

# MMIS - Konstruktionsübung Design

Werner Grassl,  
Bernd Kompass,  
Klaus Matuschek,  
Gerd Suppan

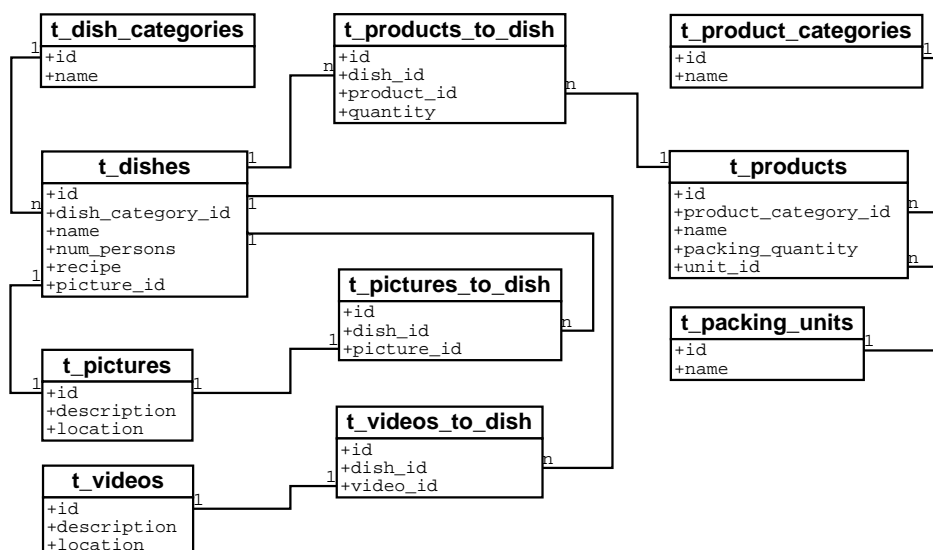
WS 2002, *Revision* : 1.6

# 1 Systemüberblick

Das System besteht aus drei Hauptteilen, einer Datenhaltung in Form einer MySQL Datenbank, einer serverseitigen PHP4 Skripting-Logik zum Zugriff auf die Datenbank, zur benutzerinteraktionsbedingten Webseitengenerierung, sowie aller notwendigen Berechnungen und einem clientseitigen Webinterface. Webinterface und Skripting Logik bilden ihrerseits zusammen die sogenannte Kontrolle des Gesamtsystems. Im folgenden werden nun Datenbank, Kontrolle, sowie zusätzlich Skripting Logik und Webinterface genauer erläutert.

## 1.1 Datenbank

Es kommt eine MySQL Datenbank mit folgenden Tabellen zum Einsatz:



**Gerichtekategorien (t\_dish\_categories)** Eine Sammlung aller möglichen Kategorien für Gerichte, z.B. Suppen, Fleischgerichte, etc., wobei jeder Eintrag aus folgenden Feldern besteht:

**id** Eindeutiger Identifier für Gerichtkategorie.

**name** Aussagekräftiger Name für Gerichtkategorie.

**Produktkategorien (t\_product\_categories)** Eine Sammlung aller möglichen Kategorien für Produkte, z.B. Teigwaren, Wurstwaren, etc., wobei jeder Eintrag aus folgenden Feldern besteht:

**id** Eindeutiger Identifier für Produktkategorie.

**name** Aussagekräftiger Name für Produktkategorie.

**Gerichte (t\_dishes)** Jeder Eintrag in dieser Tabelle entspricht einem Gericht aus einer bestimmten Kategorie und besteht aus den folgenden Feldern:

**id** Eindeutiger Identifier für Gericht.

**dish\_category\_id** Eindeutiger Identifier der Gerichtskategorie, der dieses Gericht zugeordnet ist.

**name** Aussagerkräftiger Name für Gericht.

**num\_persons** Anzahl der Personen für die das erfasste Gericht bzw. sein Rezept grundsätzlich gedacht ist.

**recipe** Das Rezept für das Gericht.

**picture\_id** Ein eindeutiger Identifier eines optional vorhandenen Fotos in der t\_picture Tabelle für das Gericht.

**Produkte (t\_products)** Die Einträge dieser Tabelle representieren jeweils ein Produkt, wobei die einzelnen Felder nachfolgende Bedeutung haben:

**id** Eindeutiger Identifier für Produkt.

**product\_category\_id** Existierender Identifier der Produktkategorie, der dieses Produkt zugeordnet ist.

**name** Aussagekräftiger Name für Produkt.

**packing\_quantity** Grundmenge in der dieses Produkt verfügbar ist.

**unit\_id** Eindeutiger Identifier für Einheit der Grundmenge aus der t\_packing\_units Tabelle.

**Einheiten (t\_packing\_units)** Tabelle zur Haltung aller möglichen Mengeneinheiten für Produkte, die Felder sind wie folgt definiert:

**id** Eindeutiger Identifier für Einheit.

**name** Name bzw. genormtes Kurzzeichen für Einheit.

**Produkt/Gericht Zuordnung (t\_products\_to\_dish)** Diese Tabelle stellt die Verbindung zwischen einem Gericht, seinen Zutaten (Produkten) und den notwendigen Mengen bzgl. einer vordefinierten Anzahl von Personen dar.

**id** Eindeutiger Identifier für Zuordnung.

**dish\_id** Eindeutiger Identifier für Gericht aus der t\_dishes Tabelle.

**product\_id** Eindeutiger Identifier für Produkt aus der t\_products Tabelle.

**quantity** Mengenbedarf des Produktes in Grundmengenheiten zur Zubereitung der grundsätzlich erfassten Portionen des Gerichtes.

**Bilder (t\_pictures)** Tabelle zur Haltung aller Bilder (Bilder von Gerichten, sowie Bilder für Rezept Fotoserien), die Felder sind wie folgt definiert:

**id** Eindeutiger Identifier für Bild.

**description** Eine kurze textuelle Beschreibung des Bildes.

**location** Der URI des Bildes.

**Bild/Gericht Zuordnung (t\_pictures\_to\_dish)** Diese Tabelle stellt die Verbindung zwischen einem Gericht und einer möglichen Rezept Fotoserie für das Gericht dar, die Felder sind wie folgt definiert:

**id** Eindeutiger Identifier für Zuordnung.

**dish\_id** Eindeutiger Identifier des Gerichtes aus der Gerichte Tabelle.

**picture\_id** Eindeutiger Identifier des Bildes aus der Bilder Tabelle.

**Videos (t\_videos)** Tabelle zur Haltung aller Koch Videos für Gerichte, die Felder sind wie folgt definiert:

**id** Eindeutiger Identifier für Video.

**description** Eine kurze textuelle Beschreibung des Videos.

**location** Der URI des Videos.

**Video/Gericht Zuordnung (t\_videos\_to\_dish)** Diese Tabelle stellt die Verbindung zwischen einem Gericht und möglichen Koch Videos für das Gericht dar, die Felder sind wie folgt definiert:

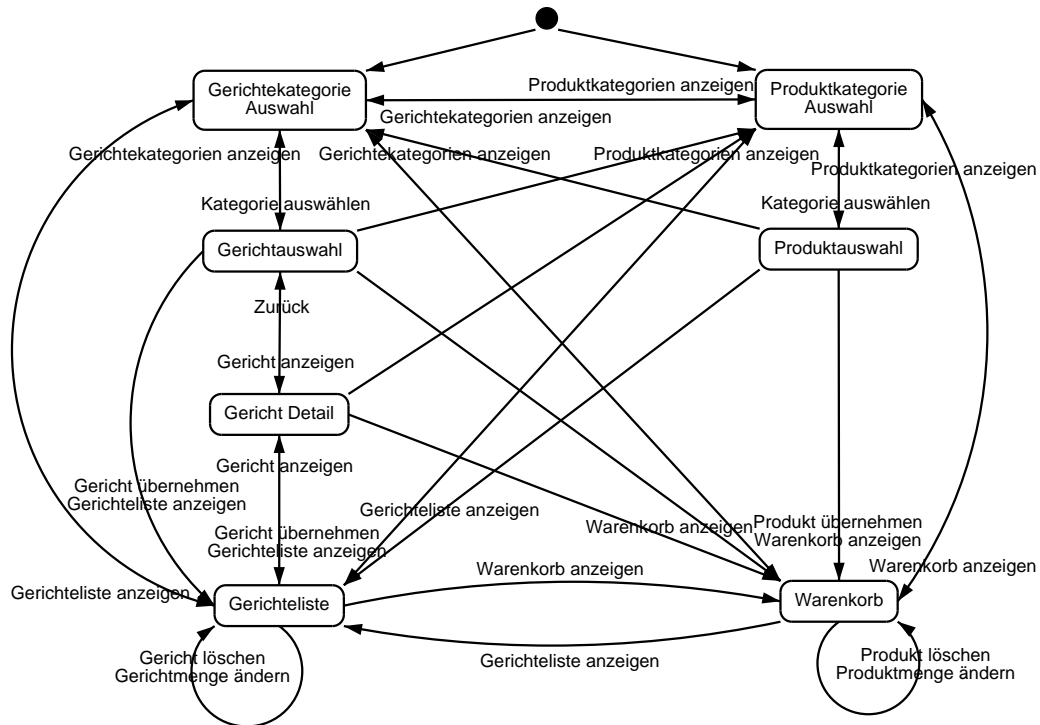
**id** Eindeutiger Identifier für Zuordnung.

**dish\_id** Eindeutiger Identifier des Gerichtes aus der Gerichte Tabelle.

**video\_id** Eindeutiger Identifier des Videos aus der Videos Tabelle.

## 1.2 Kontrolle

Der Kontrollteil des Systems basiert auf serverseitigem Skripting mittels PHP4, wobei sessionorientiert gearbeitet wird und dem Webinterface. Folgendes Zustandsdiagramm bietet die Grundlage des Systems und seiner Interaktionsmöglichkeiten:



In obigem Diagramm stellt jeder Zustand einen sogenannten View in Form einer Webseite auf das System dar und bietet eine bestimmte Anzahl von Interaktionsmöglichkeiten an, die in einen Folgezustand führen und zusätzlich eine Anzahl weiterer Aktionen auslösen können. Die einzelnen Zustände, ihre Transitionen und dabei durchgeführte weitere Aktionen werden jetzt beschrieben.

### 1.2.1 Startzustand

Dies ist eigentlich kein eigener Zustand, er wird nur eingeführt, um die notwendigen Aktionen zu erklären, die zur Gerichtekategorie- bzw. Produktkategorieauswahl führen und um die sessionweiten Objekte zu definieren.

**Startzustand-Gerichtekategorieauswahl** Hierbei ist es notwendig alle Datensätze aus der Gerichtekategorien Tabelle zu holen und dem View

für die Gerichtekategorieauswahl zur Darstellung zu übergeben.

**Startzustand-Produktkategorieauswahl** Hierbei ist es notwendig alle Datensätze aus der Produktkategorien Tabelle zu holen und dem View für die Produktkategorieauswahl zur Darstellung zu übergeben.

### 1.2.2 Gerichtekategorieauswahl

Dieser Zustand dient zur Darstellung aller Gerichtekategorien und bietet primär die Möglichkeit eine Kategorie für Gerichte auszuwählen.

**Gerichtekategorieauswahl-Gerichtauswahl** Wird eine Gerichtekategorie ausgewählt, so wird eine Liste mit allen Gerichten zu dieser Kategorie aus der Gerichte Tabelle erstellt. Diese Liste wird dann dem View für die Gerichtauswahl zur Darstellung übergeben.

**Gerichtekategorieauswahl-Produktkategorieauswahl** Hierbei ist es notwendig alle Datensätze aus der Produktkategorien Tabelle zu holen und dem View für die Produktkategorieauswahl zur Darstellung zu übergeben.

**Gerichtekategorieauswahl-Gerichteliste** Hier erfolgt einfach der Wechsel in den View zur Darstellung der aktuellen Gerichteliste, also der bereits selektierten Gerichte. Wird das erste Mal in diesen View gewechselt, so erfolgt die Erzeugung des initialen Gerichtelisten Objektes und die anschließende Registrierung desselben bei der Session.

**Gerichtekategorieauswahl-Warenkorb** Hier erfolgt einfach der Wechsel in den View zur Darstellung des aktuellen Warenkorbes. Wird das erste Mal in diesen View gewechselt, so erfolgt die Erzeugung des initialen Warenkorb Objektes und die anschließende Registrierung desselben bei der Session.

### 1.2.3 Gerichtauswahl

Die Gerichtauswahl dient der Darstellung aller Gerichte zu einer bestimmten Gerichtkategorie und bietet primär die Möglichkeit ein Gericht in die Gerichteliste zu übernehmen.

**Gerichtauswahl-Gerichteliste** Es gibt zwei Möglichkeiten um von der Gerichtauswahl zur Gerichteliste zu gelangen:

1. Einfacher Wechsel in den View zur Darstellung des Gerichtelisten Objektes. Wird das erste Mal in diesen View gewechselt, so erfolgt die Erzeugung des initialen Gerichtelisten Objektes und die anschließende Registrierung desselben bei der Session.
2. Auswahl eines Gerichtes und hinzufügen zum sessionweiten Gerichtelisten Objektes, dies bedeutet zuerst das Nachladen aller Zutaten (Produkte) dieses Gerichtes aus der Produkt/Gericht Zuordnungs Tabelle. Danach wird das sessionweite Gerichtelisten Objekt in seinem vorgesehenen View dargestellt.

**Gerichtauswahl-Gerichtdetail** Zusätzlich wird die Möglichkeit geboten, eine Detailansicht jedes Gerichtes zu erhalten. Hierbei ist es ebenfalls notwendig zuerst alle Zutaten (Produkte) des Gerichtes aus der Produkt/Gericht Zuordnungs Tabelle nachzuladen bevor das Gericht dann in seinem Detail View dargestellt werden kann.

**Gerichtauswahl-Produktkategorieauswahl** Hierbei ist es notwendig alle Datensätze aus der Produktkategorien Tabelle zu holen und dem View für die Produktkategorieauswahl zur Darstellung zu übergeben.

**Gerichtauswahl-Gerichtekategorieauswahl** Hierbei ist es notwendig alle Datensätze aus der Gerichtekategorien Tabelle zu holen und dem View für die Gerichtekategorieauswahl zur Darstellung zu übergeben.

**Gerichtauswahl-Warenkorb** Hier erfolgt einfach der Wechsel in den View zur Darstellung des bei der Session eingangs registrierten Warenkorb Objektes.

#### 1.2.4 Gerichtdetail

Der Gerichtdetail Zustand bietet die Möglichkeit der detaillierten Darstellung eines Gerichtes mit Bild, Rezept, Zutaten, etc. und ermöglicht primär auch das Übernehmen des Gerichtes in die Gerichteliste.

**Gerichtdetail-Gerichteliste** Es gibt zwei Möglichkeiten um vom Gerichtdetail View zur Gerichteliste zu gelangen:

1. Einfacher Wechsel in den View zur Darstellung des bei der Session registrierten Gerichtelisten Objektes.
2. Auswahl eines Gerichtes und hinzufügen zum sessionweiten Gerichtelisten Objektes, dies bedeutet zuerst das Nachladen aller

Zutaten (Produkte) dieses Gerichtes aus der Produkt/Gericht Zuordnungs Tabelle. Danach wird das sessionweite Gerichtelisten Objekt in seinem vorgesehenen View dargestellt.

**Gerichtdetail-Produktkategorieauswahl** Hierbei ist es notwendig alle Datensätze aus der Produktkategorien Tabelle zu holen und dem View für die Produktkategorieauswahl zur Darstellung zu übergeben.

**Gerichtdetail-Gerichtekategorieauswahl** Hierbei ist es notwendig alle Datensätze aus der Gerichtekategorien Tabelle zu holen und dem View für die Gerichtekategorieauswahl zur Darstellung zu übergeben.

**Gerichtdetail-Warenkorb** Hier erfolgt einfach der Wechsel in den View zur Darstellung des bei der Session registrierten Warenkorb Objektes.

**Gerichtdetail-Gerichtsauswahl** Es erfolgt wieder der Übergang zur Gerichtsauswahl, wobei als Gerichtskategorie jene des Gerichtes angenommen wird für welches die Detailansicht geöffnet wurde.

### 1.2.5 Gerichteliste

Die Gerichteliste beinhaltet alle aktuell selektierten Gerichte und die Anzahl der Personen für die jedes Gericht bestellt wurde.

**Gerichteliste-Gerichteliste** Es gibt zwei mögliche Aktionen, nach deren Ausführung kein Wechsel in einen anderen Zustand erfolgt:

1. Löschen eines Gerichtes aus der Gerichteliste und darauffolgender Neudarstellung der Gerichteliste.
2. Ändern der Anzahl der Personen für die die Zutaten des Gerichtes zusammengestellt werden sollen, was ebenfalls eine darauffolgende Neudarstellung der Gerichteliste zur Folge hat.

**Gerichteliste-Gerichtdetail** Um die Detailansicht eines Gerichtes aus der Gerichteliste zu erhalten ist es notwendig zuerst alle Zutaten (Produkte) des Gerichtes aus der Produkt/Gericht Zuordnungs Tabelle nachzuladen bevor das Gericht dann in seinem Detail View dargestellt werden kann.

**Gerichteliste-Gerichtekategorieauswahl** Hierbei ist es notwendig alle Datensätze aus der Gerichtekategorien Tabelle zu holen und dem View für die Gerichtekategorieauswahl zur Darstellung zu übergeben.

**Gerichteliste-Produktkategorieauswahl** Hierbei ist es notwendig alle Datensätze aus der Produktkategorien Tabelle zu holen und dem View für die Produktkategorieauswahl zur Darstellung zu übergeben.

**Gerichteliste-Warenkorb** Hier erfolgt einfach der Wechsel in den View zur Darstellung des bei der Session registrierten Warenkorb Objektes.

### 1.2.6 Produktkategorieauswahl

Dieser Zustand dient zur Darstellung aller Produktkategorien und bietet primär die Möglichkeit eine Kategorie für Produkte auszuwählen.

**Produktkategorieauswahl-Gerichtekategorieauswahl** Hierbei ist es notwendig alle Datensätze aus der Gerichtekategorien Tabelle zu holen und dem View für die Gerichtekategorieauswahl zur Darstellung zu übergeben.

**Produktkategorieauswahl-Gerichteliste** Hier erfolgt einfach der Wechsel in den View zur Darstellung des bei der Session registrierten Gerichtelisten Objektes.

**Produktkategorieauswahl-Warenkorb** Hier erfolgt einfach der Wechsel in den View zur Darstellung des bei der Session registrierten Warenkorb Objektes.

**Produktkategorieauswahl-Produktauswahl** Wird eine Produktkategorie ausgewählt, so wird eine Liste mit allen Produkten zu dieser Kategorie aus der Produkte Tabelle erstellt. Diese Liste wird dann dem View für die Produktauswahl zur Darstellung übergeben.

### 1.2.7 Produktauswahl

Die Produktauswahl dient der Darstellung aller Produkte zu einer bestimmten Produktkategorie und bietet primär die Möglichkeit ein Produkt in den Warenkorb zu übernehmen.

**Produktauswahl-Gerichtekategorieauswahl** Hierbei ist es notwendig alle Datensätze aus der Gerichtekategorien Tabelle zu holen und dem View für die Gerichtekategorieauswahl zur Darstellung zu übergeben.

**Produktauswahl-Produktkategorieauswahl** Hierbei ist es notwendig alle Datensätze aus der Produktkategorien Tabelle zu holen und dem View für die Produktkategorieauswahl zur Darstellung zu übergeben.

**Produktauswahl-Gerichteliste** Hier erfolgt einfach der Wechsel in den View zur Darstellung des bei der Session registrierten Gerichtelisten Objektes.

**Produktauswahl-Warenkorb** Es gibt zwei Möglichkeiten um von der Produktauswahl zum Warenkorb zu gelangen:

1. Einfacher Wechsel in den View zur Darstellung des bei der Session registrierten Warenkorb Objektes.
2. Auswahl eines Produktes, welches in das sessionweite Warenkorb Objekt übernommen wird, welches anschliessen in seinem View dargestellt wird.

### 1.2.8 Warenkorb

Der Warenkorb beinhaltet alle aktuell selektierten Produkte, also jene, die durch die Auswahl eines Gerichtes eingetragen wurden, sowie auch zusätzlich in der Produktauswahl selektierten.

**Warenkorb-Warenkorb** Es gibt zwei mögliche Aktionen, nach deren Ausführung kein Wechsel in einen anderen Zustand erfolgt:

1. Löschen eines Produktes aus dem Warenkorb und darauffolgender Neudarstellung des Warenkorbes.
2. Ändern der Menge des Produktes und darauffolgende Neudarstellung des Warenkorbes.

**Warenkorb-Gerichteliste** Hier erfolgt einfach der Wechsel in den View zur Darstellung des bei der Session registrierten Gerichtelisten Objektes.

**Warenkorb-Gerichtekategorieauswahl** Hierbei ist es notwendig alle Datensätze aus der Gerichtekategorien Tabelle zu holen und dem View für die Gerichtekategorieauswahl zur Darstellung zu übergeben.

**Warenkorb-Produktkategorieauswahl** Hierbei ist es notwendig alle Datensätze aus der Produktkategorien Tabelle zu holen und dem View für die Produktkategorieauswahl zur Darstellung zu übergeben.

## 1.3 Skriptbasierte Logik

Die PHP4 Skriptlogik basiert auf einer objektorientierten Struktur. Die Verbindung mit der Datenbank ist in einer eigenen Klasse gekapselt, welche über ihre Zugriffsmethoden Abbildungen bzw. Kompositionen der verschiedenen



### 1.3.2 Warenkorbberechnung

Wie vorher erwähnt, erfolgt die Berechnung des aktuellen Warenkorb Inhaltes dynamisch, je nach Bedarf. Dabei wird ein Objekt der Klasse *ShoppingList* erzeugt, der Referenzen auf die sessionpermanente Gerichteliste und Produktliste übergeben werden. Aus den Zutaten jedes Gerichtes (unter Berücksichtigung der Portionen, die für das Gericht gewählt wurden) und den zusätzlich selektierten Produkten wird dann die notwendige Menge für jedes Produkt ermittelt und von dieser Menge auf die nächstmöglich verfügbare Verpackungsmenge aufgerundet.

### 1.3.3 Zusätzliche (De-)Selektion von Produkten

Damit es möglich ist, auch im Warenkorb die Anzahl der Produkte zu erhöhen bzw. zu vermindern, oder ein Produkt sogar zu löschen, wird eine eigene Produkteliste in der Session mitgeführt. Diese kann intern auch negative Werte für die Menge bzw. die Anzahl halten. Dies ist z.B. notwendig, um nach der Deselektion eines Produktes aus dem Warenkorb, welches aber von einem Gericht benötigt wird, dieses Produkt bei einer abermaligen Warenkorbberechnung nicht wieder anzuzeigen.

## 1.4 Webinterface

Das Webinterface des Systems, also der sogenannte View ist wie im vorhergehenden Abschnitt bereits ersichtlich war, Teil der Kontrolle. Prinzipiell gibt es für jeden Zustand der obigen Zustandsmaschine eine Darstellung in Form einer PHP Seite, welche neben ihrem statischen Inhalt die notwendigen Darstellungsroutinen der einzelnen Objekte des Modells aufruft um den dynamischen Inhalt zu generieren. Eingaben, also Interaktionen, finden in Form von Hyperlinks bzw. Formularen statt.

### 1.4.1 Seitenlayout

Mit Hilfe der HTML-Ergänzungssprache Cascading Stylesheets (CSS) lassen sich HTML Elemente exakt formatieren. Somit lässt sich sehr elegant die Gestaltung des Seitenlayouts von der Formatierung der einzelnen Elemente entkoppeln. Die formatbeschreibende Datei *styles/allib.css* wird von allen PHP generierten Seiten im HTML HEAD Abschnitt inkludiert. Im Folgenden werden die Eigenschaften der wichtigsten Klassen erläutert:

Globale Seitenvorgaben *body*: Text/Hintergrundfarbe, Schriftart/ größe, Ersatzschriftarten.

Tabellenklasse für Hauptseitenlayout *table.main*: Enthält unter anderem die Angabe für das sich in X-Richtung wiederholende Hintergrundbild für den Seitenkopf, sowie die Hintergrundfarbe für den Datenbereich.

Tabellenklasse für Navigationselemente *table.navigation*: Enthält diverse Angaben für Farben und Hyperlinkformatierung (z.B. *hover* Effekt).

Tabellenklassen für Gerichte (*table.dishes*, Waren (*table.products*), Gerichteliste (*table.dishlist*), sowie Warenkorb (*table.shoppingcart*): Diese enthalten hauptsächlich kleinere Farbvorgaben.

CSS ermöglicht sehr elegant die Einhaltung der *corporate identity*, Änderungen des Layouts der gesamten Site müssen normalerweise lediglich an einer Stelle, dem Stylesheet vorgenommen werden und wirken sich sofort auf alle Seiten aus.

## 2 Installation

### 2.1 Systemvoraussetzungen

Zum Betrieb des ALLIB Systems sind ein Apache Webserver mit PHP4.1.2 Erweiterung (<http://www.apache.org/>) und ein MySQL Server (<http://www.mysql.com/>) notwendig. Zur einfachen Datenbankadministration per Webinterface empfiehlt sich *phpMyAdmin* (<http://www.phpmyadmin.net/>).

### 2.2 Datenbank

Die gepackte Datenbank befindet sich im Verzeichnis *database* der Datei *allib.tar.gz*. Diese Datei in ein lokales Verzeichnis entpacken und anschliessend die Datenbank auf dem MySQL Server durch Einspielen der Datei *database/allib.sql*, z.B. mit Hilfe von *phpmyadmin* erstellen. In der MySQL Datenbank ist ein Benutzername und ein Passwort zum Zugriff auf die Datenbank anzugeben.

### 2.3 Site

Der Verzeichnisbaum der Site befindet sich, je nach PHP Version im Verzeichnis *allib-4.0.6* oder *allib-4.1.2* der gezippten Datei. Den gewünschten Verzeichnisbaum in z.B. das *DocumentRoot* Verzeichnis des Webservers entpacken. Unter Umständen müssen die MySQL Verbindungsdaten und der *include\_path* in der Datei *allib/include/Database.class.inc* angepasst werden.