscheine befinden. Dies wird anhand der zuvor registrierten Order-Item-ID erkannt. Die Rechnungspositionen (Invoice-Items) werden in der Datenbank-Tabelle **modulwerft_giftcard_giftcard** zur späteren Verarbeitung gespeichert. Diese arbeitet quasi als Warteschlange.

3. **Erstellung und ggf. automatischer Versand der Gutscheine**

Der asynchrone Cronjob prüft alle 15 Minuten, ob neue Gutscheine in der Warteschlange vorhanden sind. Dies passiert anhand der Spalten **rule_id** und **coupon_id** auf der Tabelle **modulwerft_giftcard_giftcard**. Für die neuen Elemente wird ein entsprechender Gutschein erzeugt und – je nach Konfiguration – per E-Mail verschickt.

Datenbank-Struktur

In der Magento-Datenbank werden insgesamt vier Tabellen angelegt. Alle Datenbank-Tabellen des Moduls tragen derzeit und zukünftig das Präfix **modulwerft_giftcard**.

modulwerft_giftcard_giftcard

In dieser Tabelle werden die noch nicht erstellten Gutscheine (`rule_id == null AND coupon_id == null`) sowie die bereits erstellten und verschickten (`coupon_id > 0 AND sent_at = <Zeitstempel`) Gutscheine

hinterlegt. Es existieren über Fremdschlüssel direkte und kaskadierende Verknüpfungen zu

- Rechnungen (invoice_id = sales_flat_invoice.entity_id)
- Rechnungspositionen (item_id = sales_flat_invoice_item.entity_id)
- Warenkorb-Preisregel (rule_id = salesrule.rule_id)
- Gutschein-Code (coupon_id = salesrule_coupon.coupon_id)
- Gutscheindesign (giftcard_design_id=modulwerft_giftcard_giftcard_design.giftcard_design_id)

Sobald der Gutschein einmal gedruckt wurde, werden die Felder **filename** und **mime_type** mit entsprechenden Werten gefüllt.

Die Zeitstempel sent_at und created_at markieren den genauen Zeitpunkt, wann ein Gutschein erzeugt bzw. der Datensatz in der Tabelle erstellt wurde.

modulwerft_giftcard_giftcard_design

In dieser Tabelle werden die im Modul angelegten Gutschein-Designs gespeichert. Jedes Design besteht aus einem Titel, einem Dateinamen, Gestaltungsinformationen (Element-Positionen, Schriftfarbe etc.). Auf diese Datensätze wird in der Produkt-Konfiguration in einem Attribut verwiesen.

modulwerft_giftcard_giftcard_history

In dieser Tabelle wird eine Historie zum Gutschein gespeichert. Darin folgende Zeitpunkte festgehalten:

- Gutschein gekauft („purchased“)
- Gutschein bezahlt („paid“)
- Gutschein-Grafik erstellt („printed“)
- Gutschein versendet mit Kopieempfänger („sent_with_copy“)
- Gutschein versendet ohne Kopieempfänger („sent_without_copy“)

modulwerft_giftcard_giftcard_ordered

In dieser Tabelle werden die verkauften Gutschein-Produkte in Bezug auf die Bestellung vorgemerkt.

Neue Events

Das Modul erstellt ein paar Events, die für die Anpassung und Erweiterung des Moduls genutzt werden können.

```
Mage::dispatchEvent('modulwerft_giftcard_design_edit_form', array('form' => $form));
```

Event:	modulwerft_giftcard_design_edit_form
Kontext:	Das Formular zur Bearbeitung eines Gutschein-Designs wurde aufgebaut.
Parameter:	form (Varien_Data_Form): Formular-Objekt des Gutschein-Designs

Benutzte Events

Event:	sales_model_service_quote_submit_success
Event-Typ:	Core-Event (Sales-Modul)
Observer-Klasse:	Modulwerft_GiftCard_Model_Observer
Observer-Methode:	checkForGiftCards
Aufgabe:	Vermerken eines verkauften Gutschein-Produktes (Stufe 1, s.o.)

Event:	sales_order_invoice_save_after
Event-Typ:	Core-Event (Sales-Modul)
Observer-Klasse:	Modulwerft_GiftCard_Model_Observer
Observer-Methode:	createGiftCards
Aufgabe:	Erstellung der Geschenkgutschein-Objekte, wenn das Produkt einer Rechnung hinzugefügt und als bezahlt markiert wurde. (Stufe 2, s.o.)

Cronjob

Die Erzeugung sowie der Versand der Gutschein-Codes per E-Mail erfolgt asynchron über einen Cronjob. Dieser läuft alle 15 Minuten und betrachtet nur bisher unbearbeitete Gutscheine aus der Warteschlange (Datenbank-Tabelle **modulwerft_giftcard_giftcard**).

Soll ein Gutschein per E-Mail in gedruckter Form verschickt werden, übernimmt der Cronjob auch das Drucken des Gutscheines. In diesem Fall werden automatisch die Einträge in die Log-Tabelle **modulwerft_giftcard_giftcard_history** geschrieben.

Frontend-Integration

Der Kunde erhält im Frontend eine direkte Rückmeldung nur über verschickte Benachrichtigungsmails, sofern dies entsprechend der Konfigurationsanleitung eingerichtet wurde. Eine weitere Integration, etwa im Kundenmenü oder im Checkout, findet derzeit nicht statt. Die Gutscheine können als normale Magento-Produkttypen angeboten werden.

Erweiterbarkeit

Das Modul lässt sich an den unterschiedlichsten Stellen erweitern. Wir empfehlen den Verzicht auf Class-

Rewrites, um das Modul updatefähig zu halten. Weitere Events können wir in weiteren Modul-Versionen auf Wunsch gerne integrieren.

Unterstützung durch die Modulwerft

Sollten Sie Fragen zur Erweiterbarkeit des Moduls haben, melden Sie sich gerne per E-Mail bei unserem Support unter support@modulwerft.com.

Vielen Dank, dass Sie sich für ein Modul der Modulwerft entschieden haben. Sie haben damit ein Qualitätsprodukt erworben. Sollten Sie einmal nicht zufrieden sein oder haben Ideen für Erweiterungen oder Verbesserungen, würden wir uns über eine E-Mail an support@modulwerft.com freuen.