

Fachhochschule Oldenburg / Ostfriesland / Wilhelmshaven
Fachbereich Technik
Studiengang Informatik

Diplomarbeit

***Entwicklung einer mandantenfähigen, webbasierten,
datenbank-gestützten eCommerce-Applikation mit
dem Schwerpunkt „Systemadministration“***

Martin Amelsberg

Matr.-Nr. 2083918
eMail: Amel@AmelFin.de

Abgabedatum: 22. Januar 2005

Erstprüfer: Prof. Dr. Schiemann-Lillie
Zweitprüfer: MBA Michael Franz
Betreuung durch: EuroCommerce Consulting Limited, Großbritannien

Vorwort

Mit dieser Diplomarbeit schlieÙe ich mein Studium zum Diplom-Informatiker (FH) an der *Fachhochschule Ostfriesland Oldenburg Wilhelmshaven* ab.

Bereits sehr fröh erkannte ich, dass mich die Computertechnik faszinierte und so habe ich mir (wenn auch auf Umwegen) meinen Traum erfüllt.

In all den Jahren der Ausbildung haben mir viele Menschen geholfen, ohne die ich sicher nicht so weit gekommen wäre. Auf diesem Weg bedanke ich mich bei allen, die Einfluss auf meine Laufbahn zum Diplom-Informatiker genommen haben.

Insbesondere gilt mein Dank

- meinen Eltern, die immer Verständnis für mich aufbrachten und stets mit Rat und Tat zur Seite standen.
- Prof. Dr. Schiemann-Lillie für die Unterstützung während dieser Diplomarbeit und bei vielen anderen Projekten

Inhaltsverzeichnis I

	Seite
1	Einleitung 1
1.1	Kurzreferat 1
1.2	Aufgabenstellung..... 1
1.3	Randbedingungen im Überblick..... 2
2	Projektplanung 3
2.1	Anforderungen und Lösungen 3
2.2.1	Die Template Engine „SMARTY“ 3
2.2.2	Das Datenbank Management System „mysql“ 6
2.2.3	Die Skriptsprache „PHP“ 6
2.3	Datenbank: themenorientierte Tabellennamen 7
2.4	Datenbank: eine Produkt-Tabelle im Detail 9
3	Die Applikation 11
3.1	Benutzergruppen..... 11
3.2	Produkte 12
3.3	Benutzergruppe „Wiederverkäufer“..... 15
3.3.1	Hierarchie..... 15
3.3.2	Rechteverwaltung..... 17
3.3.3	Menü und Funktionen 21
3.3.4	Von der Registrierung zum Produktverkauf 28
3.4	Benutzergruppe „Administratoren“ 32
3.4.1	Menü und Funktionen 32
3.4.2	Statistiken 33
3.4.3	Protokolle / Log-Daten..... 34
3.5	Benutzergruppe „Kunden“ 36
3.5.1	Menü und Funktionen 36
4	Ausgewählter Anwendungsfall 39
4.1	Rechte-Änderung eines Wiederverkäufers..... 39

Inhaltsverzeichnis II

		Seite
5	Sicherheit und Datenschutz	54
5.1	Sicherheit.....	54
5.2	Datenschutz.....	57
6	Literaturübersicht und Quellen	59
7	Bilderverzeichnis	60
8	Abbildungsverzeichnis	61
9	Erklärung	62
10	Kontaktadresse	62
Anhang A	Beispiel eines Kontakt-Formulars	63
Anhang B	Beispiel eines VISA-Bestell-Formulars	64

1 Einleitung

1.1 Kurzreferat

Diese Diplomarbeit beschreibt die Planung und Erstellung einer komplexen eCommerce-Anwendung, die ich zusammen mit Daniel Finger als Diplomarbeit angefertigt habe.

Anhand von verständlichen Beispielen und ausführlichen Erklärungen stelle ich diese umfassende Applikation ebenso für Nichtinformatiker dar, so dass auch ohne detaillierte Hintergrundinformationen das Grundsystem verstanden werden kann.

1.2 Aufgabenstellung

Um die Verwaltung von Wiederverkäufer-Daten und Produkten zu vereinfachen, ist eine webbasierte und datenbank-gestützte Applikation zu erstellen.

Ein besonderer Wert wird auf die Individualität und Dynamik der Anwendung gelegt. Dies bedeutet, dass nicht nur die Leiter (Administratoren) des Systems, sondern auch die Wiederverkäufer (entsprechend der vergebenen Rechte) die Preise, Beschreibung und Präsentation der zu verkaufenden Produkte bestimmen können.

Mittels der Template Engine SMARTY¹ erfolgt die Trennung zwischen Programmierung und Darstellung, so dass ein Wiederverkäufer, der seine Produkt-Präsentation anpassen möchte, lediglich HTML-Kenntnisse besitzen muss. Im Hintergrund erledigen PHP-Skripte und eine mySQL-Datenbank die erforderlichen Arbeiten zur Organisation der vom Wiederverkäufer eingegebenen Daten.

¹ dient zur Verwaltung von Vorlagen; <http://smarty.php.net>

1.3 Randbedingungen im Überblick

Betriebssystem : Linux

Websserver : Apache Version 1.3.33 ⇨ <http://httpd.apache.org>

Skriptsprache : PHP Version 4.3.9 ⇨ <http://www.php.net>

Datenbank : mySQL 3.23.49 ⇨ <http://www.mysql.de>

Template Engine : SMARTY Version 2.6.5-dev ⇨ <http://smarty.php.net>

PDF Generator : PDFlib Version 6.0.0p1 ⇨ <http://www.pdflib.de>

2 Projektplanung

2.1 Anforderungen und Lösungen

Wie bereits in „1.2 Aufgabenstellung“ kurz beschrieben wurde, geht es in erster Linie um die Verwaltung von Wiederverkäufern-Daten. Die zu verkaufenden Produkte sollen mit dem System verwaltet und vertrieben werden.

Entsprechend der Wiederverkäufer-Hierarchie und der Rechteverwaltung (siehe dazu Abschnitt „3.3 Benutzergruppe Wiederverkäufer“) ist darauf zu achten, dass die Administration nicht nur von „oben“ erfolgen kann, sondern jedem Wiederverkäufer gesondert Möglichkeiten zur Verfügung stehen, seine Produkte darzubieten.

Es wird mit einem hohen Aufkommen von verschiedenen Daten (Angaben zu Benutzern, Rechnungen, Formularen usw.) gerechnet, so dass die Nutzung einer Datenbank unumgänglich ist. (Was nicht bedeuten soll, dass bei kleinen Projekten mit wenigen Daten keine Datenbank genutzt werden sollte.)

All dies ist webbasiert zu realisieren, um von jedem Ort mit einem Internetanschluss ohne Zusatzsoftware auf das System zugegriffen zu können.

2.2.1 Die Template Engine „SMARTY“

Als Herzstück der individuellen Präsentation gelten die HTML-Vorlagen, die von den Administratoren bzw. Programmierern als Grundgerüst vorgegeben und unterschiedlichst angepasst werden können.

Bereits bei der Erstellung dieser Rohversion wird auf eine fehlerfreie und optisch ansprechende Darstellung geachtet, so dass ein Wiederverkäufer im Idealfall nur minimale Änderungen vornehmen muss.

Die Realisierung geschieht durch SMARTY, einer seit 3 Jahren in der Entwicklung befindlichen OpenSource¹ Template Engine. Nicht nur diese lange Test- und Entwicklungsphase, sondern auch der Vertrieb unter einer Subdomain von <http://www.php.net> zeigt, dass es sich um ein ernsthaftes Projekt handelt, das (mehr oder weniger) ohne Bedenken eingesetzt werden kann.

Hier soll nun als kleines Beispiel die Template erklärt werden, die im Kundenbereich die Bestellungen auflistet:

```
{* File: 00 03 01, USER, "Bestellung(en) anzeigen" *}
<html>
  <head>
    <title>Bestellungen anzeigen</title>
    {literal}
    <style type="text/css">
      body { background: #9aa url({/literal}{$imagedir}btgrdk.gif{literal})
        bottom left fixed repeat-x;color:white; }
      #ueberschrift { font-size:28px;
        font-weight:bold;
        text-decoration:underline;
        width:100%;
        text-align:center; }
    </style>
  {/literal}
</head>
<body scroll=auto>
  <br>
  {if $counter eq 0}
    <div id=ueberschrift>Sie haben keine Bestellungen.</div>
  {else}
    {if $counter eq 1}
      <div id=ueberschrift>Bestellung anzeigen</div>
    {else}
      <div id=ueberschrift>Bestellungen anzeigen</div>
    {/if}
    {$data}
  {/if}
  <br>
  <center> <a href="main.php"> Zur&uuml;ck zum Hauptmen&uuml;</a> </center>
</body>
</html>
```

Abb. 2.2-1 Quellcode einer SMARTY-Template

¹ sowohl der Quellcode des Programms als auch das Programm selber stehen einem Anwender zur freien Verfügung

Zusätzlich zum üblichen HTML- und CSS-Code, auf den nicht näher eingegangen werden soll, ist durch SMARTY sowohl der Einsatz von Variablen und Kommentaren als auch die Nutzung von Funktionen (hier: Kontrollstruktur) möglich:

a) Kommentar

Für die Programmierung von komplexeren Vorlagen (bspw. eines Bestellformulars) ist es ratsam, Kommentare zur Übersicht zu nutzen.

Ein Kommentar wird zwischen `{*` und `*`} geschrieben und im Gegensatz zu HTML-Kommentaren nicht in der dargestellten HTML-Seite angezeigt.

b) Variablen

Wie bei jeder „normalen“ Programmiersprache stehen Variablen auch hier für veränderliche Inhalte. Diese können in einer Kontrollstruktur (*\$counter*) genutzt werden oder (wie im Fall von *\$data*) anzuzeigenden Content beinhalten.

c) Funktionen

Dem Benutzer von SMARTY stehen eine Reihe von einfachen, aber effektiven Funktionen zur Verfügung. Eine Kontrollstruktur dient dazu, auf ein Ereignis bzw. einen Zustand („*wenn...dann...*“) zu reagieren.

Im obigen Quellcode-Beispiel wird ermittelt, ob der Kunde keine, eine oder mehrere Bestellungen hat, die angezeigt werden sollen. Je nach Ereignis wird ein unterschiedlicher Text als Überschrift angezeigt.

Anmerkung: ein Screenshot der HTML-Seite „*Bestellungen anzeigen*“ wird im Abschnitt „*3.5.1 Menü und Funktionen*“ gezeigt.

2.2.2 Das Datenbank Management System „mySQL“

Jede professionelle Web-Applikation bedient sich heutzutage einer Datenbank, um die Ordnung der Daten zu sichern.

Da bereits in anderen Projekten sehr gute Erfahrungen mit mySQL gemacht wurden, ist auch bei dieser Anwendung mySQL als **Datenbank-Management-System** eingesetzt worden.

Obwohl mySQL ein OpenSource-DBMS ist, muss es den Vergleich mit kommerziellen Produkten wie ORACLE¹ nicht scheuen.

Im Gegenteil: mySQL besticht durch seine hohe Geschwindigkeit, läuft sehr stabil auf verschiedenen Plattformen, ist relativ einfach zu bedienen und kann mit jeder gängigen Programmiersprache genutzt werden.

Nähere Informationen zur Datenbank-Planung und den eingesetzten Tabellen können den Abschnitten „2.3 Datenbank: themenorientierte Tabellennamen“ und „2.4 Datenbank: eine Produkt-Tabelle im Detail“ entnommen werden.

2.2.3 Die Skriptsprache „PHP“

Ebenfalls ein OpenSource-Produkt ist PHP, eine speziell für die Webprogrammierung entwickelte Skript-Sprache, die in HTML eingebettet wird.

Eine Syntax, die bereits vielen (C-) Programmierern bekannt war und ein grandioser Funktionsumfang sorgten seit 1995 für eine rasante Verbreitung dieser Sprache, so dass mittlerweile bei etwa 17 Mio. Domains² PHP eingesetzt wird.

PHP unterstützt mehrere DBMS (wie bspw. mySQL, MS-SQL,...) und kann plattform-unabhängig (Windows, Linux,...) eingesetzt werden, sofern für die Plattform ein Web-Server (bspw. Apache) existiert.

¹ <http://www.oracle.com/lang/de/database/index.html>

² Stand: Juli 2004; Quelle: <http://www.php.net/usage.php>

2.3 Datenbank: themenorientierte Tabellennamen

Bei der Planung und Erstellung der Datenbank-Tabellen wurde nicht nur auf die Effektivität Wert gelegt, sondern auch auf einen logischen Aufbau und eine verständliche Namensgebung der Tabellen.

Durch die Vergabe eines Präfixes werden die Tabellen nach „Themen“ geordnet. Da Tabellen in phpMyAdmin¹ alphabetisch sortiert werden, vereinfacht das Präfix nicht nur die Suche nach einer Tabelle, sondern sorgt auch gleichzeitig für eine logische Gruppierung:

- a) Präfix „global_“
 - ⇒ beschreibt Tabellen, die unabhängig von einer Personen-
gruppe (Admin, Kunde, Wiederverkäufer,...), einem Produkt
und einer Vorlage sind
 - ⇒ *global_user, global_vat_country,...*

- b) Präfix „reseller_“
 - ⇒ fasst Tabellen zusammen, die im direkten Zusammenhang
mit den Wiederverkäufern stehen, also bspw. die Relation
zu den Kunden oder Log-Daten
 - ⇒ *reseller_clients, reseller_log_data,...*

- c) Präfix „product_“
 - ⇒ Tabellen, in denen Daten zu einem Produkt gespeichert sind
 - ⇒ um wiederum Produkte in der phpMyAdmin-Übersicht zu-
sammenzufassen, wird an das Präfix „product_“ der Name
des Produktes gehängt
 - ⇒ *product_visa_productinfos, product_visa_formdata,...*
product_maestro_productinfos, product_maestro_formdata,...

¹ http://www.phpmyadmin.net/home_page

d) Präfix „templates_“

⇒ Abbildung von Tabellen, die mit den Vorlagen verknüpft sind

⇒ *templates_module_id, templates_supported_languages,...*

In vielen Projekten wurden bereits themenorientierte Datenbank-Tabellen genutzt und die Verwendung eines Präfixes hat sich auch in dieser Applikation als sehr effektiv und hilfreich erwiesen.

2.4 Datenbank: eine Produkttabelle im Detail

Im Abschnitt „3.3.3 Menü und Funktionen“ wird auf Tabellen-Daten eingegangen, um bspw. offene Rechnungen oder die Kunden eines Wiederverkäufers zu ermitteln. Um die Verknüpfungen verstehen zu können, wird hier eine Produkttabelle genauer betrachtet.

Die Daten zu einem Produkt werden in insgesamt 7 Tabellen gespeichert:

- product_*[Produktname]*_productinfos
 - ⇒ Wird ein neues Produkt im System gespeichert, gibt der Master-Master-Reseller¹ Vorgaben bezüglich Beschreibung, Preis und eMail-Text für eine Empfehlung vor.
Je nachdem, ob ein Reseller das Recht² dazu erhält, kann er die Vorgaben ändern.

- product_*[Produktname]*_reseller_formdata
 - ⇒ Bei einer Bestellung wird ein Formular vom Kunden ausgefüllt. Je nach Produkt sind die erforderlichen Angaben in einem Produkt verschieden, so dass diese Angaben in der „..._formdata“-Tabelle individuell für jedes Produkt gespeichert werden.

- product_*[Produktname]*_reseller_invoices
 - ⇒ Nachdem ein Kunde das Produkt-Formular ausgefüllt hat, wird eine Rechnung als PDF generiert und dem Kunden zugeschickt.
Um eine Rechnung auch nach dem Verlust der PDF rekonstruieren zu können, speichert das System nicht nur Daten, die im direkt Zusammenhang zu einer Rechnung stehen: um Redundanzen³ zu vermeiden, würde man lediglich die Kunden-ID speichern und diese ID mit den Personendaten aus der Tabelle „*global_user*“ verbinden.

¹ siehe dazu „3.3.1 Hierarchie“

² siehe dazu „3.3.2 Rechteverwaltung“

³ als „Redundanz“ bezeichnet man zusätzliche (und ggf. überflüssige) Informationen

Da Kunden aber die Möglichkeit haben, Ihre Daten zu ändern, wäre eine vollständige Rekonstruktion der Rechnung nicht möglich.

Gleiches gilt für den Verkäufer des Produktes, da auch er seine Personendaten ändern kann.

Deshalb werden Namen, Post- und eMail-Adressen usw. komplett gespeichert.

- product_*[Produktname]*_reseller_invoices_items
 - ⇒ Da ein Produkt ggf. nicht nur aus einer Einheit besteht, sondern Optionen anbietet, werden diese getrennt von der Rechnung gespeichert.

- product_*[Produktname]*_reseller_languages
 - ⇒ Ein Produkt sollte in verschiedenen Sprachen beworben werden können.
 - Jeder Reseller hat also die Möglichkeit, individuell für sich und separat für jedes seiner Produkte zu entscheiden, welche Sprachen bei der Bestellung unterstützt werden.

- product_*[Produktname]*_reseller_possible_countries
 - ⇒ Ähnlich wie bei den Sprachen kann der Wiederverkäufer bestimmen, aus welchen Länder ein Kunde bestellen darf, da es ggf. Produkte geben wird, die in einem bestimmten Land nicht verkauft werden sollen oder dürfen.
 - Somit wird das abgewählte Land nicht in der DropDown-Box bei der Kunden-Anmeldung angezeigt.

- product_*[Produktname]*_reseller_productinfos
 - ⇒ Wie bei „product_*[Produktname]*_productinfos“ beschrieben, kann ein Wiederverkäufer (je nach Recht) Produktbeschreibung, -preis und -empfehlungstext bestimmen, die in dieser Tabelle gespeichert werden.

3 Die Applikation

3.1 Benutzergruppen

Das System stellt für folgende Benutzergruppen Funktionen zur Verfügung:

a) Administratoren

⇒ Als „überwachendes Organ“ dienen die Administratoren. Im Menü eines Administrators finden sich grundsätzlich die gleichen Funktionen wie bei den Resellern mit dem Unterschied, dass hier keine Rechteverwaltung vorliegt. Ein Administrator wird also in seiner Handlung nicht eingeschränkt.

b) Wiederverkäufer / Verkäufer

⇒ Das System stellt den Wiederverkäufern Produkte bzw. die Präsentation derselben zur Verfügung und verbindet diese Wiederverkäufer somit unter ein System.

Als Grundlage dient eine Hierarchie und eine Rechteverwaltung, die in den Abschnitten 3.3.1 bzw. 3.3.2 näher betrachtet werden.

c) Kunden

⇒ Die Kunden können sowohl von den Administratoren als auch von den Wiederverkäufern verwaltet werden.

Entsprechend der Rechteverwaltung und Hierarchie ist ein Kunde nur für einen Wiederverkäufer verwaltbar, wenn der Kunde bei ihm bestellt hat oder der Wiederverkäufer die Rechnung stellt. Somit kann eine individuelle Betreuung stattfinden und die Kontaktperson für den Kunden bleibt gleich.

In den Abschnitten 3.3, 3.4 und 3.5 wird genauer auf die Benutzergruppen eingegangen.

3.2 Produkte

Ein wichtiger Bestandteil der Applikation sind die zu verkaufenden Produkte. Bei der Programmierung wurde ein großer Wert auf die Erweiterbarkeit, also die Möglichkeit, Produkte hinzuzufügen, gelegt.¹

Während der Diplomarbeit wurden u. a. folgende Produkte eingebunden:

- VISA² Electron
 - ⇒ Als Zusatz („Debitkarte“) zur VISA-Kreditkarte ist das internationale Zahlungsprodukt „VISA Electron“ gedacht. Im Gegensatz zu einer Kreditkarte basiert eine Debitkarte auf einem Guthaben; ähnlich einer PrePaid-Karte bei Mobiltelefonen.

- Maestro³ Traveller
 - ⇒ Ähnlich der „VISA Electron“ von einem anderen Anbieter

Bei einer Erweiterung der Produktpalette wird für jedes Produkt individuell eine Vorlage der Formulare erstellt. Standardmäßig sind immer folgende HTML-Seiten (genauer: HTML-SMARTY-Formulare) erforderlich:

- Eingangs-Seite
 - ⇒ Nachdem der Kunde auf den Link zur Bestellung geklickt hat, erscheint eine Seite, die das Firmenlogo des Anbieters zeigt und Informationen wie Beschreibung und Preis zu dem Produkt liefert. Außerdem beinhaltet diese Seite auch eine „Tell-A-Friend“-Funktion, die es dem Kunden erlaubt, einen Bekannten über dieses Produkt zu informieren. Alle Elemente dieser Seite (und auch die Seite selber) können im Bereich des Wiederverkäufers durch leicht verständliche Funktionen angepasst werden.

¹ siehe dazu die Diplomarbeit von Daniel Finger: *Entwicklung einer mandantenfähigen, webbasierten, datenbank-gestützten eCommerce-Applikation mit dem Schwerpunkt „Externe Schnittstellen“*

² Mehr Informationen hierzu unter <http://www.visa.de>

³ Mehr Informationen hierzu unter <http://www.maestro.at>

Dabei sind das Firmenlogo, die Beschreibung, der Preis und der Text für die Produktempfehlung separat in einfachen Eingabemasken einzugeben.

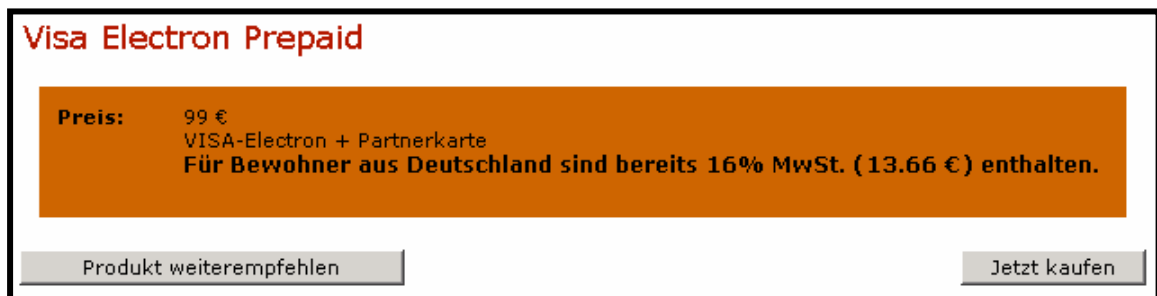


Bild 3.2-1 Beispiel für „Eingangsseite“

- Weiterempfehlungs-Seite
 - ⇒ Erlaubt es einem Kunden, das jeweilige Produkt einem Bekannten zu empfehlen.
 - Zusätzlich zu dem vom Wiederverkäufer definierten Werbetext kann vom Kunde eine persönliche Nachricht mitgeschickt werden.

- Bestell-Seite
 - ⇒ Über diese Seite wird der eigentliche Kauf geregelt und die Kontaktdaten (Name, Adresse,...) des Kunden entgegen-
genommen.
 - Hat dieser Kunde bereits ein Produkt bestellt und ist er somit dem System bekannt, kann er mit der Eingabe seiner UICI¹ und seinem Passwort den Bestellvorgang verkürzen, da seine Kontaktdaten aus der Datenbank geladen werden.
 - (siehe dazu „Anhang B: Beispiel eines Kontakt-Formulars“)

- Kontroll-Seite zur Bestellung
 - ⇒ Zeigt noch einmal alle Daten zur Kontrolle an, die bei der Bestellung vom Kunden eingegeben wurden.

¹ Universal Internet Company Identification

- Abschluss-Seite
 - ⇒ Bestätigt den Empfang der Bestellung und den Versand der Rechnung.
Im Hintergrund wird der Bestellprozess abgearbeitet und eine Rechnung generiert, der jeweilige Wiederverkäufer über die Bestellung per eMail benachrichtigt usw.

- Antrags-Formular
 - ⇒ Dieses Formular ist dem echten Antrags-Formular (im Normalfall eine PDF) nachempfunden. Hierbei werden erneut die Kontaktdaten des Kunden abgefragt und ggf. gewünschte Optionen, die zu dem Produkt gehören.
(siehe dazu „Anhang A: Beispiel eines VISA-Bestell-Formulars“)

- Kontroll-Seite zum Antrag
 - ⇒ Zeigt noch einmal alle Daten zur Kontrolle an, die vom Kunden beim Antrag eingegeben wurden.

- eMails
 - ⇒ Für einen Bestellprozess werden zahlreiche eMails verschickt:
 - + Begrüßung eines Neukunden
 - + Bestell-Bestätigung mit Rechnung
 - + Nachricht über den Zahlungseingang
 - + Produktempfehlung
 - + ...
 - Die Texte der eMails sind wiederum frei vom Wiederverkäufer definierbar und können für jedes Produkt angepasst werden.

3.3 Benutzergruppe „Wiederverkäufer“

3.3.1 Hierarchie

Die Anwendung verwaltet nicht nur die genannten Benutzergruppen, sondern unterscheidet bei den Wiederverkäufern auch zwischen unterschiedlichen Rängen.

Je nach Rang kann ein darüberliegender Verkäufer die Rechte und Produkte vergeben:

- Master-Master-Reseller
 - ⇒ Eine besondere Position bekleidet der Master-Master-Reseller, der in seiner Funktion nur wenig eingeschränkt ist. Im Gegensatz zu den anderen Reseller-Rängen kann sich dieser Reseller alle Kunden, Bestellungen, Rechnungen, Reseller-Daten usw. ansehen und übt so mehr eine kontrollierende und überwachende Funktion aus und beteiligt sich weniger am eigentlichen Verkaufsprozess. Im Normalfall legt er neue Master-Reseller-Accounts an und vergibt hierfür die Rechte und Produkte.
- Master-Reseller
 - ⇒ Einem Master-Reseller unterstehen Reseller und Empfehlungsgeber. Je nach Rechtevergabe bestimmt der Master-Reseller das Aussehen der Produkt-Präsentation (Vorlagen) sowie Produkt-Beschreibung und -Preis von allen Produkten, die er verwaltet.
- Reseller
 - ⇒ Üblicherweise bietet ein Reseller nur Produkte an und sorgt explizit für den Verkauf derselben. Auch beim Reseller kann über die Rechteverwaltung bestimmt werden, was ihm erlaubt wird.

- Empfehlungsgeber
 - ⇒ Ein Empfehlungsgeber ist das kleinste Glied in der Hierarchie und sorgt lediglich für den Verkauf von Produkten durch gezielte Werbung; bspw. auf der eigenen Homepage.
 - Dem Empfehlungsgeber unterstehen keine weiteren Wiederverkäufer.

Ein (grafisches) Beispiel:

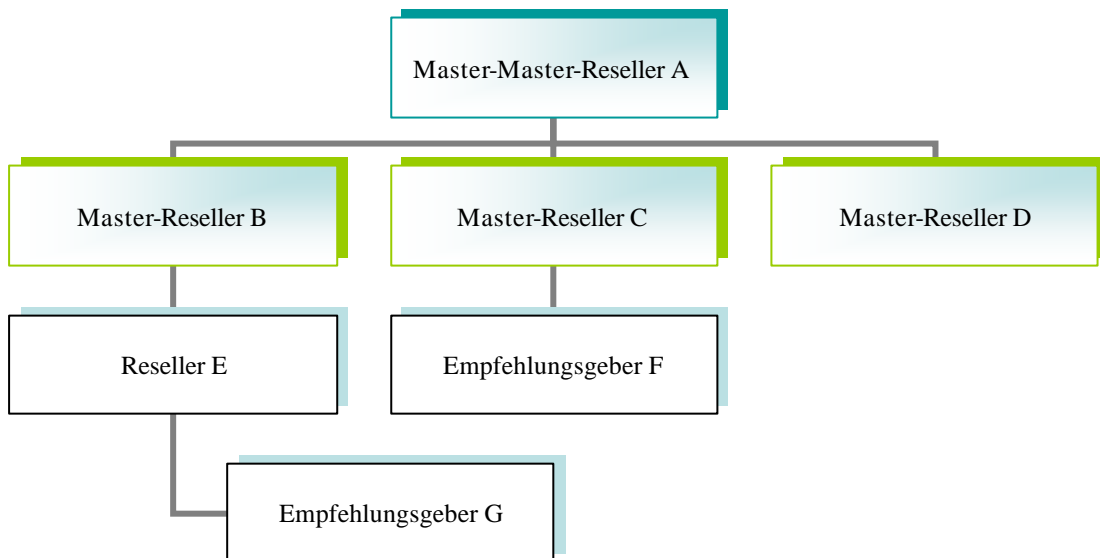


Bild 3.3-1 Grafische Abbildung einer Reseller-Hierarchie

Master-Reseller B kann die Daten der Kunden von Reseller E einsehen, jedoch nicht die Daten von Master-Reseller C bzw. Empfehlungsgeber F. Lediglich Master-Master-Reseller A kann alle Daten von B, C, D, E, F und G sehen; inklusive der Daten, die zu einer Bestellung oder einem Kunden gehören.

Master-Reseller B bestimmt darüber, welche Produkte Reseller E verkaufen darf und wie die Präsentation (Aufbau des Formulars, Firmenlogo,..) auszusehen hat. Entweder erlaubt er ihm keine Änderung, so dass alle relevanten Daten von Master-Reseller B übernommen werden oder er vergibt Rechte, die im nächsten Abschnitt beschrieben werden.

3.3.2 Rechteverwaltung

Je nach Kompetenz und Vertrauen können Master-Resellern und Resellern durch den überstehenden Wiederverkäufer Rechte zugewiesen oder vorenthalten werden.

Eine Änderung der Rechte ist jederzeit möglich, aufgrund der Systemkomplexität aber programmiertechnisch sehr umfassend und verzweigt.

Um dies zu verdeutlichen, wird im Abschnitt „4 Ausgewählter Anwendungsfall“ das Skript zur Rechte-Änderung genauer betrachtet.

Reseller-Rechte setzen/ändern

Rechte setzen von Hans Mustermann (UICI: mustermannmbh)

Recht	kein Recht	hat das Recht / keine Weitergabe	hat das Recht / Weitergabe erlaubt
Empfehlungsgeber anlegen	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
Reseller anlegen	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
Reseller nach Reseller verschieben	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
Kunden nach Reseller verschieben	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>

VISA Electron prepaid

Darf dieses Produkt anbieten	<input checked="" type="checkbox"/>		
Templates bearbeiten	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
Beschreibung/Preis ändern	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
Rechnungen von:	<input type="text" value="Hans Mustermann (ID: 8, UICI: mustermannmbh)"/>		

Maestro Debit Karte

Darf dieses Produkt anbieten	<input checked="" type="checkbox"/>		
Templates bearbeiten	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
Beschreibung/Preis ändern	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
Rechnungen von:	<input type="text" value="Hans Mustermann (ID: 8, UICI: mustermannmbh)"/>		

Bild 3.3-2 Screenshot „Rechte setzen/ändern“

Es wird unterschieden zwischen produktunabhängigen und produktabhängigen Rechten:

a) Produktunabhängige Rechte

- *Empfehlungsgeber anlegen*
 - ⇒ Dem Reseller wird es erlaubt (oder untersagt), selber Empfehlungsgeber anzulegen, die in der Hierarchie unter ihm platziert werden würden.

- *Reseller anlegen*
 - ⇒ Dieses Recht kann nur bei Master-Resellern vergeben werden, da ein Reseller keinen Reseller unter sich haben kann und demnach auch keinen Reseller anlegen darf. (Deshalb wird dieses Recht bei einem Reseller ausgeblendet.)

- *Reseller nach Reseller verschieben*
 - ⇒ Aus wirtschaftlichen oder organisatorischen Gründen kann es vorkommen, dass ein Reseller (und dessen Kunden) zu einem anderen Reseller verschoben werden müssen. (Auch dieser Punkt ist bei einem Reseller ausgeblendet, da kein Reseller gleichgestellte Reseller verschieben darf.)

- *Kunden nach Reseller verschieben*
 - ⇒ Ebenfalls ein Recht, das ein Reseller nicht besitzen darf, sondern lediglich ein Master-Reseller. Aus verständlichen Gründen sollte ein Reseller nicht seine Kunden an einen anderen Reseller geben können. Genauer betrachtet wäre dies auch unlogisch, da sein Umsatz ja durch diese Kunden entsteht. Es müssten also triftige Gründe für eine Verschiebung vorliegen, die lediglich ein (Master-) Master-Reseller beurteilen kann.

Eine weitere Abstufung findet zusätzlich bei der Rechtevergabe eines Master-Resellers statt: es kann bestimmt werden, ob er das Recht hat und auch (an einen Reseller) weitergeben darf oder ob er lediglich das Recht bekommt und ihm unterstehende Reseller das Recht keinesfalls (oder im Zweifelsfall durch den Master-Master-Reseller) beziehen.

Für Empfehlungsgeber kann keines dieser Rechte vergeben werden.

b) Produktabhängige Rechte

- *Darf dieses Produkt anbieten*

⇒ Legt fest, ob der Master-Reseller, Reseller oder Empfehlungsgeber ein bestimmtes Produkt präsentieren und verkaufen darf oder nicht.

Durch eine Checkbox wird der Zustand dieses Rechts festgelegt. Eine Änderung ist zu einem späteren Zeitpunkt möglich, so dass zugewiesene Produkte wieder entfernt bzw. nicht zugewiesene Produkte hinzugefügt werden können.

Wurde einem Wiederverkäufer der Verkauf von Produkt „X“ erlaubt, erscheint in der Auflistung das Icon .

- *Templates bearbeiten*

⇒ Die Vorlage ist für die Präsentation und den Bestellprozess eines Produktes relevant. Unter gewissen Umständen kann ein Master-Reseller wollen, dass „seine“ Reseller zwar ein Produkt X verkaufen dürfen, aber die HTML-Seiten mit dem Firmenlogo von ihm kommen.

- *Beschreibung/Preis ändern*

⇒ Zu einem Produkt gehören Beschreibung und Preis, die vom untergeordneten Wiederverkäufer verändert werden dürfen oder nicht.

- *Rechnung von*

⇒ Als Rechnungsteller stehen zur Auswahl

a) der Reseller selber

b) „sein“ Master-Reseller

c) der Master-Master-Reseller

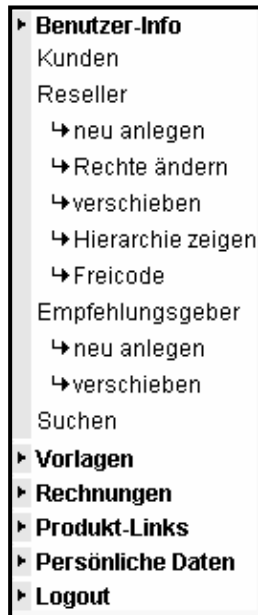
Somit kann nicht nur der Absender der Rechnung, sondern auch das eingebundene Rechnungslogo (das vom Rechnungsteller kommt) bestimmt werden.

Auch hier wird unterschieden zwischen dem eigentlichen Recht und dem Recht der Weitergabe an untergeordnete Wiederverkäufer.

3.3.3 Menü und Funktionen

Einem Wiederverkäufer stehen zahlreiche Funktionen zur Verfügung, die übersichtlich und nach Themen gruppiert in einem Menü dargestellt sind:

1. Menüpunkt „Benutzerinfo“



Anmerkung:

Je nach Recht kann die Anzahl der Menüpunkte variieren, da das Menü dynamisch zur Laufzeit generiert wird.

Menüpunkte, die der eingeloggte Benutzer nicht ausführen darf, werden im Menü nicht angezeigt.

Bild 3.3-3 Screenshot „Menü eines Resellers: Benutzer-Info“

- a) „Kunden“ zeigt die Kunden des jeweiligen Wiederverkäufers. Existieren auch Kunden bei einem untergeordneten Wiederverkäufer, werden diese ebenfalls aufgelistet.
- b) „Reseller“ erzeugt eine Übersicht aller Wiederverkäufer, die in der Hierarchie unterhalb des Wiederverkäufers stehen. Dieser Menüpunkt gilt nur für Master-Master-Reseller und Master-Reseller.
- c) „Reseller - neu anlegen“ erlaubt die Erstellung eines Benutzerkontos und die Zuweisung von Rechten für einen Master-Reseller bzw. Reseller.
- d) „Reseller - Rechte ändern“ ermöglicht die Änderung der Rechte.

- e) „Reseller - verschieben“ wird genutzt, um einen Reseller einem anderen Master-Reseller zuzuordnen.
- f) „Reseller - Hierarchie zeigen“ erzeugt eine Übersicht aller unterhalb des eingeloggten Wiederverkäufers stehenden Reseller und Empfehlungsgeber.
- g) „Reseller - Freicode“ erlaubt die Generierung von Freicodes, die zur Reseller-Registrierung genutzt werden können. Mit der Angabe eines Freicodes bei der Registrierung erhält ein Reseller keine Rechnung. Dies dient dazu, bereits bekannten Wiederverkäufern die Möglichkeit zu geben, kostenlos auf das neue System zu wechseln.
- h) „Empfehlungsgeber“ zeigt alle unterhalb des eingeloggten Wiederverkäufers stehenden Empfehlungsgeber an.
- i) „Empfehlungsgeber - neu anlegen“ erlaubt die Erstellung eines Benutzerkontos.
- j) „Empfehlungsgeber - verschieben“ gestattet die Verschiebung zu einem anderen Master-Reseller oder Reseller.

Die Auflistung von Resellern bzw. Empfehlungsgebern erzeugt eine Übersicht, die es dem Ausführer des Skriptes erlaubt, zusätzliche Informationen abzurufen. Als zusätzliches Steuerelement dient die Visualisierung der bereits im Menü genannten Funktionen durch verständliche Icons.






































Benutzer-Info: Reseller			
UICI	Name	Firma	Aktionen
berta1	Bäcker, Berta	Bäcker und Söhne	    
hans4711	Mustermann, Hans	Mustermann GmbH	    
holgerr	Richter, Holger	Richter und Sohn AG	    
klausmey	Meyer-Schmidt, Klaus	Schmidt AG	    
manni	Müller, Manfred	Müller Ltd.	    
sebkla	Klammer, Sebastian	Klammer GmbH	    

Bild 3.3-4 Screenshot „Auflistung der Reseller“

Die Icons haben folgende Bedeutung:

-  Anzeige von zusätzlichen Informationen, wie Adresse, Telefon, FAX, eMail-Adresse, Homepage, letzter Login, Registrier-Datum usw.
-  Zeigt alle Kunden des Wiederverkäufers an
-  Präsentiert die Hierarchie und markiert den jeweiligen Wiederverkäufer in gelber Schrift.
-  Ermöglicht die Änderung der Rechte des Wiederverkäufers (äquivalent zum Menüpunkt „Reseller – Rechte ändern“)
-  Ruft das Skript zum Verschieben eines Wiederverkäufers auf (äquivalent zum Menüpunkt „Reseller – verschieben“)
-  Sortiert die entsprechende Spalte absteigend
-  Sortiert die entsprechende Spalte aufsteigend

2. Menüpunkt „Vorlagen“



Bild 3.3-5 Screenshot „Menü eines Resellers: Vorlagen“

- a) „bearbeiten“ erlaubt (je nach Produkt und Recht) die Änderung von bestehenden Vorlagen, die zur Bestellung von Produkten genutzt wird. (siehe auch „Anhang A „Beispiel eines VISA-Bestell-Formulars“)

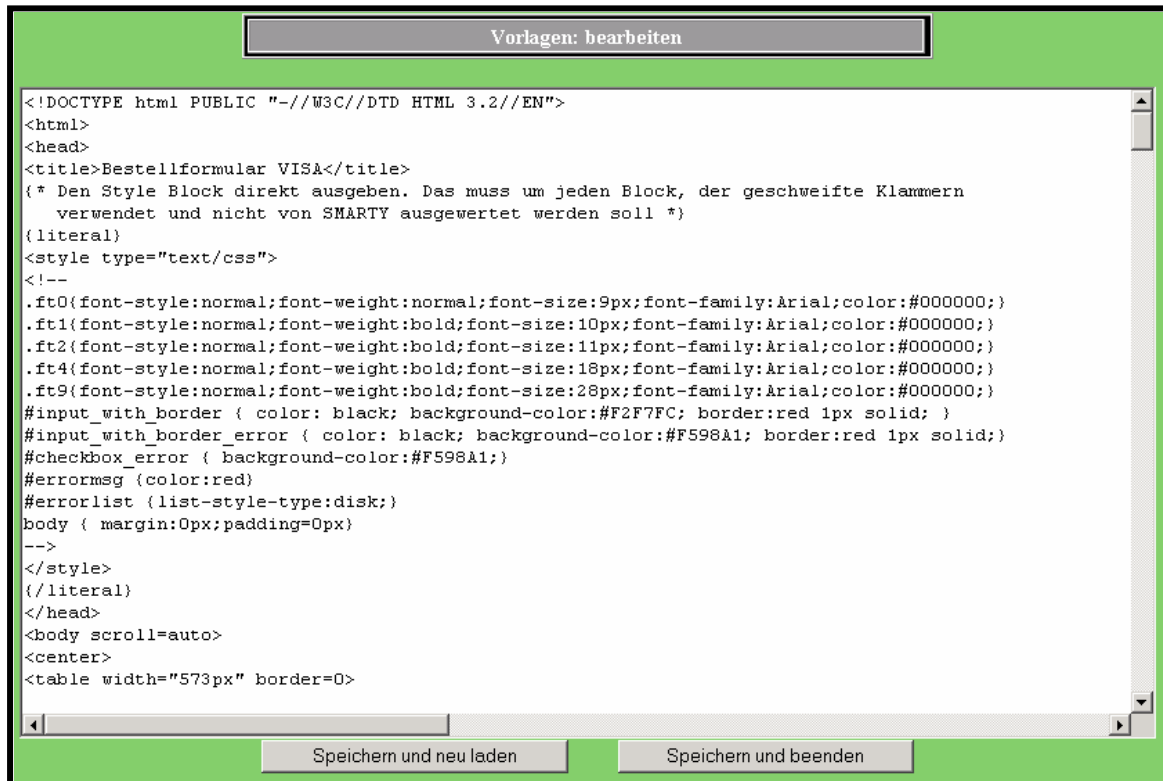


Bild 3.3-6 Screenshot „Bearbeitung einer Vorlage“

- b) „Firmenlogo“ startet ein Skript, das zum Upload von Bildern (dem Firmenlogo bzw. den Firmenlogos) dient. Jedem Produkt kann ein individuelles Firmenlogo zugewiesen werden, so dass für die Vorlage eines Bestell-Formulars ganz gezielt Zusatzinformationen o. ä. darstellbar sind.

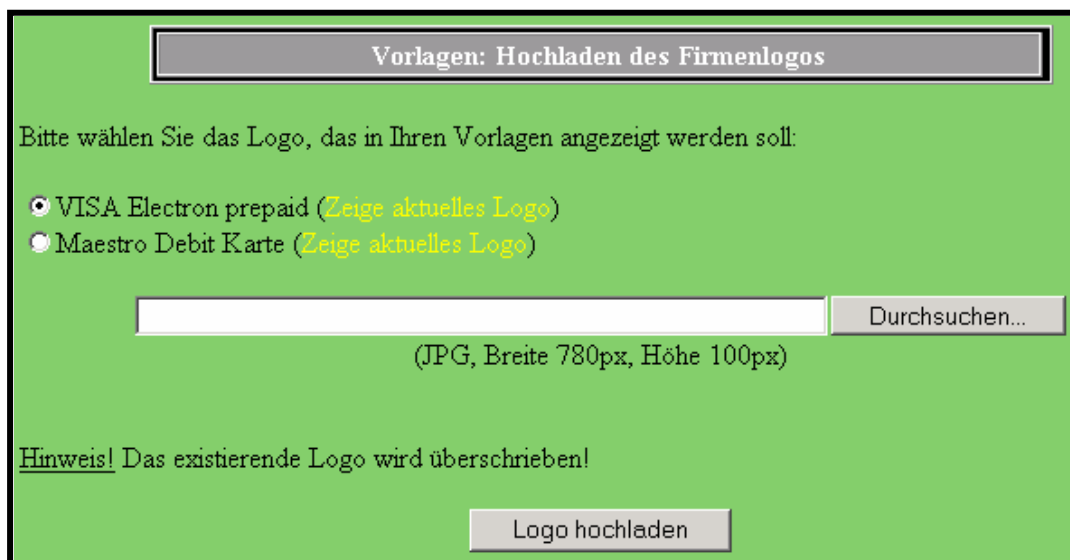


Bild 3.3-7 Screenshot „Firmenlogo hochladen“

- c) „*Sprachen*“ lässt dem Wiederverkäufer die Entscheidung, für welches Produkt er welche Sprache unterstützen will. Ist er bspw. nicht an englischsprachigen Kunden interessiert, fällt die Erstellung der englischsprachigen Vorlagen weg.
- d) „*Preis/Beschreibung*“ ermöglicht (wie immer entsprechend der zugewiesenen Rechte) die Änderung von Preis und Beschreibung für ein Produkt.

3. Menüpunkt „Rechnungen“



Bild 3.3-8 Screenshot „Menü eines Resellers: Rechnungen“

- a) „*bezahlt*“ zeigt alle bezahlten Rechnungen an.
- b) „*unbezahlt*“ zeigt alle unbezahlten Rechnungen an.

Auch bei dem Menüpunkt „*Rechnungen*“ greift der Zugriffsschutz: eine Rechnung wird nur angezeigt, wenn der Skript-Ausführer dazu berechtigt ist. Es muss im Datensatz der Rechnung als Rechnungsteller oder Verkäufer eingetragen sein.

4. Menüpunkt „Produkt-Links“



Bild 3.3-9 Screenshot „Menü eines Resellers: Produkt-Links“

a) „anzeigen“ dient nicht nur zur Anzeige, sondern steuert auch gleichzeitig die Generierung der Produkt-Links.

Einen Produkt-Link bringt der jeweilige Verkäufer auf seiner Homepage unter. Klickt ein Kunde auf diesen Link, wird die Bestellung bzw. der Kunde dem Wiederverkäufer zugeordnet. (siehe dazu auch 3.3.4 „Von der Registrierung bis zum Produktverkauf“)

Beispiel für einen Link:

<http://www.domain.de/annahme.php?id=7RWIYJI4T7E1SEBOT4RS>

In der Datenbank besteht eine eindeutige Relation von Produkt und Wiederverkäufer zu der Produkt-Link-ID „7RWIYJI4T7E1SEBOT4RS“.

Es wird die entsprechende Vorlage zur Produkt-Bestellung geladen und der Kunde kann das Formular ausfüllen.

5. Menüpunkt „Persönliche Daten“



Bild 3.3-10 Screenshot „Menü eines Resellers: Persönliche Daten“

a) „Benutzerdaten“ ermöglicht es dem Wiederverkäufer, seine Kontaktdaten zu bearbeiten. Diese sind lediglich für den „internen“ Zugriff (also für Administratoren und andere autorisierte Personen) gedacht.

Bereits in anderen Projekten wurde Wert darauf gelegt, dass für die Kundenbetreuung ein anderer Gesprächspartner eingetragen werden kann.

- b) „*Rechnungs-Daten*“ verwaltet die Konto- und Rechnungsdaten, die in der PDF-Rechnung dem Kunden mitgeteilt werden.
- c) „*Rechnungslogo*“ ist äquivalent zum Menüpunkt „*Firmenlogo*“ und dem damit verbundenen Skript. Hierbei wird das Rechnungslogo aber auf der PDF-Rechnung angezeigt und nicht in den Vorlagen genutzt.
- d) „*Bestellungen aus...*“ beschränkt die Länder, aus denen ein Produkt bestellt werden kann, in der Auswahl ein.



Bild 3.3-11 Screenshot „Ausschnitt aus einem Bestellformular“

- e) „*eMail-Daten*“ dient zur Eingabe/Verwaltung von eMail-Absender und eMail-Adressen, die bspw. zur Neukunden-Begrüßung oder zum Rechnungsversand genutzt werden.
- f) „*Passwort*“ startet ein Skript, mit dem der Wiederverkäufer sein Passwort ändern kann.
(siehe dazu auch „Kapitel 5 – Sicherheit und Datenschutz“)
- g) „*Sprache*“ ermöglicht die Wahl zwischen unterschiedlichen Sprachen (z. Z. deutsch und englisch), in denen der gesamte Wiederverkäufer-Bereich dargestellt wird.
Bereits zu Projektbeginn wurden Sprach-Dateien eingesetzt, so dass alle Text-Elemente zentral verwaltet und auf einfache Art und Weise an einen Übersetzer (zur Implementierung anderer Sprachen) übergeben werden können.

3.3.4 Von der Registrierung zum Produktverkauf

Bei der Realisierung des Weges von der Registrierung zum Produktverkauf wurde darauf geachtet, dass ein neuer Wiederverkäufer relativ schnell mit dem Verkauf von Produkten beginnen kann. Aus diesem Grunde gibt es nur wenige „Stationen“, die durchlaufen werden müssen:

⇨ Die Registrierung

Hier muss unterschieden werden zwischen der sogenannten „Selbst-Registrierung“, bei der der Impuls vom Interessenten selber ausgeht und einer Registrierung durch einen Administrator bzw. (Master-) Master-Reseller.

Bei der Selbst-Registrierung werden im Kontaktformular die Daten des neuen Verkäufers aufgenommen. Um sich vor Registrierungen zu schützen, die nicht ernst gemeint waren, ist vom Antragsteller eine Schutzgebühr (z. B. 10 Euro) zu zahlen. Da aber auch bereits bekannte¹ Wiederverkäufer diese Registrierung machen sollen, kann ein „Freicode“ eingegeben werden, der von der Zahlung der Gebühr befreit. Dieser kann vom Master-Reseller generiert und dem jeweiligen Wiederverkäufer vor seiner Registrierung zur Verfügung gestellt werden.

Wer keinen Freicode besitzt, erhält die Rechnung für die Schutzgebühr und hat in der Regel 14 Tage Zeit, die Rechnung zu begleichen. Der zuständige Master-Reseller sieht in seinem Bereich sofort nach der Registrierung die offene Rechnung und markiert diese als „bezahlt“, sobald ein Zahlungseingang verzeichnet werden konnte. Zusätzlich dazu vergibt er die Rechte/Produkte und schaltet den Antragsteller für den Login im Wiederverkäufer-Bereich frei.

Bei der Registrierung durch einen Administrator bzw. (Master-) Master-Reseller fällt die Rechnung, also die Zahlung der Gebühr, weg und der neue Verkäufer ist sofort nach der Vergabe der Rechte/Produkte freigeschaltet.

¹ diese sind dem (Master-) Master-Reseller bekannt und wechseln nur zum neuen System

⇒ Einrichtung des Benutzerkontos

Loggt sich der Wiederverkäufer im Reseller-Bereich ein, kann er sein Benutzerkonto sofort einrichten und (sofern das Recht dazu erteilt wurde) entsprechend seinen Vorstellungen gestalten.

In erster Linie gehört dazu die Präsentation der Produkte, die sich in der Gestaltung der Vorlagen und der Wahl des Firmenlogos widerspiegeln.

Der Kern für den Verkauf von Produkten sind die Produkt-Links, die er auf seiner Homepage nutzen muss, damit seine Kunden in das System geleitet werden. Um Anzeige-Fehlern vorzubeugen und einen reibungslosen Verkauf gewährleisten zu können, sind ggf. Daten erforderlich, die der Verkäufer unbedingt eingeben muss. Hierzu gehören Rechnungs-Daten (Bankverbindung, Adresse, Steuernummer,...) und Kontaktdaten (eMail-Absender und -Adresse). Erst nach der Eingabe dieser Informationen bekommt der Verkäufer die Produkt-Links zu sehen:

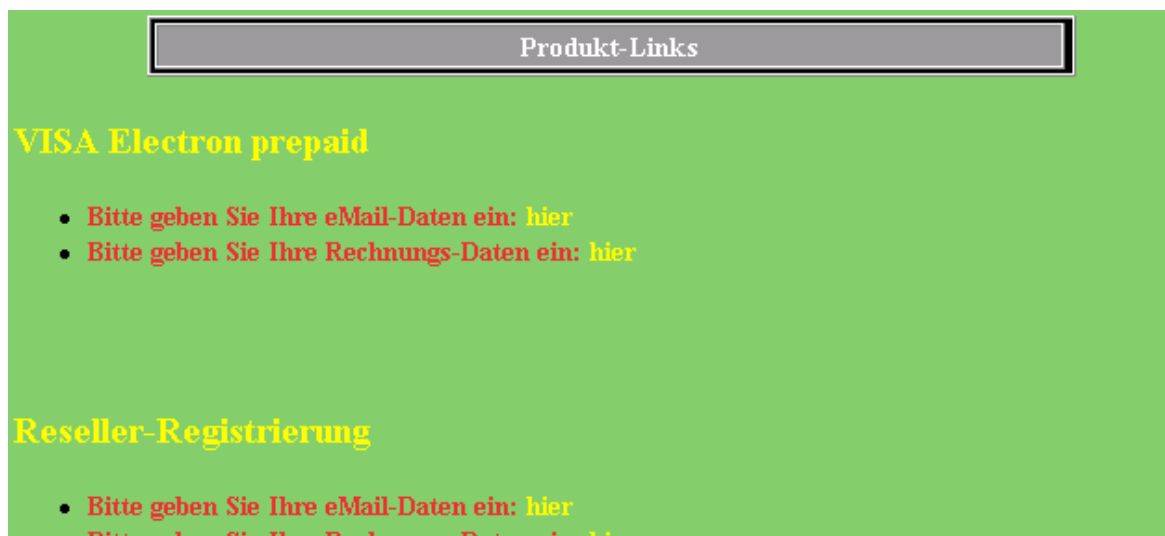


Bild 3.3-12 Screenshot „Produkt-Links“ (Fehler)

⇒ Produkt-Links

Sind alle erforderlichen Angaben gemacht worden, erhält der Wiederverkäufer seine (nur für ihn gültigen) Produkt-Links:

Produkt-Links

Um für das Produkt bzw. die Produkte zu werben, haben Sie 2 Möglichkeiten. Entweder verwenden Sie folgenden Link für alle Produkte und Sie verweisen auf die Homepage des Ihnen überstehenden Resellers.

<http://www.MasterDomain.de/BestellAnnahme.php?resid=7>

Oder Sie nutzen folgende produktabhängige Links:

VISA-Electron
http://www.MasterMasterDomain.de/get_product.php?product=2FCRVSOVS3MSUVJAW0NI

VISA-Electron + Partnerkarte
http://www.MasterMasterDomain.de/get_product.php?product=T3NTXST8W06U5S76J3FE

Bild 3.3-13 Screenshot „Produkt-Links“

Wie im Text in der Grafik geschrieben, hat der Wiederverkäufer zwei Möglichkeiten, einen Produkt-Link auf seiner Webseite zu setzen:

- a) Er nutzt nur seine ihm zugeteilte Reseller-ID (in der Grafik „resid=7“) und verweist auf die Homepage seines ihm übergeordneten Verkäufers.

Hierzu muss der übergeordnete Verkäufer wiederum auf seiner Homepage an den „resid“-Link seinen eigenen (langen) Produkt-Link hängen. Das System erkennt daran das Produkt und anhand von „resid“ die ID des Wiederverkäufers, von dem der Kauf ausging.

Es erfolgt die Darstellung der entsprechenden Produkt-Präsentation.

Dieser vom Ablauf etwas komplizierte Weg wird dem Verkäufer zur Verfügung gestellt, damit er es möglichst einfach hat. Er muss sich nicht um die (langen) Produkt-Links kümmern, sondern einfach nur seine ID übergeben, wie er es jetzt (bei dem ursprünglichen System) bereits macht.

- b) Er nutzt die produktabhängigen Links, die direkt auf das neue System verweisen und eine eindeutige Relation zwischen Produkt und Verkäufer darstellen.

3.4 Benutzergruppe „Administratoren“

3.4.1 Menü und Funktionen

Als administratives Organ des System stehen den Administratoren nicht nur die Möglichkeiten zur Verfügung, die ein Wiederverkäufer hat, sondern auch weiterführende Funktionen, die selbst einem Wiederverkäufer mit allen Rechten nicht zugänglich sind.

Im Gegensatz zu den Wiederverkäufern existiert für die Administratoren keine Hierarchie oder Rechteverwaltung. Somit wird in den Berechtigungen nicht zwischen Administrator A und Administrator B unterschieden und ihnen wird keine Funktion vorenthalten:

- ⇒ Wiederverkäufer sehen stets nur die Kunden, die bei ihnen bestellt haben bzw. bei denen sie als Rechnungsteller eingetragen sind.
Administratoren sehen grundsätzlich alle Kunden.

- ⇒ Wiederverkäufer können lediglich die Vorlagen, den Preis, die Beschreibung usw. der Produkte ändern, die sie auch verkaufen.
Administratoren verwalten alle Produkte und legen somit die Grundlage für alle Vorlagen der Verkäufer fest.
(Die von den Administratoren generierten Vorlagen dienen also dem Verkäufer als „Vorlage zur Vorlage“.)

- ⇒ Beschränkt durch seinen Rang sieht ein Wiederverkäufer nur einen Teil der Hierarchie und erhält somit keine Informationen über gleichgestellte Wiederverkäufer.
Für die Administratoren ist die gesamte Hierarchie der Wiederverkäufer (und somit auch deren Daten) zugänglich. Das beginnt beim Master-Master-Reseller und endet beim Empfehlungsgeber.

3.4.2 Statistiken

Um einen schnellen Überblick über das System und die Effektivität der Wiederverkäufer zu erhalten, können sich die Administratoren der implementierten Statistiken bedienen.

Durch entsprechende Datenbank-Abfragen und einer grafischen Komponente¹ können binnen Sekunden aussagekräftige Diagramme erzeugt werden.

Beispiel 1:

„Anzahl Kunden/Reseller“ stellt die jeweiligen Werte in einem Tortendiagramm dar:

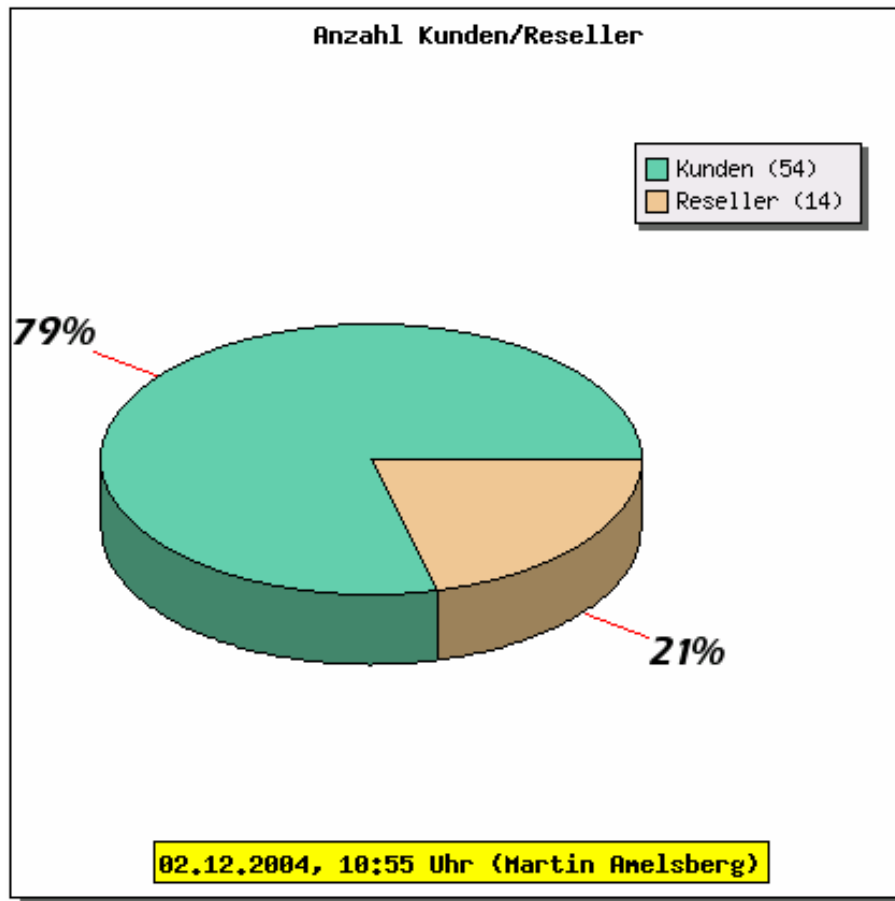


Bild 3.4-1 Beispiel: Statistik „Anzahl Kunden/Reseller“

¹ JGraph; eine objektorientierte Bibliothek für PHP; <http://www.aditus.nu/jgraph>

Beispiel 2:

„Registrierungen (der letzten 12 Monate)“ zeigt die Anzahl der Kundenregistrierungen der letzten 12 Monate; dynamisch abhängig vom Zeitpunkt der Statistikerstellung:

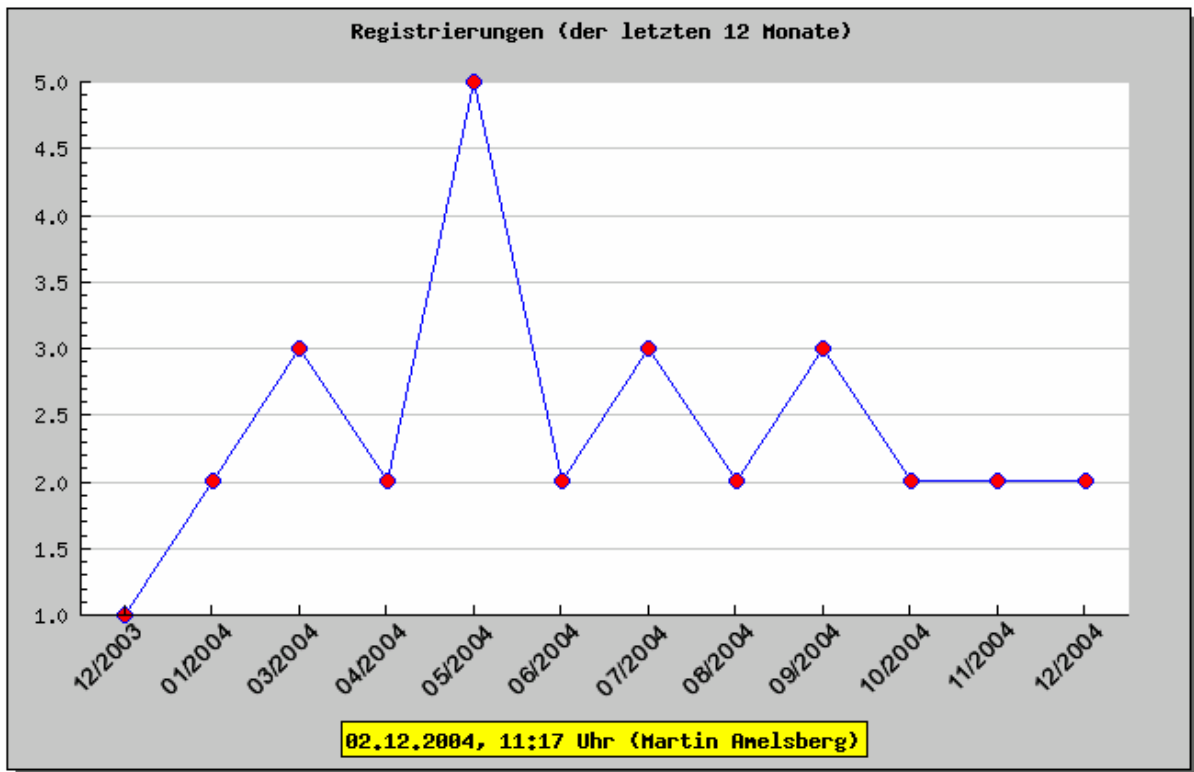


Bild 3.4-2 Beispiel: Statistik „Registrierungen der letzten 12 Monate“

3.4.3 Protokolle / Log-Daten

Alle wichtigen Aktionen eines Administrators, Wiederverkäufers und Kunden werden protokolliert. Diese Protokolle dienen im Zweifelsfall als Beweisgrundlage bei Streitigkeiten und zeigen den Administratoren auch indirekt die Auslastung der Applikation.

05.12.2004, 16:33 Uhr	Hans Mustermann beginnt mit dem Ausfüllen der VISA-Formulare.
05.12.2004, 16:34 Uhr	Hans Mustermann hat sich eingeloggt.
05.12.2004, 16:30 Uhr	Hans Mustermann hat sich ausgeloggt.
05.12.2004, 16:29 Uhr	Hans Mustermann hat sich eine Rechnung heruntergeladen.
05.12.2004, 16:29 Uhr	Hans Mustermann hat sich eingeloggt.
05.12.2004, 16:20 Uhr	Hans Mustermann hat eine Bestellung getätigt.
05.12.2004, 15:50 Uhr	Hans Mustermann hat sich angemeldet.

Bild 3.4-3 Beispiel: Kunden-Protokoll

Die Administratoren können sich das chronologisch sortierte Protokoll einer Benutzergruppe (Kunden, Wiederverkäufer und Administratoren) oder eines bestimmten Benutzers (Identifikation über die Eingabe der UICI) anzeigen lassen.

Je nach Ereignis wurden Links unter bestimmte Text-Elemente gesetzt (hier: „Hans Mustermann“), um den Administrator bspw. direkt auf die Seite der Benutzerinformationen zu führen.

3.5 Benutzergruppe „Kunden“

3.5.1 Menü und Funktionen

Im einfach gehaltenen und leicht verständlichen Hauptmenü hat der Kunde die Möglichkeit für folgende Aktionen:

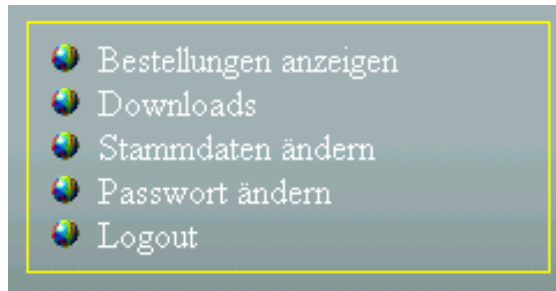


Bild 3.5-1 Screenshot „Kunden-Hauptmenü“

⇒ *Bestellungen anzeigen*

Listet alle Bestellungen mit dem jeweiligen Status auf:

<u>Bestellungen anzeigen</u>				
VISA Bestellung:				
ID	Datum	Preis	Status	Aktion
1000	19.11.2004	69.00 €	<input type="checkbox"/> <input checked="" type="checkbox"/> <input type="checkbox"/>	
Maestro Debit Bestellung:				
ID	Datum	Preis	Status	Aktion
1005	19.11.2004	59.00 €	<input type="checkbox"/> <input checked="" type="checkbox"/> <input checked="" type="checkbox"/>	
Miet-Handy Bestellung:				
ID	Datum	Preis	Status	Aktion
1002	19.11.2004	50.00 €	<input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/>	

Bild 3.5-2 Screenshot „Bestellungen anzeigen“

Der Status zeigt dem Kunden den Fortschritt der Bestellung. Es gibt 4 Zustände, die jeweils grafisch durch ein Ampelsymbol dargestellt sind:

a) rote Ampel

Es ist noch keine Zahlung vom Kunden eingegangen bzw. der Eingang wurde noch nicht bestätigt.

b) gelbe Ampel



Der Zahlungseingang wurde vom zuständigen Wiederverkäufer verzeichnet, aber für die Fortsetzung des Bestellprozesses (=Ausfüllen der Formulare) ist der Kunde gesperrt.

c) grün-gelbe Ampel

Die Zahlung vom Kunden ist gebucht und der Kunde darf seine Bestellung fortsetzen.

d) grüne Ampel

Die Bestellung ist abgeschlossen.

Über das -Symbol hat der Kunde Zugriff auf die jeweiligen PDF-Dokumente; das -Symbol dient zum Fortführen bzw. Abschließen einer offenen Bestellung.

⇒ *Downloads*

Erlaubt dem Kunden das Herunterladen von PDF-Dokumenten, wie Rechnungen, Formulare, usw.

⇒ *Stammdaten ändern*

Muss der Kunde aufgrund eines Umzugs o. ä. seine Kontaktdaten ändern, kann er diesen Menüpunkt nutzen. Ähnlich dem Registrierungs-Formular werden Namen, Adresse, Telefonnummer usw. abgefragt.

⇒ *Passwort ändern*

Ermöglicht es dem Kunden, das vom System vergebene Passwort zu ändern.

⇒ *Logout*

Mit diesem Menüpunkt verlässt der Kunde das System.

4 Ausgewählter Anwendungsfall

4.1 Rechte-Änderung eines Wiederverkäufers

Die nur scheinbar einfache, aber aufgrund der Systemkomplexität und -dynamik sehr umfangreiche Funktion der Rechte-Änderung eines Wiederverkäufers wird in diesem Abschnitt genauer betrachtet.

Da das Einbinden des kompletten 1000-Zeilen-Skriptes unsinnig wäre und den Rahmen dieser Betrachtung sprengen würde, wird nur ein Teil des Quellcodes veröffentlicht und besprochen. Hierbei handelt es sich um einen Auszug der Prüf- bzw. Schreibroutinen für produktabhängige Rechte, die einsetzen, wenn der Ausführer des Skriptes seine Änderungen bestätigt, also das Formular mit den (geänderten) Rechten abgeschickt hat.

Um die Beschreibung und somit die Lesbarkeit zu vereinfachen, soll hier folgende Hierarchie als Beispiel genommen werden. So kann größtenteils auf die abstrakten Bezeichnungen „übergeordneter Wiederverkäufer“ und „untergeordneter Wiederverkäufer“ verzichtet und anstelle dessen Namen verwendet werden:

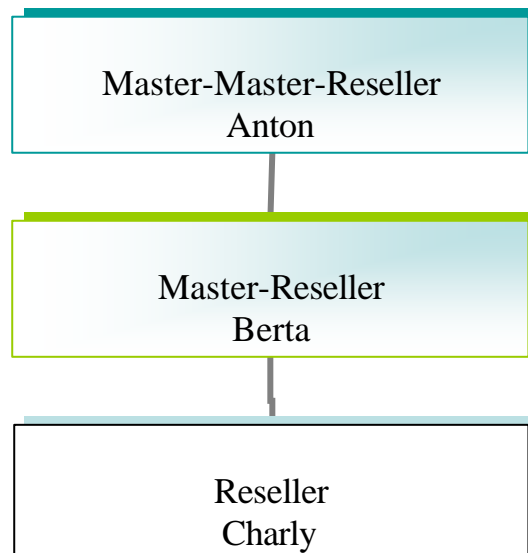


Bild 4.1-1 Abbildung einer Reseller-Hierarchie als Beispiel

Mittels einer *while*-Schleife werden Zweig 1 bzw. Zweig 2 für jedes Produkt in Abhängigkeit von *\$hasproduct* durchlaufen.

Hierbei wurden bereits vorher in *\$result_products_all* alle zur Verfügung stehenden Produkte aus der Datenbank geladen und in *\$hasproduct_[Produktname]* der Zustand der Checkbox „Darf dieses Produkt anbieten“ (siehe Bild 3.3-2) vom Formular übergeben.

```
while ($row = db_fetcharray($result_products_all))
{
    $varname = "hasproduct_" . $row['module_name'];
    $hasproduct = assignvar($varname, "");
    if ($hasproduct == "YES")
    {
        // Zweig 1
    }
    else
    {
        // Zweig 2
    }
}
```

Abb. 4.1-1 Quellcode-Auszug „Rechte-Änderung“ - Teil 1

Betrachtet werden soll zuerst „Zweig 1“, der durchlaufen wird, wenn die Checkbox aktiviert wurde, also der jeweilige Wiederverkäufer das Produkt verkaufen darf.

Eingangs tritt ein „Sicherheitsmechanismus“ in Kraft, der ein bereits entferntes Produkt berücksichtigt. Wurde einem Wiederverkäufer ein Produkt genommen, so wirkt sich das auch auf die untergeordneten Wiederverkäufer aus.

Beispiel: Anton hat Berta das Produkt „VISA“ weggenommen. Somit muss auch bei Charly das Produkt entfernt werden. Es würde eine Verletzung der Hierarchie vorliegen, wenn Charly weiterhin das Produkt verkaufen könnte, denn Berta hätte keine Möglichkeit mehr, dieses Produkt bei Charly zu administrieren, da sie es selber nicht mehr hat.

Aus diesem Grund muss geprüft werden, ob der übergeordnete Wiederverkäufer das Produkt besitzt und ggf. die Verarbeitung für das aktuelle Produkt abgebrochen werden muss.

Die Funktion „*get_boss*“ liefert die ID des überordneten Wiederverkäufers von *Sreseller_ID* zurück, also im Fall von Charly die ID von Berta.

```
// Pruefen, ob der ueberstehende Wiederverkaeuer
// dieses Produkt verkauft
$found_boss_product = FALSE;
if ($row['requires_invoice'] == "1")
{
    $bossID = get_boss($reseller_ID,$conn);
    if ($bossID != "")
    {
        $statusBOSS = get_status($bossID,
            $row['module_name'], "101", $conn);
        if ($statusBOSS > 0)
        {
            $missing_products[] = $row['module_name'];
            $found_boss_product = TRUE;
        }
    }
}
```

Abb. 4.1-2 Quellcode-Auszug „Rechte-Änderung“ – Teil 2

Wurde das Produkt beim übergeordnete Wiederverkäufer gefunden (die Variable *\$found_boss_product = TRUE*), kann die Prüfroutine fortgesetzt werden.

Es erfolgt eine Verzweigung entsprechend der Rechte, da jedes produktabhängige Recht eine andere Wirkung hat und so der Quellcode auch übersichtlicher wird.

In der ersten Verzweigung wird das Recht „100“, also das Bearbeiten der Vorlagen behandelt. Hierfür existiert in der Datenbank bei den Produkteigenschaften das Feld „*requires_templates*“, das darüber Auskunft gibt, ob für dieses Produkt überhaupt Vorlagen notwendig sind. (Normalerweise ist das für jedes Produkt der Fall, aber da ein besonderer Wert auf die Flexibilität des Systems gelegt wurde, werden vom System auch Produkte unterstützt, die nicht nach einem Standard-Verfahren behandelt werden.)

Um prüfen zu können, ob sich bei diesem Produkt/Recht etwas geändert hat, wird in *\$status_tmp* der aktuelle Zustand des Rechts gespeichert. Da in *\$priv100* der Zustand steht, der durch das Rechte-ändern-Formular übergeben wurde, kann mit „*if (\$status_tmp != \$priv_100)*“ diese Routine abgebrochen werden. Dies dient der Performance, da somit Prüfungen und Datenbank-Zugriffe gespart werden, wenn sowieso keine Änderung erfolgt ist.

Damit der Wiederverkäufer per eMail über eine Änderung informiert wird, speichert das Array *\$need_email[]* dessen ID und *\$emailforclick* erhält den Wert *TRUE*.

Da dieses Skript auch bei einer Wiederverkäufer-Neuanmeldung zum Einsatz kommt, entscheidet der Zustand von *\$new*, ob eine Benachrichtigung erforderlich ist oder nicht.

Trifft „*if (\$status_tmp == "0")*“ zu, so hat der Wiederverkäufer dieses Produkt im Moment nicht und die Produkt-Zuordnung wäre neu. Das Skript schreibt die erforderlichen Datenbank-Einträge und kopiert alle Vorlagen des übergeordneten Wiederverkäufers.

Handelt es sich nicht um ein neues Produkt, muss mit „*if (\$priv_100 == "0")*“ unterschieden werden, ob das Recht 100 nun genommen (*\$priv_100=0*) oder zugewiesen (*else*) wurde. Dementsprechend erfolgt eine DELETE- oder eine UPDATE-Anweisung der Datenbank-Daten.

Wie bereits eingangs erklärt, wirken sich Änderungen auch auf untergeordnete Wiederverkäufer aus. Wurde also bspw. Berta das Recht 100 vollständig (*\$priv_100=0*) bzw. das Recht der Weitergabe genommen (*\$priv_100=1*), muss bei Charly das Recht 100 ebenfalls gelöscht werden.

Eine Prüfung, welchen Zustand das Recht bei Charly hat, kann entfallen, da er bei keinem der möglichen Fälle das Recht besitzen darf.

Die gleiche Struktur der Prüf- und Schreibroutinen ist auch im eingebetteten Abschnitt „Unter-Unter-Wiederverkäufer“ zu finden, da sich die Rechte und die damit verbundenen Datenbank-Einträge für alle Wiederverkäufer in dem behandelten Hierarchie-Zweig ändern.

Natürlich muss auch bei den (unter-) untergeordneten Wiederverkäufern darauf geachtet werden, dass eine Benachrichtigung (*\$emailforclick* und *\$need_email*) erfolgt.

Anmerkung: es existieren für jeden Datenbank-Zugriff eigene Funktionen, die mit „db_[Funktionsname]“ bezeichnet werden und meist nur die von PHP zur Verfügung gestellten Datenbank-Funktionen beinhalten.

Der große Vorteil in den selbst-definierten Funktionen liegt darin, dass alle Datenbank-Dienste in einer separaten Datei „*functions_database.inc.php*“ erfasst sind und lediglich durch die Anpassung dieser Datei ein anderes **DatenBank-Management-System** unterstützt werden kann, da nur die PHP-eigenen Datenbank-Routinen ausgetauscht werden müssen.

Die Funktionsaufrufe bleiben somit auch bei einem wechselnden DBMS gleich und die Skripte müssen nicht verändert werden.

```
if ($found_boss_product == TRUE)
{
    // Recht 100 (Bearbeiten von Vorlagen)
    if ($row['requires_templates'] == "1")
    {
        // alte Einstellungen laden
        $status_tmp = get_status($reseller_ID,$row['module_name'],'100',$conn);

        // neue Einstellungen laden
        $varname = "setright_" . $row['module_name'];
        $priv_100 = assignvar($varname, "0");

        // pruefen, ob sich etwas geaendert hat
        if ($status_tmp != $priv_100)
        {
            // nur Benachrichtigung, wenn keine Neuanmeldung
            if ($new != "1")
            {
                $emailforclick = TRUE;
                $need_email[] = $reseller_ID;
            }
            /*
            if ($status_tmp == "0")
            {
                $sql_insert =
                "INSERT INTO reseller_rights
                (id, privilege, status, product)
                VALUES
                ('" . $reseller_ID . "', '100', '" . $priv_100 . "',
                '" . $row['module_name'] . "')";
                $result_insert = db_query($sql_insert, $conn);
                copyr(SMARTY_TEMPLATE_DIR . $bossID . "/" . $row['module_name'],
                SMARTY_TEMPLATE_DIR .
                $reseller_ID . "/" . $row['module_name']);
                copyr(IMAGE_LOGO_DIR . $bossID . "/logo_" .
                $row['module_name'] . ".jpg",
                IMAGE_LOGO_DIR . $reseller_ID . "/logo_" .
                $row['module_name'] . ".jpg");
            }
        }
    }
}
```

Abb. 4.1-3 Quellcode-Auszug „Rechte-Änderung“ – Teil 3

```

else
{
  if ($priv_100 == "0")
  {
    $sql_delete = "DELETE FROM reseller_rights
      WHERE (id = ' " . $reseller_ID . "')
      AND (privilege = '100')
      AND (product = ' " . $row['module_name'] . "')";
    $result_delete = db_query($sql_delete, $conn);
  }
  else
  {
    $sql_update = "UPDATE reseller_rights
      SET status = ' " . $priv_100 . "'
      WHERE (id = ' " . $reseller_ID . "')
      AND (privilege = '100')
      AND (product = ' " . $row['module_name'] . "')";
    $result_update = db_query($sql_update, $conn);
  }
  // wenn kein Recht -> (unter-) untergeordnete Wiederverkäufer berücksichtigen
  if (($priv_100 == "0") || ($priv_100 == "1"))
  {
    // Unter-Wiederverkäufer
    $reseller_under = get_reseller_under($reseller_ID,$conn);
    while ($row_under = db_fetcharray($reseller_under))
    {
      // Recht prüfen
      $status_under = get_status($row_under['reseller_under'],
        $row['module_name'], "100", $conn);
      if ($status_under != "0")
      {
        $sql_delete = "DELETE FROM reseller_rights
          WHERE (id = ' " . $row_under['reseller_under'] . "')
          AND (privilege = '100')
          AND (product = ' " . $row['module_name'] . "')";
        $result_delete = db_query($sql_delete, $conn);
        copyr(SMARTY_TEMPLATE_DIR . $reseller_ID . "/" . $row['module_name'],
          SMARTY_TEMPLATE_DIR .
            $row_under['reseller_under'] . "/" . $row['module_name']);
        copyr(IMAGE_LOGO_DIR . $reseller_ID . "/logo_ " .
          $row['module_name'] . ".jpg",
          IMAGE_LOGO_DIR .
            $row_under['reseller_under'] . "/logo_ " . $row['module_name'] . ".jpg");
        $need_email[] = $row_under['reseller_under'];
      }
      // Unter-Unter-Wiederverkäufer
      $reseller_under_under = get_reseller_under
        ($row_under['reseller_under'], $conn);
      while ($row_under_under = db_fetcharray($reseller_under_under))
      {
        $status_under_under = get_status($row_under_under['reseller_under'],
          $row['module_name'], "100", $conn);
        if ($status_under_under != "0")
        {
          $sql_delete = "DELETE FROM reseller_rights
            WHERE (id = ' " . $row_under_under['reseller_under'] . "')
            AND (privilege = '100')
            AND (product = ' " . $row['module_name'] . "')";
          $result_delete = db_query($sql_delete, $conn);
          copyr(SMARTY_TEMPLATE_DIR . $row_under['reseller_under'] .
            "/" . $row['module_name'],
            SMARTY_TEMPLATE_DIR . $row_under_under['reseller_under'] .
            "/" . $row['module_name']);
          copyr(IMAGE_LOGO_DIR . $row_under['reseller_under'] .
            "/logo_ " . $row['module_name'] . ".jpg",
            IMAGE_LOGO_DIR . $row_under_under['reseller_under'] .
            "/logo_ " . $row['module_name'] . ".jpg");
          $need_email[] = $row_under_under['reseller_under'];
        }
      }
    } // while
  }
}

```

Abb. 4.1-4 Quellcode-Auszug „Rechte-Änderung“ – Teil 4

Der nächste Abschnitt behandelt das Recht 101 (Rechnungsteller) und ist etwas übersichtlicher, da die Abfragen für die (unter-) untergeordneten Wiederverkäufer entfallen können.

Alle anderen Prüfroutinen sind denen ähnlich, die schon beim Recht 100 genutzt wurden.

```
// Recht 101 (Rechnungsteller)
if ($row['requires_invoice'] == "1")
{
    // alte Einstellungen laden
    $status_tmp = get_status($reseller_ID,$row['module_name'],"101",$conn);

    // neue Einstellungen laden
    $varname = "invoice_from_" . $row['module_name'];
    $priv_101 = assignvar($varname, "0");

    // pruefen, ob sich etwas geaendert hat
    if ($status_tmp != $priv_101)
    {
        if ($new != "1")
        {
            $emailforclick = TRUE;
            $need_email[] = $reseller_ID;
        }
        if ($priv_101 != $reseller_ID)
        {
            copyr(IMAGE_INVOICE_DIR . $priv_101 . "/logo_" .
                $row['module_name'] . ".jpg",
                IMAGE_INVOICE_DIR . $reseller_ID . "/logo_" .
                $row['module_name'] . ".jpg");
        }
        // pruefen, ob Recht bereits existierte
        if (($status_tmp == "0") || ($new == "1"))
        {
            $sql_insert = "INSERT INTO reseller_rights
                (id, privilege, status, product)
                VALUES
                ('" . $reseller_ID . "',
                '101',
                '" . $priv_101 . "',
                '" . $row['module_name'] . "')";
            $result_insert = db_query($sql_insert, $conn);
        }
        else
        {
            $sql_update = "UPDATE reseller_rights
                SET status = '" . $priv_101 . "'
                WHERE (id = '" . $reseller_ID . "')
                AND (privilege = '101')
                AND (product = '" . $row['module_name'] . "')";
            $result_update = db_query($sql_update, $conn);
        }
    }
}
```

Abb. 4.1-5 Quellcode-Auszug „Rechte-Änderung“ – Teil 5

Das letzte Recht 102 beschreibt die Befugnis, ob Preis und Beschreibung eines Produktes bearbeitet werden dürfen. Leicht zu erkennen sind die Ähnlichkeiten zur Programmier-Struktur von Recht 100.

```

// Recht 102 (Bearbeiten von Produkt-Preis und -Beschreibung)
if ($row['requires_ pricedescription '] == "1")
{
    // alte Einstellungen laden
    $status_tmp = get_status($reseller_ID,$row['module_name'], "102", $conn);
    // neue Einstellungen laden
    $varname = "changeproduct_" . $row['module_name'];
    $priv_102 = assignvar($varname, "0");
    // pruefen, ob sich etwas geaendert hat
    if ($status_tmp != $priv_102)
    {
        if ($new != "1")
        {
            $emailforclick = TRUE;
            $need_email[] = $reseller_ID;
        }
        // Preis/Beschreibung vom uebergeordneten Wiederverkaeuer laden
        $sql_productinfos = "SELECT * FROM
            product_" . $row['module_name'] . "_reseller_productinfos
            WHERE reseller_id = '" . $bossID . "'";
        $result_productinfos = db_query($sql_productinfos, $conn);

        if ($status_tmp == "0")
        {
            $sql_insert = "INSERT INTO reseller_rights
                (id, privilege, status, product)
                VALUES ('" . $reseller_ID . "', '102', '" . $priv_102 . "',
                    '" . $row['module_name'] . "')";
            $result_insert = db_query($sql_insert, $conn);
            // fuer alle Produkt-Beschreibungs-Elemente
            while ($row3 = db_fetcharray($result_productinfos))
            {
                $sql_insert = "INSERT INTO
                    product_" . $row['module_name'] . "_reseller_productinfos ";
                $sql_insert .= "(id, reseller_id, lang, description, price,
                    mail_recommend_subject, mail_recommend_body)";
                $sql_insert .= "VALUES ";
                $sql_insert .= "(" . $row3['id'] . ", ";
                $sql_insert .= "' . $reseller_ID . ", ";
                $sql_insert .= "' . $row3['lang'] . ", ";
                $sql_insert .= "' . $row3['description'] . ", ";
                $sql_insert .= "' . $row3['price'] . ", ";
                $sql_insert .= "' . $row3['mail_recommend_subject'] . ", ";
                $sql_insert .= "' . $row3['mail_recommend_body'] . ") ";
                $result_insert = db_query($sql_insert, $conn);
            }
        }
        else
        {
            if ($priv_102 == "0")
            {
                $sql_delete = "DELETE FROM reseller_rights
                    WHERE (id = '" . $reseller_ID . "')
                    AND (privilege = '102')
                    AND (product = '" . $row['module_name'] . "')";
                $result_delete = db_query($sql_delete, $conn);
            }
            else
            {
                $sql_update = "UPDATE reseller_rights
                    SET status = '" . $priv_102 . "'
                    WHERE (id = '" . $reseller_ID . "') AND (privilege = '102')
                    AND (product = '" . $row['module_name'] . "')";
                $result_update = db_query($sql_update, $conn);
            }
        }
    }
}

```

Abb. 4.1-6 Quellcode-Auszug „Rechte-Änderung“ – Teil 6

```

if (($priv_102 == "0") || ($priv_102 == "1"))
{
$reseller_under = get_reseller_under($reseller_ID,$conn);
while ($row_under = db_fetcharray($reseller_under))
{ // Recht pruefen
$status_under = get_status($row_under['reseller_under'],
$row['module_name'],'102',$conn);
if ($status_under != "0")
{
$sql_delete = "DELETE FROM reseller_rights
WHERE (id = ' " . $row_under['reseller_under'] . "')
AND (privilege = '102')
AND (product = ' " . $row['module_name'] . "')";
$result_delete = db_query($sql_delete, $conn);
while ($row3 = db_fetcharray($result_productinfos))
{
$sql_insert = "INSERT INTO product_ " .
$row['module_name'] . "_reseller_productinfos ";
$sql_insert .= "(id, reseller_id, lang, description, price
mail_recommend_subject, mail_recommend_body) ";
$sql_insert .= "VALUES ";
$sql_insert .= "(' " . $row3['id'] . " ', ";
$sql_insert .= "' " . $reseller_ID . " ', ";
$sql_insert .= "' " . $row3['lang'] . " ', ";
$sql_insert .= "' " . $row3['description'] . " ', ";
$sql_insert .= "' " . $row3['price'] . " ', ";
$sql_insert .= "' " . $row3['mail_recommend_subject'] . " ', ";
$sql_insert .= "' " . $row3['mail_recommend_body'] . " ') ";
$result_insert = db_query($sql_insert, $conn);
}
$need_email[] = $row_under['reseller_under'];
}
}
$reseller_under_under = get_reseller_under
($row_under['reseller_under'],$conn);
while ($row_under_under = db_fetcharray($reseller_under_under))
{
$status_under_under = get_status
($row_under_under['reseller_under'],
$row['module_name'],'102',$conn);
if ($status_under_under != "0")
{
$sql_delete = "DELETE FROM reseller_rights
WHERE
(id = ' " . $row_under_under['reseller_under'] . "')
AND (privilege = '102') AND (product = ' " . $row['module_name'] . "')";
$result_delete = db_query($sql_delete, $conn);
while ($row3 = db_fetcharray($result_productinfos))
{
$sql_insert = "INSERT INTO product_ " . $row['module_name'] .
"_reseller_productinfos ";
$sql_insert .= "(id, reseller_id, lang, description,price,
mail_recommend_subject, mail_recommend_body) ";
$sql_insert .= "VALUES (' " . $row3['id'] . " ', ";
$sql_insert .= "' " . $reseller_ID . " ', ";
$sql_insert .= "' " . $row3['lang'] . " ', ";
$sql_insert .= "' " . $row3['description'] . " ', ";
$sql_insert .= "' " . $row3['price'] . " ', ";
$sql_insert .= "' " . $row3['mail_recommend_subject'] . " ', ";
$sql_insert .= "' " . $row3['mail_recommend_body'] . " ') ";
$result_insert = db_query($sql_insert, $conn);
}
$need_email[] = $row_under['reseller_under'];
} // if ($status_under_under != "0")
} // while
} // while
} // if (($priv_102 == "0") || ($priv_102 == "1"))
} // else --> $status_tmp != "0"
} // if ($status_tmp != $priv_102)
} // if ($row['requires_pricedescription'] == "1")

```

Abb. 4.1-7 Quellcode-Auszug „Rechte-Änderung“ – Teil 7

Somit ist der Zweig 1 abgehandelt.

Es folgt die Beschreibung von Zweig 2, also dem *else*-Fall, der durchlaufen wird, wenn dem jeweiligen Wiederverkäufer ein Produkt entzogen wird.

Um die Hierarchie und System-Integrität nicht zu verletzen, müssen zuerst die untergeordneten Wiederverkäufer (bzw. deren Produkte) geprüft werden.

Die Hierarchie besagt, dass ein übergeordneter Wiederverkäufer immer die Rechte (Produkte) eines ihm unterstehenden Wiederverkäufers verwalten kann. Das bedeutet, dass bei der gezeigten Reseller-Hierarchie Berta die Produkt- und Rechte-Kontrolle über Charly hat, was wiederum impliziert, dass Charly kein Recht/Produkt haben darf, das Berta nicht besitzt/verkauft.

Wird Berta das Produkt X entzogen, muss auch Charly dieses Produkt genommen werden, da Berta sonst keine Kontrolle über ein Produkt von Charly hätte.

Außerdem ist es erforderlich, dass jeder Wiederverkäufer mindestens ein Produkt verkaufen muss. (Das Anlegen eines Benutzerkontos im System würde keinen Sinn machen, wenn von dem Benutzer kein Produkt verkauft wird.)

Beide Vorschriften gelten selbstverständlich auch für (unter-) untergeordnete Wiederverkäufer, also im Beispiel-Fall für Empfehlungsgeber, die „unter“ Charly eingetragen sind.

Aus diesen Gründen wird zuerst geprüft, ob ein untergeordneter Wiederverkäufer (hier: Charly) gerade dies Produkt verkauft, das dem übergeordnetem Wiederverkäufer (hier: Berta) genommen werden soll. Gleichzeitig erfolgt diese Prüfung für die unter-untergeordneten Wiederverkäufer.

Um den Ausführer des Skriptes zu unterstützen, werden die Benutzer-IDs der jeweiligen „Ein-Produkt-Verkäufer“ in das Array *\$reseller_ids_with_one_product* gespeichert und im Falle von *\$found == TRUE* mit einer Fehlermeldung angezeigt.

```

else
{
    // Produkt wird entzogen

    // pruefen, ob untergeordnete "Ein-Produkt-Verkaefer" existieren
    $found = FALSE;
    $reseller_under = get_reseller_under($reseller_ID,$conn);
    while ($row_under = db_fetcharray($reseller_under))
    {
        // alle Produkte des untergeordneten Wiederverkaeufers aus der DB holen
        $all_products_under = get_products_reseller_all($row_under['reseller_under'],$conn);
        $his_one_product = db_fetcharray($all_products_under);
        if ((db_numrows($all_products_under) == 1) &&
            ($his_one_product['product'] == $row['module_name']))
        {
            $reseller_IDs_with_one_product[] = array($row_under['reseller_under'],
                                                    $row['module_name']);

            $found = TRUE;
        }

        // Pruefen, ob unter-untergeordnete "Ein-Produkt-Verkaefer" existieren
        $reseller_under_under = get_reseller_under($row_under['reseller_under'],$conn);
        while ($row_under_under = db_fetcharray($reseller_under_under))
        {
            // alle Produkte des unter-untergeordneten Wiederverkaeufers aus der DB holen
            $all_products_under_under = get_products_reseller_all(
                $row_under_under['reseller_under'],$conn);
            $his_one_product = db_fetcharray($all_products_under_under);
            if ((db_numrows($all_products_under_under) == 1) &&
                ($his_one_product['product'] == $row['module_name']))
            {
                $reseller_IDs_with_one_product[] = array($row_under['reseller_under'],
                                                            $row['module_name']);

                $found = TRUE;
            }
        } // while
    } // while
} // while

```

Abb. 4.1-8 Quellcode-Auszug „Rechte-Änderung“ – Teil 8

Wird kein „Ein-Produkt-Verkäufer“ gefunden, bleibt *\$found* auf *FALSE* und der folgende Zweig wird durchlaufen.

Wird einem Wiederverkäufer ein Produkt genommen, darf dieses Produkt nicht einfach gelöscht werden, da über den Eintrag der Rechte und Produkte entschieden wird, welche Funktionen dem Wiederverkäufer zur Verfügung gestellt werden.

Hat Charly nicht das Recht, die Vorlagen des Produktes „VISA“ zu bearbeiten, erhält er auch den Menüpunkt dafür nicht bzw. bei der Auswahl der Produkt-Vorlagen wird „VISA“ nicht angezeigt.

Es gibt aber auch Operationen, bei denen nicht nur der aktuelle Stand der Rechte-Tabelle wichtig ist, sondern auch der Stand, wie er einmal war.

Ein Beispiel hierfür ist die Anzeige der bezahlten und unbezahlten Rechnungen: wird Charly das Produkt „VISA“ entzogen, so muss er trotzdem noch auf seine „alten“ VISA-Kunden und -Rechnungen zugreifen können.

Dies wäre aber nicht mehr möglich, wenn das Produkt „VISA“ achtlos aus seiner Rechte-Tabelle gelöscht wird.

Deshalb werden Produkte, die schon einmal zugewiesen, aber nun genommen wurden, mit einem negativen Status gespeichert. Somit bleibt zwar das Produkt bzw. der Eintrag dafür erhalten, aber durch den negativen Wert wird das Produkt als „inaktiv“ erkannt.

Diese Änderung erfolgt durch die einzeilige SQL-Anweisung UPDATE, bei dem „status“ auf „-status“ gesetzt wird.

Da das entzogene Produkt auch nicht mehr verkauft werden darf, wird der Produkt-Link gelöscht.

Danach wird (sofern vorhanden) auch den (unter-) untergeordneten Wiederverkäufern das Produkt durch die Negation entzogen und der Produkt-Link gelöscht.

Klickt ein Kunde auf einen in der Datenbank nicht mehr vorhandenen Produkt-Link, wird geprüft, ob der übergeordnete Wiederverkäufer dieses Produkt verkauft, also ob ein Produkt-Link existiert. (Dieser Mechanismus wurde realisiert, da der Verlust eines Kunden durch einen fehlenden Produkt-Link das Schlimmste ist, was passieren könnte.)

Darf aber auch der übergeordnete Wiederverkäufer das Produkt nicht verkaufen, erscheint eine Fehlermeldung. In diesem Fall ist das System „machtlos“.

```

if ($found == FALSE)
{
  if ($new != "1")
  {
    $emailforclick = TRUE;
  }
  // alte Einstellungen laden und pruefen, ob Produkt ueberhaupt vorhanden ist
  $status_tmp = get_status($reseller_ID,$row['module_name'],"101",$conn);
  if ($status_tmp > 0)
  {
    // Produkt entfernen (Produkt-Status negieren und Produkt-Link loeschen)
    $sql_update = "UPDATE reseller_rights SET status = -status
      WHERE (id = ' " . $reseller_ID . "')
      AND (product = ' " . $row['module_name'] . "')";
    $result_update = db_query($sql_update, $conn);

    $sql_delete = "DELETE FROM product_all_keys
      WHERE (reseller = ' " . $reseller_ID . "')
      AND (LEFT(template_id,2) = ' " . $row['template_prefix'] . "')";
    $result_delete = db_query($sql_delete, $conn);

    // untergeordnete Verkaeufel beruecksichtigen
    $reseller_under = get_reseller_under($reseller_ID,$conn);
    while ($row_under = db_fetcharray($reseller_under))
    {
      $status_under = get_status($row_under['reseller_under'],
        $row['module_name'], "101", $conn);
      if ($status_under > 0)
      {
        $sql_update = "UPDATE reseller_rights SET status = -status
          WHERE (id = ' " . $row_under['reseller_under'] . "')
          AND (product = ' " . $row['module_name'] . "')";
        $result_update = db_query($sql_update, $conn);

        $sql_delete = "DELETE FROM product_all_keys
          WHERE (reseller = ' " . $row_under['reseller_under'] . "')
          AND (LEFT(template_id,2) = ' " . $row['template_prefix'] . "')";
        $result_delete = db_query($sql_delete, $conn);
        $need_email[] = $row_under['reseller_under'];
      }
    }

    // unter-untergeordnete Verkaeufel beruecksichtigen
    $reseller_under_under = get_reseller_under($row_under['reseller_under'],$conn);
    while ($row_under_under = db_fetcharray($reseller_under_under))
    {
      $status_under_under = get_status($row_under_under['reseller_under'],
        $row['module_name'], "101", $conn);
      if ($status_under_under > 0)
      {
        $sql_update = "UPDATE reseller_rights SET status = -status
          WHERE (id = ' " . $row_under_under['reseller_under'] . "')
          AND (product = ' " . $row['module_name'] . "')";
        $result_update = db_query($sql_update, $conn);

        $sql_delete = "DELETE FROM product_all_keys
          WHERE (reseller = ' " . $row_under_under['reseller_under'] . "')
          AND (LEFT(template_id,2) = ' " . $row['template_prefix'] . "')";
        $result_delete = db_query($sql_delete, $conn);
        $need_email[] = $row_under_under['reseller_under'];
      } // if
    } // while
  } // while
} // if

```

Abb. 4.1-9 Quellcode-Auszug „Rechte-Änderung“ – Teil 9

Nun ist auch Zweig 2 abgearbeitet.

Ist in einem der Zweige ein Fehler aufgetreten, also verkauft ein übergeordneter Wiederverkäufer ein bestimmtes Produkt nicht (siehe Zweig 1) oder existieren „Ein-Produkt-Verkäufer“ (siehe Zweig 2), so wird dem Ausführer des Skriptes mitgeteilt, dass der Vorgang nicht vollständig abgeschlossen werden konnte. Er erhält die Information über den Grund und die Namen der Produkte und Wiederverkäufer.

Des Weiteren werden eMails bezüglich der Änderungen verschickt, um die jeweiligen Wiederverkäufer über die neue bzw. geänderte Produkt-Palette zu informieren. (Es kann zwar davon ausgegangen werden, dass das Entfernen oder Hinzufügen eines Produktes vorher mit dem Verkäufer abgeklärt wurde, aber trotzdem ist eine Benachrichtigung sinnvoll.)

5 Sicherheit und Datenschutz

Wie bei jedem Projekt, musste auch hier gewissenhaft mit den Themen „Sicherheit“ und „Datenschutz“ umgegangen werden.

5.1 Sicherheit

Aus verständlichen Gründen können nicht alle Sicherheitsmechanismen im Detail beschrieben werden, da gerade dies einem möglichen Angreifer interne Strukturen aufzeigen und ihm die Arbeit erleichtern würde.

Trotzdem beschreibt diese Diplomarbeit einige Schutzeinrichtungen, die jeder Anwendungs-Programmierer (insbesondere im Web-Bereich) grundsätzlich beachten sollte:

⇒ vom System generierte Passworte

Bei der Anmeldung eines neuen Benutzerkontos wird im System automatisch ein Passwort generiert.

Dieses Passwort genügt kryptografischen Ansprüchen und ist mit einer Länge von 8 Zeichen (bestehend aus Klein- und Großbuchstaben und Ziffern) als „sicher“ anzusehen.

⇒ vom Benutzer gewählte Passworte

Da davon ausgegangen werden kann, dass ein Benutzer das vom System generierte Passwort ändern will, muss auch bei seiner Eingabe auf ein Mindestmass an Sicherheit geachtet werden.

Es werden nur Passworte mit einer Länge von wenigstens 8 Zeichen (ebenfalls bestehend aus Klein- und Großbuchstaben und Ziffern) akzeptiert.

Passworte, die nur aus Buchstaben bestehen, sind meistens einfache Worte, die einem „*Wörterbuch-Angriff*“¹ nicht standhalten würden.

¹ Beim sogenannten „Wörterbuch-Angriff“ werden alle Worte aus einem Lexikon/Wörterbuch ausprobiert, da die Wahrscheinlichkeit besteht, dass ein Benutzer als Passwort ein „normales“ Wort (Hund, Katze Maus, Apfel, Banane,...) gewählt hat.

Bei weniger als 8 Zeichen verkleinert sich die Anzahl der möglichen Kombinationen, so dass ein „Brute Force-Angriff“¹ schnell zum Ziel führen würde.

Bei einer Zeichenklasse von 62 Zeichen (a-z A-Z 0-9) und einer Passwort-Länge von 8 Zeichen ergeben sich 62^8 (also etwa 200 Billionen) mögliche Kombinationen. Bei 1 Mio. Versuchen pro Sekunde² würde der Angriff im Maximum rund 6 Jahre³ dauern.

⇒ gespeicherte Passworte

Sowohl das generierte als auch das vom Benutzer gewählte Passwort werden nicht im Klartext gespeichert.

Hierfür existieren so genannte HASH-Funktionen, die von einer Zeichenkette eine in der Länge fest definierte Zeichenkette zurückliefern. Sicherheit bietet die nicht vorhandene Möglichkeit der Umkehrung, heißt: aus einem HASH kann nicht die ursprüngliche Zeichenketten gewonnen werden.

Nebeneffekt: das Datenbank-Management-System unterscheidet nicht zwischen der Groß- und Kleinschreibung. Das Passwort „123Monitor“ ist also für MySQL mit dem Passwort „123monitor“ identisch und beiden würden als korrekt erkannt werden. (Zur Erinnerung: da „a-z“ dasselbe wie „A-Z“ ist, verringert sich die Zeichenklasse von 62 auf 35 Zeichen und somit auf die Maximalzeit für einen erfolgreichen Angriff auf nur 26 Tage!)

Bei der Verwendung von HASH-Werten liefert „123Monitor“ aber einen anderen Wert als „123monitor“, so dass nur eines dieser Passworte den Zugang gestatten würde.

Hat ein Angreifer den HASH-Wert und versucht er nun mittels „Brute Force“ auf die Zeichenkette zurückzukommen, würde dieser Versuch mit großer Wahrscheinlichkeit erfolglos bleiben: die MD5-Funktionen liefert einen Wert mit 32 Zeichen zurück und nutzt als Zeichenklasse 36 Zeichen (a-z 0-9). Somit ergibt sich eine unvorstellbar(!) große Anzahl von möglichen Kombinationen.

¹ Bei dieser Methode werden alle möglichen Kombinationen durchgespielt: a, aa, aaa,...,ZZZZZ

² Diese Zahl ist sehr hoch gegriffen; kein normaler Web-Server wird so eine hohe Anzahl verarbeiten können

³ Zum Vergleich: bei einer Passwortlänge von 6 Zeichen wären es 51 Mrd. Kombinationen (~14 Stunden)

⇒ Rechte-Verwaltung

Um zu verhindern, dass ein Benutzer Zugriff auf Funktionen hat, die er nicht nutzen darf, sollte eine Rechte-Verwaltung genutzt werden, die klar definiert, was dem Benutzer zugänglich ist und was nicht.

Hierbei sollte aber nicht nur das jeweilige Menü des Benutzers angepasst (nicht nutzbare Menüpunkte werden gar nicht erst angezeigt) werden. Jedes Skript ist entsprechend zu schützen, indem bei Skript-Beginn der Ausführer des Skriptes autorisiert und ggf. der Ablauf des Skriptes abgebrochen wird.

⇒ Einstellungen der Programmiersprache

Die Skriptsprache PHP bietet zahllose Optionen, die mittels der Konfigurationsdatei *php.ini* angepasst werden können.

Seit Version 4 wird die Variable „*register_globals*“ auf OFF gesetzt, was zwar einen etwas höheren Programmieraufwand bedeutet, aber aus Sicherheitsgründen zu empfehlen ist.

Des Weiteren sind (sofern nicht explizit von der Anwendung gefordert) Funktionen wie „*system*“, „*exec*“, „*popen*“ und „*passthru*“ zu deaktivieren.

⇒ Benutzereingaben

Die Eingaben eines Benutzers können eine schwere Sicherheitslücke darstellen, sofern nicht entsprechende Maßnahmen ergriffen werden, bestimmte Eingaben zu verbieten bzw. „unschädlich“ zu machen.

Wie im Fall des Online-Auktionshauses **eBay** kann ein Angreifer durch ein kleines Skript in der Programmiersprache „*JavaScript*“ das Passwort eines eBay-Kunden auslesen und so mit dem Benutzerkonto dieses Kunden Geschäfte tätigen.

Eine Möglichkeit, JavaScript und anderen schädlichen Code in einen harmlosen Text zu verwandeln, ist die Umsetzung der TAG-Klammern „*<*“ und „*>*“ in den HTML-Code „*<*“ bzw. „*>*“, so dass die Klammern zwar angezeigt, aber nicht als Teil eines ausführbaren Skriptes interpretiert werden.

⇒ Speicherung von Aktionen

Um auch später noch Benutzer-Aktionen nachvollziehen zu können, sollten alle wichtigen Handlungen (Änderung von Personendaten usw.) geloggt werden.

Hierbei kann auch die Speicherung der IP-Adresse¹ erfolgen, um im Streitfall ggf. ein weiteres Beweismittel zu haben.

5.2 Datenschutz

Ein Grund für die hohen Sicherheitsansprüche von (Web-) Applikationen ist der Datenschutz, also der Schutz personenbezogener Daten.

Als „personenbezogene Daten“ sind Name, Anschrift, Telefonnummer usw. eines Benutzers anzusehen, wobei aber auch nicht-personenbezogene Daten (bspw. die Zeit für das Ausfüllen eines Formulars) als wertvoll einzustufen sind.

Mittels der implementierten Rechte-Verwaltung kann ein Wiederverkäufer nur auf Daten von Kunden zugreifen, die in direktem Zusammenhang mit ihm stehen, also bei ihm bestellt haben, für die er der Rechnungsteller ist oder wenn er den Rang eines übergeordneten Wiederverkäufers innehat.

Zur Verdeutlichung:

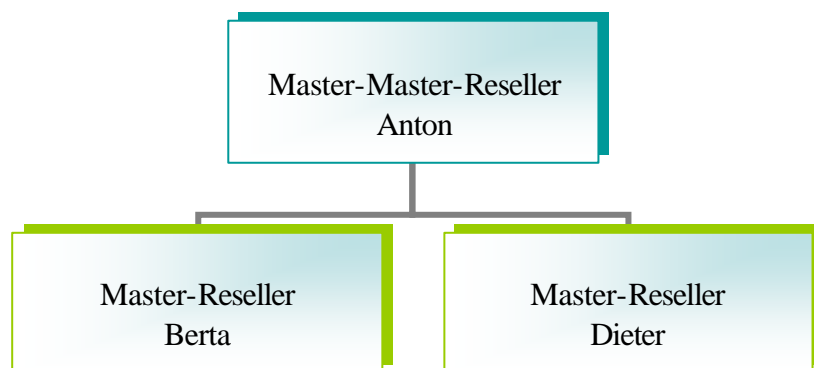


Bild 5.1-1 Abbildung einer Reseller-Hierarchie als Beispiel

¹ die Adresse eines Rechners innerhalb eines Netzwerkes

Die Daten von Kunden, die bei Berta (oder ihr untergeordneten Wiederverkäufern) bestellt haben, können nicht von Dieter eingesehen oder verändert werden.

Lediglich Anton als oberster Wiederverkäufer und somit als „Kontrollorgan“ hat Zugriff auf die Daten der Kunden von Berta und Dieter.

Die Weitergabe der Daten an Dritte für Werbezwecke oder ähnliches wird selbstverständlich über keine Schnittstelle unterstützt.

6 Literaturübersicht und Quellen

- Ammelburger, 2002** Ammelburger, Dirk: „Go To PHP 4“
- Burnett, Paine, 2001** Burnett, Steve und Paine, Stephen: „Kryptographie - RSA Security's Official Guide“ (Deutsche Ausgabe)
- Däßler, 2002** Däßler, Rolf: „MySQL - Das Einsteigerseminar“
- Dellwig, 2000** Dellwig, Ingo: „HTML 4.0 - Nitty Gritty“
- Kofler, 2001** Kofler, Michael: „MySQL - Einführung, Programmierung, Referenz“
- Krause, 2000** Krause, Jörg: „PHP4 - Grundlagen und Profiwissen“

Des weiteren wurden die Onlinehilfen von <http://www.php.net/manual/de> und <http://dev.mysql.com/doc/mysql/de> genutzt.

7 Bilderverzeichnis

Bild 3.2-1 Beispiel für „Eingangsseite“	13
Bild 3.3-1 Grafische Abbildung einer Reseller-Hierarchie	16
Bild 3.3-2 Screenshot „Rechte setzen/ändern“	17
Bild 3.3-3 Screenshot „Menü eines Resellers: Benutzer-Info“	21
Bild 3.3-4 Screenshot „Auflistung der Reseller“	22
Bild 3.3-5 Screenshot „Menü eines Resellers: Vorlagen“	23
Bild 3.3-6 Screenshot „Bearbeitung einer Vorlage“	24
Bild 3.3-7 Screenshot „Firmenlogo hochladen“	24
Bild 3.3-8 Screenshot „Menü eines Resellers: Rechnungen“	25
Bild 3.3-9 Screenshot „Menü eines Resellers: Produkt-Links“	25
Bild 3.3-10 Screenshot „Menü eines Resellers: Persönliche Daten“	26
Bild 3.3-11 Screenshot „Ausschnitt aus einem Bestellformular“	27
Bild 3.3-12 Screenshot „Produkt-Links“ (Fehler)	29
Bild 3.3-13 Screenshot „Produkt-Links“	30
Bild 3.4-1 Beispiel: Statistik „Anzahl Kunden/Reseller“	33
Bild 3.4-2 Beispiel: Statistik „Registrierungen der letzten 12 Monate“	34
Bild 3.4-3 Beispiel: Kunden-Protokoll	35
Bild 3.5-1 Screenshot „Kunden-Hauptmenü“	36
Bild 3.5-2 Screenshot „Bestellungen anzeigen“	36
Bild 4.1-1 Abbildung einer Reseller-Hierarchie als Beispiel	39
Bild 5.1-1 Abbildung einer Reseller-Hierarchie als Beispiel	57

8 **Abbildungsverzeichnis**

Abb. 4.1-1 Quellcode-Auszug „Rechte-Änderung“ – Teil 1	40
Abb. 4.1-2 Quellcode-Auszug „Rechte-Änderung“ – Teil 2	41
Abb. 4.1-3 Quellcode-Auszug „Rechte-Änderung“ – Teil 3	44
Abb. 4.1-4 Quellcode-Auszug „Rechte-Änderung“ – Teil 4	45
Abb. 4.1-5 Quellcode-Auszug „Rechte-Änderung“ – Teil 5	46
Abb. 4.1-6 Quellcode-Auszug „Rechte-Änderung“ – Teil 6	47
Abb. 4.1-7 Quellcode-Auszug „Rechte-Änderung“ – Teil 7	48
Abb. 4.1-8 Quellcode-Auszug „Rechte-Änderung“ – Teil 8	49
Abb. 4.1-9 Quellcode-Auszug „Rechte-Änderung“ – Teil 9	52

9 Erklärung

Ich versichere, dass ich diese Diplomarbeit ohne fremde Hilfe selbstständig verfasst und nur die angegebenen Quellen und Hilfsmittel benutzt habe.

Wörtlich oder dem Sinn nach aus anderen Werken entnommene Stellen sind unter Angabe der Quellen kenntlich gemacht.

Engelskirchen 22. Januar 2005

.....

Ort Datum Unterschrift

10 Kontaktadresse

Martin Amelsberg
Remerscheider Straße 35
51766 Engelskirchen
Telefon : 0 22 63 / 96 99 250
FAX : 0 22 63 / 96 99 251
Homepage : <http://www.AmelFin.de>
eMail : Amel@AmelFin.de

Anhang A: Beispiel eines Kontakt-Formulares

Firma Hans Mustermann

Ihr Warenkorb:

Pos.	Produktbezeichnung	Preis
1.	VISA-Electron <small>Für Bewohner aus Deutschland sind bereits 16% MwSt. (9,52 €) enthalten.</small>	69.01 €
Gesamt:		69.01 €

Persönliche Informationen

Einloggen mit der UICI

UICI

Passwort

Neu registrieren

UICI

Firma

Anrede

Vorname

Nachname

Adresszeile 1

Adresszeile 2

Adresszeile 3


PLZ

Ort

Land

eMail

Anhang B: Beispiel eines VISA-Bestell-Formulares

<p>Card Systems and Services AB 727 Postfach 66, 1011 Wien Fax: 01 717 67 049 200</p>	 <h1 style="margin: 0;">Antragsformular</h1>																								
<input type="checkbox"/> Als Hauptkarteninhaber beantrage ich die Zustellung der Umsatznachricht per E-Mail. Ich verzichte auf die postalische Zusendung.																									
<h3>Persönliche Angaben zur Ausstellung der Hauptkarte:</h3>																									
<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td colspan="2">Titel, Vor- und Zuname</td> <td>Geburtsdatum</td> </tr> <tr> <td colspan="2">Herr Hans Mustermann</td> <td>01.01.1960</td> </tr> <tr> <td>angestellt bei</td> <td>e-mail</td> <td>Telefon (tagsüber)</td> </tr> <tr> <td>Mustermann GmbH</td> <td>Test@AmelFin.de</td> <td></td> </tr> <tr> <td colspan="2">PLZ, Ort, Strasse</td> <td>Land</td> </tr> <tr> <td colspan="2">51766, Engelskirchen, Remerscheider Strasse 35</td> <td>Deutschland</td> </tr> <tr> <td colspan="2">Korrespondenzadresse (wenn nicht ident mit Wohnanschrift) für die Zusendung von Karte & PIN-Kuvert</td> <td>Land</td> </tr> <tr> <td colspan="2"></td> <td></td> </tr> </table>		Titel, Vor- und Zuname		Geburtsdatum	Herr Hans Mustermann		01.01.1960	angestellt bei	e-mail	Telefon (tagsüber)	Mustermann GmbH	Test@AmelFin.de		PLZ, Ort, Strasse		Land	51766, Engelskirchen, Remerscheider Strasse 35		Deutschland	Korrespondenzadresse (wenn nicht ident mit Wohnanschrift) für die Zusendung von Karte & PIN-Kuvert		Land			
Titel, Vor- und Zuname		Geburtsdatum																							
Herr Hans Mustermann		01.01.1960																							
angestellt bei	e-mail	Telefon (tagsüber)																							
Mustermann GmbH	Test@AmelFin.de																								
PLZ, Ort, Strasse		Land																							
51766, Engelskirchen, Remerscheider Strasse 35		Deutschland																							
Korrespondenzadresse (wenn nicht ident mit Wohnanschrift) für die Zusendung von Karte & PIN-Kuvert		Land																							
<p>Für die Benützung der VISA Electron Prepaid-Karte werden folgende Euro-Limite vereinbart:</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td rowspan="3" style="width: 30%; text-align: center; vertical-align: middle;">Für den Bargeldbezug:</td> <td style="width: 35%;"> <input checked="" type="radio"/> Standardlimite (Maximalwerte) </td> <td style="width: 35%;"> <input type="radio"/> (geringere) individuelle Limite </td> </tr> <tr> <td>700,- pro Tag 2.000,- pro Woche</td> <td> <input type="text"/> pro Tag <input type="text"/> pro Woche </td> </tr> <tr> <td>Für bargeldlose Zahlungen: 2.000,- pro Tag und Woche</td> <td> <input type="text"/> pro Tag und Woche </td> </tr> </table>		Für den Bargeldbezug:	<input checked="" type="radio"/> Standardlimite (Maximalwerte)	<input type="radio"/> (geringere) individuelle Limite	700,- pro Tag 2.000,- pro Woche	<input type="text"/> pro Tag <input type="text"/> pro Woche	Für bargeldlose Zahlungen: 2.000,- pro Tag und Woche	<input type="text"/> pro Tag und Woche																	
Für den Bargeldbezug:	<input checked="" type="radio"/> Standardlimite (Maximalwerte)		<input type="radio"/> (geringere) individuelle Limite																						
	700,- pro Tag 2.000,- pro Woche		<input type="text"/> pro Tag <input type="text"/> pro Woche																						
	Für bargeldlose Zahlungen: 2.000,- pro Tag und Woche	<input type="text"/> pro Tag und Woche																							
<p>Im Falle einer missbräuchlichen Verwendung der Karte besteht das Risiko des Verlustes der og. Beträge im jeweiligen Zeitraum. Als Woche gilt der Zeitraum von Samstag, 0.00 Uhr bis zum folgenden Freitag, 24.00 h.</p>																									
<p>Als Startguthaben gilt folgender Euro-Betrag vereinbart:</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td>Betrag Startguthaben (mind. Euro 100,-)</td> </tr> <tr> <td>250</td> </tr> </table>		Betrag Startguthaben (mind. Euro 100,-)	250																						
Betrag Startguthaben (mind. Euro 100,-)																									
250																									
<input checked="" type="checkbox"/> Ich bestätige, dass der unter "Betrag Startguthaben" angeführte Betrag auf folgendes Konto der <small>Kaufhaus Zentralbank Österreich AG, Postfach 66, 1011 Wien, Austria</small> überwiesen wurde/wird:																									
Kontonummer	Bankleitzahl	BIC	IBAN																						
100-076-002	31000	BOBAAT3333	AT770100000100076002																						
<input type="button" value="Anfrage senden"/>																									