

# Möbius Furniturus - Guide

## Änderungskontrolle

Version	Datum	Person	Kommentar
0.1	24.10.2013		Negotium 1.1 Zielgruppe definieren
0.2	01.10.2013	Andy Pollari, Janosch Rohdewald	Negotium 1.2
0.3	08.10.2013	Andy Pollari	Personas
0.9	13.01.2014	Andy Pollari, Janosch Rohdewald	Finalisierung
1.0	14.01.2014	Andy Pollari, Janosch Rohdewald	Veröffentlichung

# Inhaltsverzeichnis

[Änderungskontrolle](#)

[Inhaltsverzeichnis](#)

[About the project](#)

[Personas](#)

[Persona Albrecht](#)

[Persona Laura](#)

[Persona Matthias](#)

[Use Case](#)

[Use Case Diagramm](#)

[Detaillierter Use Case: Bestellung aufgeben](#)

[Installation Manual](#)

[Development Environment](#)

[Datenbank](#)

[Tabellen](#)

[ERM](#)

[Software Architektur](#)

[\\*](#)

[-](#)

[Mehrsprachigkeit](#)

[Form Validierung mit HTML 5, regex](#)

[Template Engine](#)

[Admin bereich](#)

[Bestellbestätigung](#)

[PLZ Webservice](#)

[AJAX Suche](#)

[ShoppingCart OO](#)

[Quellenangabe](#)

[Quellen von Bildern und Texten](#)

[Verwaltung für Administratoren](#)

[Kategorie löschen](#)

[Möbel löschen](#)

[Feature löschen](#)

## About the project

Unser Webshop bietet nur qualitativ hochwertige, preiswerte und ausgewählte Möbel. Kunden können sich auf diesem Webshop registrieren. Wenn sie eingeloggt sind, so können Sie die von uns ausgewählten Artikel wie bei einem herkömmlichen Onlineshop in den Warenkorb legen und anschliessend bei uns bestellen. Für die Zielgruppe haben wir vier Personas definiert.

### Personas

#### Persona Susanne

Susanne (38) ist eine interessierte und fortgeschrittene Computeranwenderin. Sie wohnt mit ihrem Ehemann in einer Eigentumswohnung und interessiert sich für Design-Möbel und Designklassiker. Sie kauft nur sehr selten Möbel aber sehr gezielt.

#### Persona Albrecht

Albrecht (56) ist Notar und schätzt es sehr, sich zuhause entspannen zu können. Er hat ein hohes Einkommen. Seine Wohnungseinrichtung hat er sich vor über 20 Jahren angeschafft und möchte nun frischen Wind in seine Wohnung bringen. Er sieht die Anschaffung von Möbel als langfristige Investition und ist daher auch bereit, mehr Geld für ein Möbelstück auszugeben. Der Preis spielt bei Albrecht in der Regel keine Rolle.

#### Persona Laura

Laura (23) will bald mit ihrem Freund zusammenziehen. Ihre Computeranwenderkenntnisse sind durchschnittlich, sie bestellt regelmässig Kleider auf Zalando. Für ihre neue gemeinsame Wohnung mit ihrem Freund möchte sie nicht in die IKEA gehen, sondern etwas Exklusiveres. Laura und ihr Freund arbeiten beide Vollzeit. Sie wollen ihre gemeinsame Wohnung grosszügig einrichten und können es sich auch leisten.

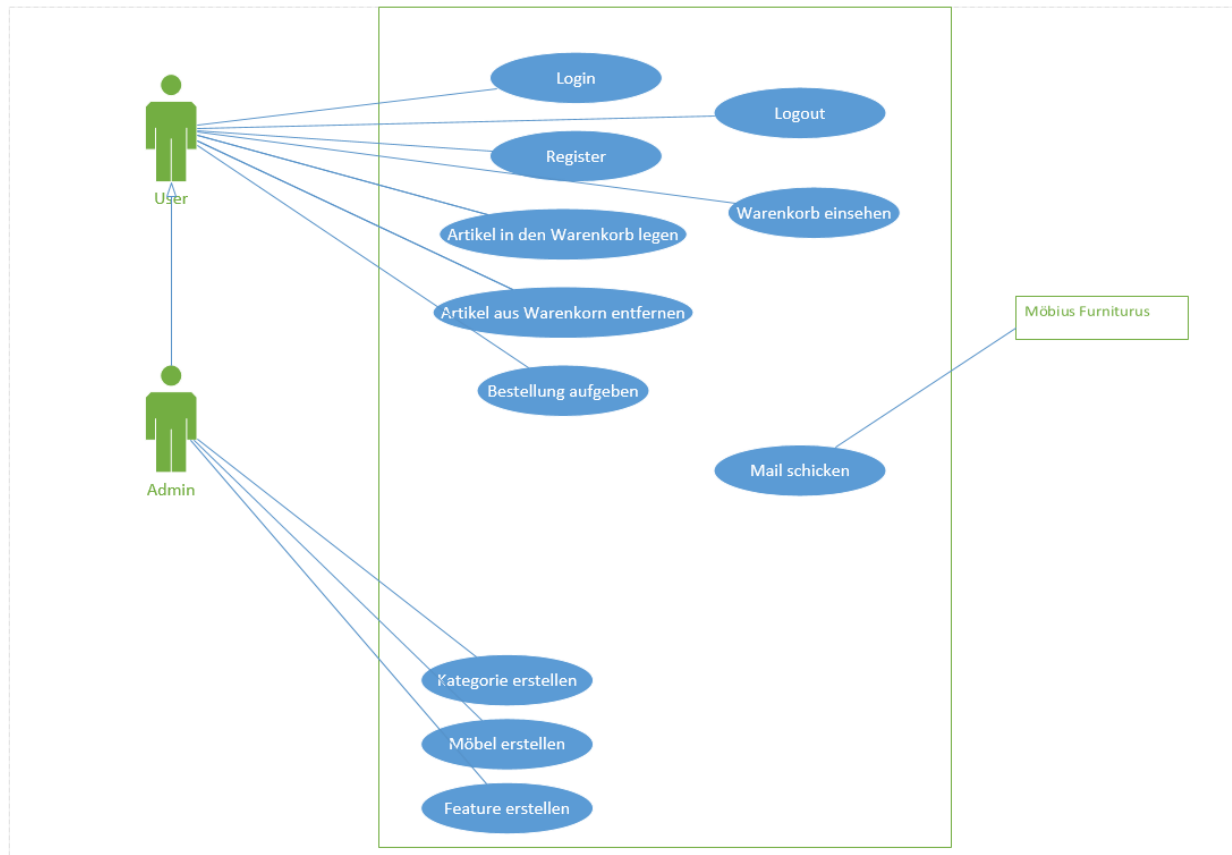
#### Persona Matthias

Matthias (28) wohnt seit über einem Jahr alleine. Einige Möbel konnte er von seiner vorherigen Wohngemeinschaft übernehmen. Die Möbel sind jedoch von schlechter Qualität und müssen demnächst ersetzt werden. Die neuen Möbel sollen qualitativ hochwertig sein aber auch das Gewisse etwas haben.

### Use Case

In diesem Kapitel sind die wichtigsten Anwendungsfälle ersichtlich. Einer davon wird genauer beschrieben.

## Use Case Diagramm



## Detaillierter Use Case: Bestellung aufgeben

Use Case Identifier	UC 3.0
Name	Möbel erstellen
Beschreibung	Der Administrator kann ein neues Möbel erfassen.
Beteiligte Akteure	User mit der Rolle Administrator
Status	Implementiert, der Administrator kann ein neuer Möbel erfassen.
Includes	Login (UC 1.0), der Admin muss dafür eingeloggt sein.
Auslöser	Ein Möbel das im Webshop neu angeboten werden soll, befindet sich noch nicht im Webshop. Es muss erfasst werden.
Vorbedingungen	Informationen Name, Beschreibung, Bild, Zusatzfeatures inkl. Preis, Kategorie muss für die Erfassung bekannt sein.
Invarianten	Andere Möbel dürfen während der Erfassung des Möbels nicht beeinflusst / verändert werden.
Nachbedingung	Das die Möbel wurde erstellt. Das heisst, es befindet sich in der Datenbank und ein Kunde kann es wählen, um es in den Warenkorb zu legen um es anschliessen zu bestellen.
Standardablauf	<ul style="list-style-type: none"><li>• Der Administrator logged sich auf der Seite ein.</li><li>• Im Footer befindet sich das Admin Menü, hier wählt er den Menüpunkt <i>Möbel erstellen</i>.</li><li>• Es erscheint eine Seite mit der Erstellmaske.</li><li>• Wenn alle erforderlichen Angaben eingegeben wurden, wird das neue Möbel mit dem Knopf <i>speichern</i> gespeichert</li><li>• Das Möbel befindet sich nun im Webshop.</li></ul>
Alternative Ablaufschritte	Nicht vorhanden
Hinweise	Der Benutzer muss die Rolle Administrator haben um das Adminmenü zu sehen.
Version	1.0, Andy Pollari

# Installation Manual

1. Code vom GitHub Repository auschecken:  
<https://github.com/polandy/ch.bfh.bti7054.w2013.q.furnitureShop.git>
2. Das Script *doc/database\_structure.sql* gegen die MySQL Datenbank ausführen, dies erstellt die DB Struktur.
3. Das Script *doc/database\_testData.sql* gegen die MySQL Datenbank ausführen, dies füllt die DB mit Testdaten.
4. In der Datei *php/Config.class.php* die folgenden Werte gegebenenfalls anpassen:
  - a. *database*
  - b. *database\_host*
  - c. *database\_user*
  - d. *database\_pw*
  - e. *smtp*
  - f. *smtp\_port*
  - g. *email*
  - h. *email\_pw*
5. Die Zeile *extension=php\_openssl.dll* in der Datei *php.ini* aktivieren, damit der Mailversand via SSL klappt.
6. Apache neu starten
7. Nun kann man die Website *WEBDIR/src* (Beispiel: <http://localhost/furnitureShop/src/>) aufrufen und einen neuen Benutzer erstellen oder sich als Administrator mit Benutzernamen *hero* und Passwort *huhu* einloggen.

## Development Environment

OS	Linux Mint 15 (Janosch) Windows 7 (Andy)
Code Repository	GitHub <a href="https://github.com/polandy/ch.bfh.bti7054.w2013.q.furnitureShop">https://github.com/polandy/ch.bfh.bti7054.w2013.q.furnitureShop</a>
Browser	Firefox 24.0, mit FireBug 1.12.3
IDE	IntelliJ IDEA 12 Plugins: PHP, PHP Advanced Autocomplete, Git Integration

Server Installation Bundle	XAMPP 1.8.2-2
	Apache 2.4.4 MySQL 5.5.32 PHP 5.4.19 phpMyAdmin 4.0.4
Frameworks	Foundation 4.3.2
	jQuery 1.10.2
	FPDF 1.7
	PHPMailer 5.1.0

# Datenbank

Die Datenbank von Möbius Furniturus ist bestehend aus acht Tabellen. Der Default-Name der Datenbank lautet *db\_furnitureshop*.

## Tabellen

### **category**

In dieser Tabelle befinden sich alle erfassten Möbelkategorien wie z.B. Stuhl, Tisch etc.

### **feature**

Ein Feature ist ein Möbel mit einem Zubehör.

Das Feature ist immer an ein furniture gebunden.

Beispiel für ein Feature:

Es existiert eine Lampe X mit dem Durchmesser von 30cm.

Nun kann man für dieses Möbel eine neue Grösse erfassen mit einem Preisaufschlag von 200.- CHF. Also erfasst man für diese Lampe X ein Feature:

- Beschreibung: Durchmesser 35cm
- Preiszuschlag: 200.- CHF

### **furniture**

Die Möbel befinden sich in der *furniture* Tabelle und haben einen Preis. Jedes Möbel gehört einer Kategorie an. Pro Möbel kann ein Bild in Form einer URL gespeichert werden.

Das Möbel hat folgende Attribute in mehreren Sprachen:

- Name
- Beschreibung

### **order**

In der Ordertabelle befinden sich die Bestellungen.

Jede Bestellung kann man eindeutig einem user zuordnen und kann den status isOpen = [true|false] haben. Jeder Benutzer kann nur eine offene Bestellung haben.

### **order\_furniture**

Die *order\_furniture* Tabelle ist die Mapping-Tabelle zwischen *Order* und *Furniture*. Hier ist gespeichert, welche Möbel und Features welcher Bestellung zuzuordnen sind.

### **paymentmethod**

In dieser Tabelle sind alle Zahlungsmöglichkeiten gespeichert, die dem Benutzer zur Verfügung stehen. Momentan stehen nur folgende Zahlungsmöglichkeiten zur Verfügung:

- Barzahlung bei Abholung
- Rechnung



## role

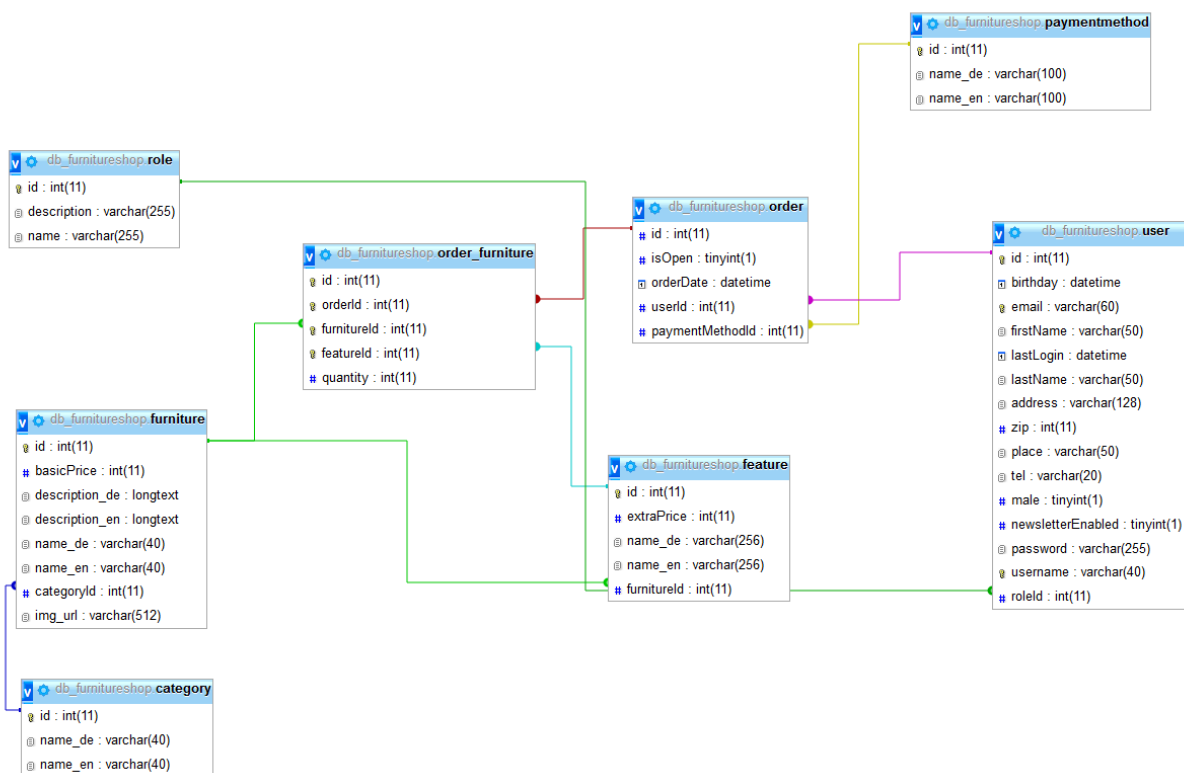
In der Tabelle *role* befinden sich alle Rollen, die den Usern zugeordnet werden können. Somit können verschiedene Benutzer auch verschiedene Berechtigungen haben. Für den Anfang wurden zwei Rollen implementiert: *default* und *admin*.

## user

In der *user* Tabelle befinden sich alle registrierten Benutzer. Hier sind benutzerspezifische Daten gespeichert, die auch für die Bestellung verwendet werden.

Jeder Benutzer hat eine Rolle.

## ERM



# Software Architektur

## MVC

Die Applikation wurde nach dem MVC Pattern aufgebaut. Dabei ist sie wie folgt gegliedert:

- In *php/controller* befinden sich die Controller für die View
- In *php/model* finden sich die einzelnen Entities, welche die Datenbank Tabellen repräsentieren.
- In *php/service* befinden sich die Services, welche für den Controller Teil, der nicht nur die Views betrifft, zuständig sind. Dies sind unter anderem alle SQL Queries, der Mailversand (*OrderService*), die PDF Generierung (*OrderService*).
- In *tmpl* befinden sich die Views, genauer gesagt der HTML Code

## Datenmodell

In Datenmodell von Möbius Furniturus gibt es zwei generische Klassen von welcher alle anderen Model Klassen erben.

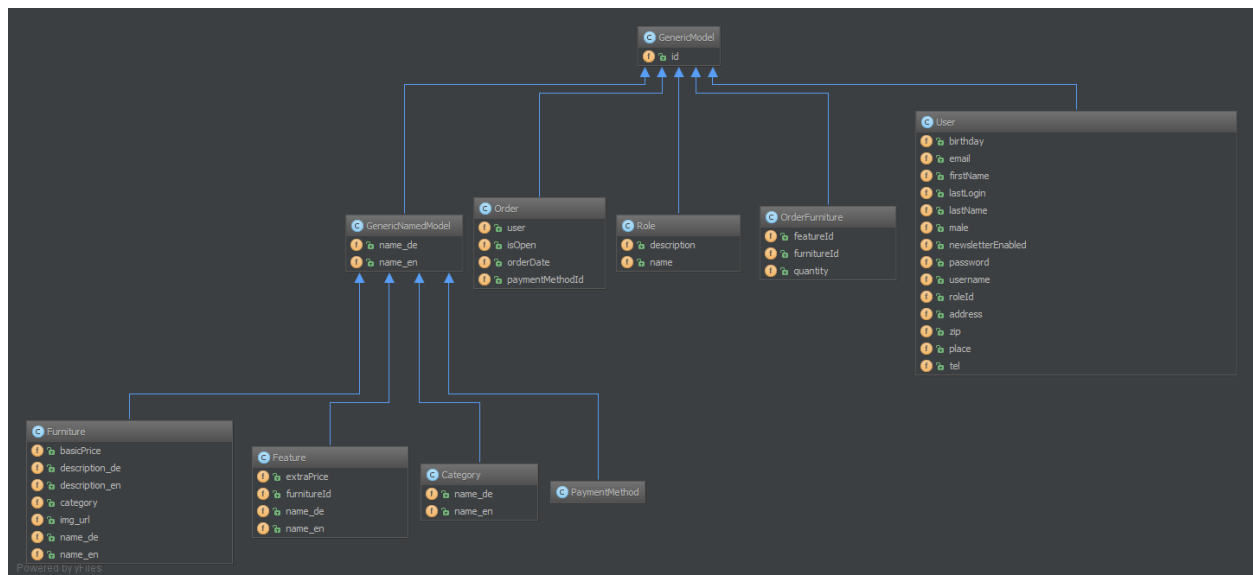
### GenericModel

Diese Klasse gibt den Primärschlüssel (*\$id*) vor.

### GenericNamedModel

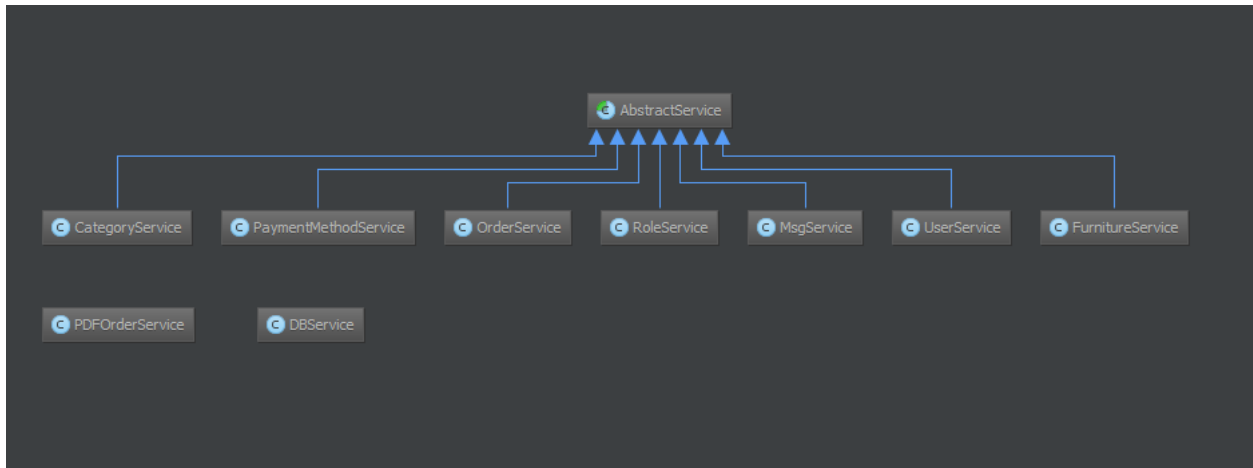
Diese Klasse gibt vor, dass das Objekt mehrsprachig. Pro Sprache gibt es ein Attribut, momentan sind das *\$name\_de*, *\$name\_en*.

Alle anderen Klassen repräsentieren Tabellen, des Datenbankmodells.



## Services

Unsere DB Services sind alles Singletons und erben von einem abstrakten Service, der die DB Verbindung zur Verfügung stellt.



## Mehrsprachigkeit

Da die Schweiz Kunden mit verschiedenen Muttersprachen bietet, wurde auf eine Mehrsprachigkeit Wert gelegt. Momentan werden Englisch und Deutsch unterstützt. Die Sprache kann im Footer gewechselt werden.

In der Datei `'php/Messages.class.php'` sind die Übersetzungen im Array `'$messages'` gespeichert. Mittels dem Service `'MsgService'` und der Methode `'getMsg()'` kann auf die einzelnen Texte zugegriffen werden. Werden in der Datenbank Übersetzungen benötigt, gibt es in der entsprechenden Tabelle pro Sprache ein Attribut (z.b. `'name_de'` und `'name_en'`).

## Bestellprozess

Um einen oder mehrere Artikel bestellen zu können muss wie folgt vorgegangen werden:

- Registrieren/Einloggen
- Einen oder mehrere Artikel oder Features mittels *'In Warenkorb'* hinzufügen.
- Den Warenkorb eventuell nochmals bearbeiten (Anzahl der Artikel ändern/Artikel entfernen)
- Im Warenkorb auf *'Bestellen'* klicken
- Auf der nächsten Seite die Zahlungsart auswählen und mittels *'Jetzt bestellen'* bestellen. Dabei kommt noch ein Javascript Popup zur finalen Bestätigung.
- Danach wird ein Email an den Kunden und an die Email Adresse des Webshops mit der Bestellübersicht in einem PDF gesendet.

## Aufbau der Seite

Die Seite ist so aufgebaut, dass immer die Datei `'index.php'` aufgerufen wird. Diese included dann die verschiedenen Bereiche der Webseite, welche die folgenden sind:

- Header

- Navigation
- Sidebar
- Content Area
- Footer

Der Content Bereich wird über das angeben der *'pageId'* gesteuert. Diese wird dann über die Konfiguration *'pageIds'* in der Datei *'config.class.php'* aufgelöst. Wird beispielsweise die URL *'index.php?pageId=99'* geladen, wird die Login Seite dargestellt. Dies passiert über die eigene Template Engine (siehe Features) wofür die Datei *'php/controller/login.php'* ausgeführt und die das Design der Datei *'tmpl/login.tpl.php'* geladen wird.

## Features

### Form Validierung mit HTML 5, regex

Die Formulare sind mit HTML 5 input Feldern aufgebaut und werden dadurch bereits vor dem Senden an das Backend auf dem Client validiert. Je nach Formular wird im Backend nochmal eine Validierung mittels Regex durchgeführt und entsprechende Fehler dem Benutzer rückgemeldet.

### Template Engine

Da wir etwas lernen wollten haben wir selbst eine kleine Template Engine geschrieben und keine bereits existierende Engine genommen. Diese ist ganz einfach aufgebaut. Es muss im Ordner *'php/controller'* der entsprechende Controller und in *'tmp'* das entsprechende Template existieren. Danach kann man im Controller mit *'\$tpl->assign(\$key, \$value)'* Variablen der View zur Verfügung stellen. In der View kann man diese dann mittels *'<?=\$TPL["key"]?>'* ausgeben. Damit hat man die grösstmögliche Trennung von PHP und HTML Code, was der Übersichtlichkeit und Wartbarkeit grosse Dienste leistet.

### Admin bereich

Damit man die Seite administrieren kann, wurde ein Administratorbereich eingerichtet. Dieser ist verfügbar, wenn man eingeloggt ist und der User der Rolle 'admin' zugeordnet ist. Ein User kann momentan nur in der Datenbank zum Administrator mutiert werden. Dafür muss die *'roleId'* auf dem User auf 1 gesetzt werden.

Im Footer erscheinen dann die Links zum Erstellen einer Kategorie oder eines Artikels. Des Weiteren erscheint auf dem Artikel selbst ein 'Bearbeiten' Knopf über welchen ein Artikel und seine Extras bearbeitet werden können.

Momentan muss ein Artikel/Extra/eine Kategorie noch in der Datenbank gelöscht werden. Das GUI könnte entsprechend ausgebaut werden, damit dies auch von darüber gemacht werden kann.

## Beispiel einer Artikel Bearbeitungsseite:

### Ein Möbel bearbeiten

Kategorie	<input type="text" value="Betten"/>
Bild URL	<input type="text" value="http://www.fachkom.ch/dateizentrale/"/>
Deutsche Bezeichnung	<input type="text" value="Letto Bett"/>
Englische Bezeichnung	<input type="text" value="Letto Bed"/>
Preis	<input type="text" value="4500"/>
Deutsche Beschreibung	<input type="text" value="Aus Massivholz, in zehn verschiedenen Holzarten und aus stehend verleimtem Multiplex."/>
Englische Beschreibung	<input type="text" value="Made of vertically glued birch multiplex or of European solid wood. Oiled."/>

Speichern

### Verfügbare Features

Beschreibung DE	Beschreibung EN	Preiszuschlag	Speichern
<input type="text" value="Birkenholz"/>	<input type="text" value="Birch"/>	<input type="text" value="1500"/>	

### Feature hinzufügen

Beschreibung DE	Beschreibung EN	Preiszuschlag	Speichern
<input type="text"/>	<input type="text"/>	<input type="text"/>	

## Bestellbestätigung

Wenn der Benutzer die Bestellung im Javascript Popup bestätigt, wird ein PDF mit der Bestellübersicht generiert und an die Email Adresse des Benutzers gesendet.

Die Einstellungen des SMTP Servers und der SMTP Login Daten können in der Datei “*Config.class.php*” geändert werden.

Für den Mailversand wird die Library ‘*PHPMailer*’ verwendet. Für die PDF Generierung die Library ‘*FPDF*’.

### Beispiel Email:

Order confirmation



Posteingang x



**Moebius furniturus** <mobius.furniturus@gmail.com>

an mich ▾

Dear Hero Superman

Your order is confirmed. Please find the order overview in the attachment.

Best regards



Beispiel PDF:

Möbius

**Möbius Furniturus - Bestellbestätigung**

Möbius Furniturus, Inc.  
Furnitinitystreet 8  
3006 Berne  
Tel: 031 563 65 32  
Fax: 031 563 65 33

Hero Superman  
myStreet 1  
3512 Walkringen  
078 897 87 45

**Bestelldatum** 2014-01-13 18:30:51  
**Zahlungsart** Rechnung

Möbel	Extra	Anzahl	Preis
Letto Bett	-	1	4'500.00 CHF
Letto Bett	Birkenholz	1	6'000.00 CHF
Akari	-	4	8'192.00 CHF

**Total 18'692.00 CHF**



## PLZ Webservice

Damit sichergestellt ist, dass die PLZ, die der Benutzer existiert wird der REST Webservice von geonames.org verwendet. Bei der Registration muss der Benutzer nur eine PLZ angeben. Anschliessend wird mittels diesem Dienst überprüft ob die PLZ auch existiert und der Ort automatisch ausgelesen.

geonames.org liefert ein JSON String zurück, welcher in PHP einfach mittels '*json\_decode*' einzulesen ist.

## Profil aktualisieren

Ist man eingeloggt, kann man mit einem Klick auf den eigenen Namen im Header das eigene Profil aktualisiert werden. Username und Email Adresse stehen für die Änderung nicht zur Verfügung. Es ist nicht vorgesehen, dass diese Werte (durch den User selbst) geändert werden.

## AJAX Suche

Oben rechts gibt es ein Suchfeld, mittels welchem die Artikel durchsucht werden können. Dabei werden die Suchresultate laufend beim eintippen des Suchbegriffs mittels AJAX aktualisiert. Dafür wird das '*onkeyup*' Event in Javascript verwendet und mittels '*jQuery.post()*' die Suchresultate von der Datei '*php/controller/search.php*' geladen.

## ShoppingCart OO

Der Warenkorb ist mit der Klasse *OrderService.class.php* in *php/service* als Objekt aufgebaut.

## Quellenangabe

Die Bilder und Texte auf Möbius Furniturus sind aus verschiedenen Quellen und können urheberrechtlich geschützt sein.

Im Falle von einer Aufschaltung in die Produktion von Möbius Furniturus müssen alle Bilder und Texte auf ihre Urheberrechte geprüft und allenfalls ersetzt werden.

## Quellen von Bildern und Texten

<http://www.design-kiste.ch/Wohnen-und-Buero/Stuehle/Stuhl-Asia.html>

<http://www.einrichten-design.de>

<http://static.ambientedirect.com>

<http://www.nest.co.uk>

<http://www.connox.de/>

<http://www.usm.com/>

<http://cdn.shopify.com/>

<http://www.stylepark.com/>

<http://www.fachkom.ch/>

<http://www.feurerdesign.ch>

<https://www.teojakob.ch/>

<http://getgoods.co.uk/>

<http://www.voltexdesign.com>

<http://immagini.webmobili.it/>

## Titelbilder für Hauptseiten

<http://www.moebelegger.ch/marke/336-sprenger/balkenbet-das-bett.html>

[http://www.pfister.ch/medias/sys\\_master/h43/h3a/8809466658846/aufmass\\_lampen\\_leuchten\\_slider01.jpg](http://www.pfister.ch/medias/sys_master/h43/h3a/8809466658846/aufmass_lampen_leuchten_slider01.jpg)

[http://www.schsch.ch/uploads/pics/stuehle\\_gross.jpg](http://www.schsch.ch/uploads/pics/stuehle_gross.jpg)

[http://www.cairo.ch/out/pictures/generated/product/1/900\\_700\\_80/moormann\\_egon.jpg](http://www.cairo.ch/out/pictures/generated/product/1/900_700_80/moormann_egon.jpg)

[http://www.honeyandspice.de/de/mainz\\_darmstadt\\_wiesbaden\\_frankfut\\_modernes\\_wohnen.html](http://www.honeyandspice.de/de/mainz_darmstadt_wiesbaden_frankfut_modernes_wohnen.html)

## Verwaltung für Administratoren

Dieses Kapitel bietet Unterstützung für Administrator User von Möbius Furniturus. Einige Funktionen werden über die Benutzeroberfläche nicht angeboten, können aber direkt in der Datenbank mittels SQL ausgeführt werden.

### Kategorie löschen

Eine Kategorie kann gelöscht werden, sofern kein Produkt für die Kategorie existiert. Wenn ein Produkt existiert, so muss sofern möglich dieses gelöscht werden.

SQL Script um eine Kategorie zu löschen:

```
DELETE FROM `category` WHERE id = $id
```

Der Wert \$id muss ersetzt werden mit der ID der Kategorie.

### Möbel löschen

Ein Möbel kann gelöscht werden, sofern es noch nicht von einem Kunden bestellt wurde.

Es darf auch noch kein Feature für das Möbel existieren, wenn das Möbel gelöscht werden soll.

SQL Script um ein Möbel zu löschen:

```
DELETE FROM `furniture` WHERE id = $id
```

Der Wert \$id muss ersetzt werden mit der ID des Möbels.

### Feature löschen

Wenn versehentlich ein Feature erstellt wurde, so muss dies über die Datenbank erfolgen. Das Feature darf noch nicht in einer Bestellung von einem Kunden erfasst sein, damit das Löschen erfolgreich erfolgen kann.

SQL Script um ein Feature zu löschen:

```
DELETE FROM `feature` WHERE id = $id
```

Der Wert \$id muss ersetzt werden mit der ID des Features.