



Hochschule Neubrandenburg  
University of Applied Sciences

Fachbereich: Landschaftsarchitektur, Geoinformatik,  
Geodäsie, Bauingenieurwesen

Fachgebiet: Geoinformatik

Bachelorarbeit  
zum Thema

## **Entwurf und Implementation einer Praktikumsbörse**

urn:nbn:de:gbv:519-thesis2008-0268-4

Zum Erlangen des akademischen Grades  
„Bachelor of Engineering“ (B.Eng.)

Vorgelegt von

Anja Diekhoff

Betreuer

Dipl.-Inform. J. Schäfer

Prof. Dr.-Ing. E. Heil

Neubrandenburg, 06. Oktober 2008

## **Eidesstattliche Erklärung**

Hiermit versichere ich, die vorliegende Bachelorarbeit ohne Hilfe Dritter und nur mit den angegebenen Quellen und Hilfsmitteln angefertigt zu haben. Alle Stellen, die aus den Quellen entnommen wurden, sind als solche kenntlich gemacht worden. Diese Arbeit hat in gleicher oder ähnlicher Form noch keiner Prüfungsbehörde vorgelegen.

Neubrandenburg, den 06. Oktober 2008

*Unterschrift*

## **Danksagung**

An dieser Stelle möchte ich die Gelegenheit nutzen, um mich bei allen zu bedanken, die mich während der Entstehung dieser Arbeit unterstützt haben.

Ein spezieller Dank richtet sich an Herrn Dipl. -Inf. J. Schäfer und Herrn Prof. Dr.-Ing. E. Heil für die hervorragende Betreuung meiner Bachelorarbeit. Mit ihren Fragen und Anmerkungen zu meiner Arbeit haben sie mir sehr geholfen.

Ein besonderer Dank geht an meine Familie und meine Freunde für das Interesse an meinen Fortschritten und der ständigen Motivation.

Neubrandenburg, den 06.Oktober 2008

## **Kurzfassung**

Im Rahmen dieser Bachelorarbeit wurde eine Praktikumsbörse entworfen und implementiert. Das daraus entstandene System macht eine Suche nach Praktikumsangeboten möglich. Dabei wird die Programmierschnittstelle von Google (Google Maps API) genutzt, um die Praktikumsorte in einer Karte zu visualisieren. Die Daten zu den Praktikumsangeboten und den Praktikanten werden in einer MySQL-Datenbank verwaltet. Die Eingabe neuer Angebote in die Datenbank ist über ein Formular im Webinterface möglich. Dabei werden die Adressen der Standorte geocodiert. Den Adressen werden Koordinaten zugeordnet.

## **Abstract**

Within the scope of this bachelor thesis an internship exchange was sketched and implemented. The resultant system makes a search for internship offers possible. Besides, the Google application programming interface is used (Google Maps API) to visualise the internship places in a map. The data to the internship offers and the interns are administered in a MySQL database. The input of new offers in the database is possible via a form in the web interface. Besides, the addresses of the locations are going to be geocoded. Coordinates are associated to the addresses.

# Inhaltsverzeichnis

1. Einleitung und Ziele dieser Bachelorarbeit.....	6
2. Grundlagen.....	7
2.1. Beispiele für Praktikumsbörsen.....	7
2.1.1. Praktikanten- und Diplomandenbörse Mecklenburg-Vorpommern.....	7
2.1.2. Praktikumsbörse Universität Mannheim.....	8
2.1.3. Praktikant24.de.....	9
2.1.4. Praktikantenbörse ingenieurkarriere.de.....	10
2.1.5. Hochschule Neubrandenburg.....	11
2.2. Technologie der Webseite der Hochschule.....	14
2.3. Google Maps.....	14
2.3.1. Google Maps API.....	15
2.4. Geocoding.....	16
3. Verwendete Software und Stand der Technik.....	17
3.1. Verwendete Software.....	17
3.2. Stand der Technik.....	18
4. Planung und Entwicklung.....	19
4.1. Funktionalitäten der Praktikumsbörse.....	19
4.2. Datenbankentwurf.....	22
5. Implementierung.....	30
5.1. MySQL-Datenbank.....	30
5.2. Webinterface.....	31
5.2.1. Layout.....	32
5.2.2. Anmeldung der Nutzer.....	33
5.2.3. Suche nach Praktikumsangeboten.....	35
5.2.4. Visualisierung der Praktikumsorte.....	36
5.2.5. Dateneingabe mit Geocodierung.....	43
6. Testphase.....	48
7. Schlussbetrachtung und weiterführende Arbeiten .....	52
Literaturverzeichnis.....	54
Abbildungsverzeichnis.....	56
Tabellenverzeichnis.....	57
Anhang.....	58

# 1. Einleitung und Ziele dieser Bachelorarbeit

Heutzutage ergreifen eine Vielzahl von Webseiten die Möglichkeit Karten in ihrem Webauftritt einzubauen. Denn die Integration einer Karte ist eine gute Möglichkeit den Nutzwert und den Service der Webseite zu steigern. Gegenwärtig gibt es mehrere Webdienste, die Karten und Satellitenbilder zur Verfügung stellen. Einer dieser Dienste ist Google Maps. Google bietet dazu eine Programmierschnittstelle (Google Maps API) an, mit welcher Karten einfach in eigene Web-Anwendungen eingebunden werden können. Zudem ist Google einer der führenden Dienste auf diesem Gebiet. Es deckt einen großen Teil der Welt in Form von Satellitenbildern, Stadtplänen, Landkarten und kombinierten Hybridansichten ab. Des Weiteren können Anwender die ständig weiter verfeinerten Inhalte um eigene Beschreibungen, Grafiken und Symbole mit Hilfe der Google Maps API ergänzen [1]. Über die Programmierschnittstelle kann durch einen JavaScript-Code auf den Google-Maps-Dienst zugegriffen werden.

Im Rahmen dieser Bachelorarbeit ist eine Praktikumsbörse zu entwerfen und zu implementieren. Dabei soll die Google Maps API genutzt werden, um eine eigene Visualisierung der Praktikumsorte umzusetzen und Informationen über die Praktikumsangebote bereitzustellen. Dazu muss zunächst definiert werden, welche Daten zu den Praktikumsstellen benötigt werden um informativ für den Nutzer zu sein und auch um die Orte der Praktika in einer Karte markieren zu können. Diese Daten sollen in einer MySql-Datenbank verwaltet werden. Mit Hilfe eines Webinterface soll es zum einen möglich gemacht werden, nach Praktikumsangeboten zu suchen und zum anderen neue Angebote in die Praktikumsbörse eingeben zu können, welche sofort geocodiert und die Längen und Breitenangaben in die Datenbank eingetragen werden.

Die Praktikumsbörse wird entwickelt, um später in die Webpräsentation des Studiengangs Geoinformatik eingebunden zu werden. Doch sie wird von Anfang an so entworfen, dass Praktika für mehrere unterschiedliche Studiengänge in der Praktikumsbörse angeboten werden können. Eine Praktikumsbörse dieser Art gibt es an der Hochschule Neubrandenburg noch nicht. Somit kann diese eine große Unterstützung für die Studenten bei der Suche nach einer geeigneten Praktikumsstelle werden.

In den nun folgenden Kapiteln werden zunächst einige Grundlagen geklärt, der Entwurf der Praktikumsbörse beschrieben und dessen Implementierung dokumentiert.

## 2. Grundlagen

Bevor damit begonnen wird eine Praktikumsbörse zu entwerfen, sollte geklärt werden, was unter einer Praktikumsbörse zu verstehen ist. Es werden Fragen aufgeworfen wie: Was muss eine Praktikumsbörse enthalten? Was muss sie leisten können? Wie sieht eine „gute“ Praktikumsbörse aus? Eine eindeutige Definition gibt es nicht. Aus diesem Grund sollen folgende Beispiele verdeutlichen, was von einer Praktikumsbörse erwartet wird. Dabei sollen Bewertungen, wie „gut“ oder „schlecht“, Angaben zur Benutzerfreundlichkeit aus der Sicht eines Studenten geben.

### 2.1. Beispiele für Praktikumsbörsen

#### 2.1.1. Praktikanten- und Diplomandenbörse Mecklenburg-Vorpommern

**Suche nach:**  
**Fachgebiet**  
**Stadt**  
**Volltext**  
**Mehr**

Sie sind der 153409. Nutzer seit dem 20.04.1997. Zur Zeit liest 1 Nutzer diese Seiten. Sie arbeiten vom Rechner "79.194.241.240" mit dem Browser "Mozilla/5.0 (Windows; U; Windows NT 6.0; de; rv:1.9.0.1) Gecko/2008070208 Firefox/3.0.1" seit 14:23:05 Uhr am 8.9.2008.

**Praktikantenbörse Mecklenburg-Vorpommern**

Seit 1993 wird vom Projektbüro **des TZW** in Zusammenarbeit mit der **Transferstelle und dem Rechenzentrum der Universität Rostock** die Praktikanten-/Diplomandenbörse M-V betreut.

Diese Datenbank enthält Themenstellungen vorwiegend von Firmen aus Mecklenburg-Vorpommern.

**Technologiezentrum Warnemünde e.V.**  
Projektbüro  
Friedrich- Barnewitz- Str. 3  
18119 Rostock-Warnemünde  
Tel. (0381) 5196 132  
Fax (0381) 5196 266

**Aufnahme neuer Themen:**  
[Word-Formular](#)

Haben Sie Fragen?  
**eMail:** [Praktikantenbörse M-V](#)

Abbildung 1: <http://www.uni-rostock.de/transfer/> (08.09.2008)

Diese Praktikanten- und Diplomandenbörse dient der Suche nach Praktikantenstellen und Diplomarbeiten, vorwiegend in Mecklenburg-Vorpommern. Sie ist sehr einfach und überschaubar aufgebaut. Die zur Verfügung gestellten Suchkriterien als Dropdown-Liste sind Fachgebiet und Stadt. Bei der Suche nach Fachgebieten werden alle Praktikantenstellen und Diplomarbeiten in dieser Kategorie ausgegeben und sortiert, wahlweise nach Stadt oder Zeit. Die Suche über das Kriterium Stadt erfolgt ähnlich, nur

dass hier nach Fachgebiet oder Zeit sortiert werden kann. Eine Alternative zu diesen Suchabläufen bietet die angebotene Volltextsuche, die alle Ergebnisse darlegt, die die gesuchte Zeichenfolge enthalten. Zunächst werden Informationen über Fachgebiet, Ort, Firma und Zeitraum (von-bis) gegeben. Über eine Verlinkung können weitere Details zum Praktikumsangebot, wie z.B. Thema, Branche, Anschrift und Ansprechpartner, eingeholt werden. An dieser Stelle erfährt der Benutzer, ob es sich bei dem Angebot um eine Praktikantenstelle oder eine Aufgabe für Diplomanden handelt.

Die Praktikanten- und Diplomandenbörse M-V ist durch ihre Einfachheit für den Benutzer leicht zu erfassen und zu bedienen. Ungünstig ist es jedoch, dass der Student erst nach der Suche erfährt, ob es sich bei dem Ergebnis um eine Praktikantenstelle oder eine Aufgabenstellung für eine Diplomarbeit handelt. Es wäre wahrscheinlich von Vorteil, wenn der Nutzer nach weiteren Kriterien, wie zum Beispiel Praktikum oder Diplomarbeit, suchen könnte. Nach einer Suchanfrage mit vielen Ergebnissen wird es schnell unübersichtlich, da alle Angebote auf einer Seite dargestellt werden.

## 2.1.2. Praktikumsbörse Universität Mannheim

The screenshot shows the homepage of the Praktikumsbörse der Fakultät BWL at the University of Mannheim. The page has a blue header with the university logo and a navigation menu on the left. The main content area is divided into sections for login, search, and newsletter sign-up.

**Navigation Menu:**

- Homepage BWL
- Start Praktikum
- Impressum

**Main Content Area:**

Herzlich Willkommen bei der Praktikumsbörse der Fakultät BWL

Unternehmen	Studenten
<p><u>Kein Login vorhanden</u></p>	<p><u>Neueste Angebote anzeigen</u></p> <p><u>Nur Diplomarbeiten anzeigen</u></p> <p><u>Die 10 meistaufgerufenen Angebote</u></p>
<p>Bestehender Login:</p> <p>Login: <input type="text"/></p> <p>Passwort: <input type="password"/></p> <p><input type="button" value="Einloggen"/></p>	<p>Suche nach Studienggebiet: (geordnet nach Fakultäten)</p> <p>Betriebswirtschaftslehre Rechtswissenschaften Informatik + Mathe Philosophische Fakultät Volkswirtschaftslehre</p> <p><input type="button" value="Aktuelle anzeigen"/></p>

**Search and Newsletter Section:**

Gewünschte Postleitzahl (Erste 2 Ziffern):

Volltextsuche:

Newsletter bestellen:

Ihre E-Mail:

Abbildung 2: <http://praktikum.wifo.uni-mannheim.de/> (08.09.2008)



Die Universität Mannheim bietet ihren Studenten, sowie jedem Anderen die Möglichkeit auf ihrer Seite nach Praktika und Diplomarbeiten zu suchen. Zugleich wird den Betrieben nach einer Anmeldung die Möglichkeit geboten, selber Praktika einzutragen.

Die Suche kann über eine Dropdown-Liste nach Studiengebieten erfolgen. Des Weiteren kann der Benutzer eine Postleitzahl eingeben und sich alle Ergebnisse in diesem Gebiet auflisten lassen. Bei jeder Suche wird nach dem Datum des Eintrags sortiert. Zudem wird angeboten, die neuesten und die 10 am häufigsten aufgerufenen Angebote darstellen zu lassen. Als weitere Möglichkeit besteht die Volltextsuche.

Von Vorteil ist bei der Praktikumsbörse der Universität Mannheim, dass die Ergebnisse immer nach dem Eintragsdatum geordnet sind, somit stehen die aktuellsten Angebote immer an erster Stelle. Es werden viele Informationen über das Praktikum bereitgestellt, wie z.B. Aufgabenstellung, Anforderung an den Bewerber, Angaben zur Firma usw..

### 2.1.3. Praktikant24.de

The screenshot shows the website 'Praktikant24' with the tagline 'Lernen, Lehren und Studieren mit Praxisbezug.' and an IHK logo. The navigation bar includes links like Home, Praktikum, Duale Studiengänge, Studium, Ausbildung, Beruf, Aktuelles, Über uns, and Mein Account. The main content area is titled 'Suche nach Stellenangeboten' and includes a search form with the following fields:

- Praktikumsart:** A dropdown menu with options: Berufspraktisches Semester (BPS), Praktikum allg. / Fachpraktikum / Industriepraktikum, and Semesterferienjob.
- Einsatzbereiche:** A list box containing: Buchhaltung / Controlling, DV. / Software, Geschäftsführung, Kaufmännischer Bereich (Einkauf / Vertrieb / Verkauf), Konstruktion / Planung / Arbeitsvorbereitung, and Labor / Forschung / Entwicklung.
- Studienrichtung:** A list box containing: Bauen, Biologie - Medizin, Chemie, Elektrotechnik - Elektronik, and Erziehung.
- PLZ Einsatzort:** A text input field and a distance dropdown menu set to '10 km'.

A 'Finden' button is located at the bottom right of the search form. On the right side of the page, there are sections for 'Seitenfunktionen' (with links to 'Seite drucken' and 'Seite empfehlen'), 'Volltextsuche' (with a search bar), 'Praktikabörse' (with a link to 'Zur Börse'), and 'Duale Studiengänge' (with a link to 'Zur Börse').

Abbildung 3: <http://www.praktikant24.de/praktikum/praktikumsboerse/angebote/studenten/> (08.09.2008)

Praktikant24 gibt vier Kriterien vor, die bei der Suche nach Praktikums- oder Diplomarbeitsstellen helfen. Dabei werden für Praktikumsart, Einsatzbereich und

Studienrichtung Auswahlfelder angeboten. Das letzte Kriterium besteht aus der Postleitzahl des Einsatzortes und dem Umkreis, in dem gesucht werden soll. Eine Volltextsuche wird hierbei lediglich für die Informationsseiten von Praktikant24.de durchgeführt, es wird nicht in den Inseraten der Praktikantenbörse gesucht. In dieser Börse sind eine Vielzahl an Informationen hinterlegt. Damit kann eine große Bandbreite an Gebieten abgedeckt werden, doch spezielle Fachgebiete, wie Geoinformatik, sind nicht benannt. Es muss auf allgemeinere Begriffe umgestiegen werden, was dazu führt, dass die Suchergebnisse unspezifischer werden.

#### 2.1.4. Praktikantenbörse ingenieurkarriere.de

Sie befinden sich hier: [ingenieurkarriere.de](http://www.ingenieurkarriere.de) » [Stellenmarkt](#) » Praktikantenbörsen

**PRAKTIKANTENBÖRSE**  
Herzlich willkommen bei START - der Praktikantenbörse für Ingenieure online.  
In Kooperation mit praktika.de bietet ingenieurkarriere.de Zugriff auf eine große Auswahl relevanter Praktikumsstellen im In- und Ausland.  
Finden Sie mit START schnell und einfach Ihr Traumpraktikum und STARTen Sie erfolgreich in Ihre berufliche Zukunft.  
ingenieurkarriere.de wünscht viel Erfolg!

**Katalogsuche**

- ☐ Einsatzgebiet
- ☐ Branche
- ☐ Studienrichtung
- ☐ Großraum
- ☐ Praktikumssprache
- ☐ Bundesland
- ☐ Land

[Suche starten ▶](#)

**Volltextsuche**  
[Suche starten ▶](#)  
Tipp: Die Volltextsuche durchsucht alle Felder der Datenbank nach dem eingegebenen Schlagwort. Sogenannte Boolesche Operatoren wie "UND", "NICHT" und "ODER" werden zur Zeit nicht unterstützt, ebenso wie Wildcards, sogenannte Platzhalter (Zeichen \*) oder Joker (?).

**FÜR UNTERNEHMEN**  
[Mediadaten, Preise, Produkte](#)

**JOBSCHNELLSUCHE**  
Suchbegriff  
  
[Suche starten ▶](#)  
[Erweiterte Suche / Hilfe](#)

**ETIKETTE FÜR INGENIEURE**  
Verspätung: Sie kommen zu spät zur Besprechung. Wie verhalten Sie sich richtig?  
Möchten Sie die Frage beantworten?  
[hier klicken](#)


**VERDIENEN SIE GENUG?**  
Ermitteln Sie Ihren Marktwert und 

Abbildung 4:

<http://www.ingenieurkarriere.de/bewerberservice/unternehmenskontakte/praktikumsboerse>  
(08.09.2008)

Die Praktikantenbörse unter ingenieurkarriere.de bietet eine Katalogsuche für die Recherche nach Praktikumsstellen im In- und Ausland an. Es kann nach 7 Kriterien gesucht werden, die nicht miteinander kombinierbar sind. Als Alternative dazu wird die

Volltextsuche angeboten, mit dem Vermerk: „Tipp: Die Volltextsuche durchsucht alle Felder der Datenbank nach dem eingegebenen Schlagwort. Boolesche Operatoren wie "UND", "NICHT" und "ODER" werden zur Zeit nicht unterstützt, ebenso wie Wildcards, sogenannte Platzhalter (Zeichen \*) oder Joker (?).“ Diese Praktikumsbörse hat zu jedem Praktikum eine Vielzahl an Informationen hinterlegt, einige davon können erst nach einem Login eingesehen werden. Des Weiteren wird die Funktion angeboten, sich eine Praktikumsstelle zu merken oder sich alle Stellen der gleichen Firma ansehen zu können. Die Praktikantenbörse von ingenieurkarriere.de ermöglicht das Suchen nach vielen verschiedenen Faktoren, doch diese können nicht miteinander verknüpft werden. Unter Umständen ergibt eine Suche in dieser Praktikumsbörse eine unübersichtliche Zahl an Ergebnissen, insbesondere bei der angebotenen Volltextsuche.

### 2.1.5. Hochschule Neubrandenburg

Hochschule Neubrandenburg  
University of Applied Sciences

 Fachbereich Bauingenieur- und Vermessungswesen  
Studiengang Geoinformatik

Praxissemester - Praktikumsbetriebe		
<ul style="list-style-type: none"> <li>• <a href="#">Deutsches Zentrum für Luft- und Raumfahrt (DLR); Berlin</a></li> <li>• <a href="#">Infoterra GmbH; Friedrichshafen</a></li> <li>• <a href="#">Geoware AG; Neubrandenburg</a></li> <li>• <a href="#">GIS - Regionalsysteme GmbH; Belling</a></li> <li>• <a href="#">Deutsche Bahn; DB Netz; Leipzig</a></li> <li>• <a href="#">Geothermie Neubrandenburg GmbH; Neubrandenburg</a></li> <li>• <a href="#">Daberkower Landhof AG</a></li> <li>• <a href="#">ARC-Ingenieure Berlin</a></li> <li>• <a href="#">Stadtverwaltung Eggesin</a></li> <li>• <a href="#">weitere Kontakte über Professoren</a></li> </ul>		
<b>Praktikumsbetrieb</b>	<b>Deutsches Zentrum für Luft- und Raumfahrt (DLR)</b> Rutherfordstr. 2 12489 Berlin	
<b>Kontakt</b>	Dipl.-Phys. Dr. Richard Woessler IVF German Aerospace Center Rutherfordstr. 2 12489 Berlin Phone: (+49) 30 670 55-230 Fax: (+49) 30 670 55-202	Prof. Dr.-Ing. A. Wehrenpfennig      Phone: (+49) 395 5693 323 email: <a href="mailto:wehrenpfennig@fh-nb.de">wehrenpfennig@fh-nb.de</a>
<b>Aufgabenstellung</b>	Unterstützung bei der Weiterentwicklung einer Echtzeit-3-d-modellbasierten automatischen Fahrzeugerkennung aus Videodaten durch Tests und Ergebnisauswertungen, optional mit Programmieraufgaben  Falls keine Programmierkenntnisse vorhanden wären, bestünde die Aufgabe darin, vorhandene Programme zu testen und die Tests in einfacher Form auszuwerten. Für diese	



Abbildung 5: <http://www.hs-nb.de/geoinf/lehre/praxis/betriebe/index.html> (09.09.2008)

Bisher konnten sich Studenten des Fachgebietes Geoinformatik über Praktikumsstellen informieren, indem sie die tabellarische Auflistung von Praktikumsbetrieben und bisherigen Praktika (2003, 2004) auf der Webseite der Hochschule nutzten.

Zusammengefasst sollte eine Praktikumsbörse bzw. Praktikantenbörse dem Nutzer in erster Linie die Informationen zur Verfügung stellen, die dieser benötigt um effektiv nach einer Praktikumsstelle zu recherchieren. Dies erfüllen weitestgehend alle herausgesuchten Beispiele. Wird ein passendes Ergebnis gefunden, werden detaillierte Informationen über Thema und Firma bereitgestellt und eventuell ein Link zur Firmenpräsentation im Internet. Worin sich Praktikantenbörsen wesentlich unterscheiden, ist die Art der Suche (Katalogsuche, Volltextsuche, Schlagwortsuche) und vor allem wonach gesucht werden kann. Von einer guten Suche hängt der Erfolg einer Praktikantenbörse jedoch maßgeblich ab.

Bei fast allen Beispielen wurde die Möglichkeit zur Volltextsuche gegeben, jedoch keine der Praktikantenbörsen hat eine Schlagwortsuche angeboten. Das wirft die Frage auf, wieso keine Schlagwortsuche verwendet wurde. Weiterhin gibt dies Anlass zu einem kurzen Vergleich von Volltext- und Schlagwortsuche.

### **Pro und Contra Volltextsuche**

Beim Ausführen einer Volltextsuche werden alle Ergebnisse angezeigt, in denen die eingegebene Zeichenfolge auftaucht.

Pro	Contra
– gibt jede Information zurück, in der die gesuchte Zeichenkette vorhanden ist	– Suche dauert länger, da alle Einträge durchsucht werden
– Suchergebnisse können gezielt eingegrenzt werden, durch Eingabe mehrerer Suchbegriffe	– unter Umständen auch unwichtige, unpassende Ergebnisse
– zunächst schnelle, einfache Lösung (für Entwickler)	

Tabelle 1: Pro und Contra Volltextsuche

### **Pro und Contra Schlagwortsuche**

Beim Ausführen einer Schlagwortsuche werden genau die Ergebnisse angezeigt, denen das Schlagwort zugeordnet wurde. Dazu muss vorher eine Verschlagwortung der Informationen durchgeführt werden.

Des Weiteren ist zu klären, ob die Schlagwortsuche über einen Schlagwortkatalog oder durch eine freie Verschlagwortung durchgeführt werden soll.

In einem Schlagwortkatalog werden die Wörter, die als Schlagworte dienen sollen, zentral festgelegt. Diese Form der Schlagwortsuche bietet den Vorteil, dass die Schlagworte eindeutig sind, da sie bewusst ausgesucht und zugeordnet werden. Die Verschlagwortung wird mit der steigenden Anzahl der zugeordneten Schlagworte detaillierter.

Im Gegensatz zu einem Katalog lässt die freie Verschlagwortung viel Freiraum bei der Wahl der Schlagworte. Das hat den Vorteil, dass die Zuordnung individuell sein kann. Doch im gleichen Zuge bringt dies einen großen Nachteil mit sich. Die Individualität kann zu gleich dazu führen, dass die Verschlagwortung an Eindeutigkeit verliert, durch Synonyme einiger Wörter oder durch eingefügte Schreibfehler. [2]

Pro	Contra
– weniger Treffer, aber von höherer Qualität	– höhere Vorarbeit, da Schlagwort gut gewählt sein muss (Schlagworte sind nur dann gut gewählt, wenn sie von den Benutzern auch als Suchbegriffe verwendet werden)
– schnelle Suche	– geringe Bandbreite (es können nie alle Schlagworte/Suchbegriffe vergeben werden, die theoretisch möglich sind)
– Inhalte werden gezielt indexiert	

Tabelle 2: Pro und Contra Schlagwortsuche

Volltext- wie auch Schlagwortsuche haben Vor- und Nachteile. Für welche Suche sich ein Entwickler bei seinem eigenen Projekt entscheidet hängt davon ab, wo dieser seine Prioritäten setzt. Jedoch muss diese Entscheidung bereits vorher feststehen, da die jeweilige Suchart den Entwurf des Systems beeinflusst.

Im Falle der zu entwerfenden Praktikumsbörse wurde sich für eine Schlagwortsuche entschieden, um dem Nutzer eine komfortable Suche zu bieten.

## **2.2. Technologie der Webseite der Hochschule**

Die Praktikumsbörse soll in die Webpräsentation des Studiengangs Geoinformatik der Hochschule Neubrandenburg integriert werden. Aus diesem Grund ist es notwendig die Technologie der Webseite der Hochschule zu betrachten. Die Einbindung kann über einen Link zur Praktikumsbörse geschehen.

Die Webseite der Hochschule Neubrandenburg beruht auf dem Open-Source-Web-Content-Management-System (CMS) TYPO3.

„Content Management Systeme (Redaktionssysteme) dienen der Verwaltung von Inhalten und ermöglichen es dem Nutzer ohne Programmierkenntnisse oder Kenntnis von HTML oder XML Inhalte zu verwalten. Content Management Systeme trennen Inhalt (Content), Datenstruktur und Design (Layout) voneinander.“ [3]

TYPO3 wird vom Webserver betrieben und vom Browser gesteuert. Es basiert auf der serverseitigen Scriptsprache PHP (TypoScript). Als Datenbank kann MySql, PostgreSQL und Oracle eingesetzt werden. Bei CMS wird generell von zwei Bereichen gesprochen, dem Frontend (FE) und dem Backend (BE). Das Frontend stellt im Prinzip die Webseite, also die reguläre Internetpräsentation dar. Im Backend hingegen wird die Präsentation erstellt und gepflegt. [4]

## **2.3. Google Maps**

„Die Jahre 2005 und 2006 werden als jene Jahre in die Internet-Geschichte eingehen, in denen die Online-Erstellung von Karten schließlich erwachsen wurde. Vor 2005 konnte man zwar über MapQuest und andere Dienste Wegbeschreibungen abrufen und nach Orten und Unternehmen suchen, aber diese Möglichkeiten waren begrenzt. Sie beschränkten sich meist auf Partnerunternehmen und die generelle Suche war üblicherweise nicht möglich. Am 8. Februar 2005 konnte Google diese Situation ändern. Wie bei vielen seiner Dienste stellte Google die Beta-Version von Google Maps still und leise zunächst über eine Labor-Adresse (<http://labs.google.com>) der Öffentlichkeit zur Verfügung und vertraute darauf, dass Mundpropaganda den neuen Dienst bekannt machen würde.“ [8, S. 11]

Google Maps umfasst einen Dienst der Google Inc., mit dem es möglich gemacht wird Orte, Straßen, Gebäude und andere Objekte zu suchen und deren Position dann auf einer

Karte, einem Satelliten- oder einem Luftbild der Erdoberfläche angezeigt wird. Außerdem können diese Karten bzw. Bilder durch den Anwender verschoben, vergrößert oder verkleinert werden. Google Maps operiert auf der gleichen Datenbasis wie das Programm Google Earth und stellt die Satellitenbilder und Landkarten über ein Webinterface bereit. Seit April 2006 ist die offizielle deutsche Version verfügbar. Bis auf die letzte Zoomstufe sind die Karten der deutschen Google Maps genauso informativ wie das Angebot in den USA und Großbritannien [5].

Google Maps basiert auf der AJAX-Technologie (Asynchronous Javascript and XML ). Somit eignet sich Google Maps besonders gut für das Integrieren von Karten in eigenen Webprojekten. Die Funktionen der Google Maps können mit Hilfe einer Programmierschnittstelle (API=*Application Programming Interface*) auf der eigenen Seite eingefügt und individuell angepasst werden. Außerdem lassen sich die Funktionen mit einer eigenen Anwendung kombinieren, um einen neuen Service anzubieten (Mashup).[5]

### **2.3.1. Google Maps API**

Google Maps API ist eine Programmierschnittstelle und ermöglicht die Integration von Google Maps in eine eigene Internetanwendung. Es kann eine eigene Karte mit eigenen Funktionen in der eigenen Homepage eingesetzt werden. Dazu ist ein API-Key notwendig, der kostenlos auf der Google Maps API Homepage über eine Registrierung zu beziehen ist. Um Google Maps zu nutzen wird der Key für die angegebene Domain generiert.

Da Google Maps-API auf der für das Web 2.0 typischen AJAX-Technologie basiert, wird Javascript sowie DHTML eingesetzt. Um eine eigene, individuelle, funktionale und ansprechende Karte zu erstellen, bietet die Google Maps-API eine zweckmäßige Bibliothek mit Objekten und Methoden an, die frei genutzt werden können. In Folge der asynchronen Datenübertragung zwischen Server und Browser muss beim Navigieren in der Karte nicht jedes Mal die komplette Seite neu geladen werden. [6]

## **2.4. Geocoding**

Allgemein betrachtet, wird unter Geocoding oder Geokodierung ein Vorgang verstanden, mit dem Daten, wie Bilder oder Adressen, mit Koordinaten versehen werden können.

Genau von diesem Vorgang soll auch in der Praktikumsbörse Gebrauch gemacht werden, um die Praktikumsstandorte in einer Karte einzeichnen zu können.

Besonders in Bezug auf Google Maps ist unter Geocoding ein Prozess zur Umwandlung von Adressen in geographische Koordinaten (Länge und Breite) zu verstehen. Die Koordinaten können dazu verwendet werden, Markierungen zu setzen oder die Karte in Position zu bringen. Die Programmierschnittstelle Google Maps API schließt diesen Geocoding Service ein. Dieser Dienst kann direkt über eine http-Anfrage oder die Verwendung eines GclientGeocoder Objekts erreicht werden. Dabei ist jedoch zu beachten, dass Geocoding viel Zeit und viele Ressourcen in Anspruch nimmt. Aus diesem Grund sollten die Adressen, wenn möglich vorher geocodiert und die Resultate z.B. in einer Datenbank abgelegt werden. [7]

Um die Dienste nutzen zu können, benötigt der Nutzer einen Webserver mit ständiger Internetverbindung, der auf die entsprechenden Dienste zugreifen kann. Mit Hilfe des Google-Dienstes sollen die Koordinaten für Straßen in den USA, Kanada, Frankreich, Italien, Deutschland und Spanien genau ermittelt werden können. Vor dem Juni 2006 gab es keinen offiziellen Geocoder von Google. Zu dieser Zeit wurde der in die Webseite maps.google.com eingebaute Geocoder verwendet und die Antworten vom Bildschirm übernommen. Doch dies war eine ausdrücklich nicht autorisierte Nutzung des Dienstes. Doch dies führte unter anderem dazu, dass Dienste entstanden, die dieses Feld abdecken wollten. [8, S. 81]



## 3. Verwendete Software und Stand der Technik

### 3.1. Verwendete Software

Die Praktikumsbörse soll in die Webpräsentation des Studiengangs Geoinformatik der Hochschule Neubrandenburg eingebunden werden. Diese beruht auf dem CMS TYPO3. In TYPO3 wird mit TypoScript gearbeitet, dessen Grundfunktionen auf PHP beruhen. Damit stehen die Programmiersprache und die Datenbank, mit denen gearbeitet werden muss, bereits fest. Die Programmiersprache ist PHP und die Daten werden dementsprechend in einer MySql-Datenbank abgelegt, da diese von TYPO3 unterstützt wird und gut mit PHP zusammenarbeitet. Während der Entwicklung der Praktikumsbörse wurde folgende Software verwendet.

Xampp für Windows Version 2.5

Dabei handelt es sich um eine Zusammenstellung von freier Software, darin enthalten ist der **A**pache Webserver mit der Datenbank **M**ysql und den Skriptsprachen **P**erl und **P**HP. Das **X** in Xampp steht für die verschiedenen Betriebssysteme, auf denen es eingesetzt werden kann.

PhpMyAdmin- 2.11.7

Dies ist ein Werkzeug, dass von Xampp zusätzlich zur Verfügung gestellt wird. Es ist eine in PHP geschriebene Open-Source-Anwendung. PhpMyAdmin bietet eine grafische Benutzeroberfläche, die sich über einen Browser bedienen lässt. Damit lassen sich MySql-Datenbanken einfacher verwalten.

PHP Editor 2.22 - Beta 2

In einem PHP Editor lässt sich über die Verbindung von PHP mit einer textbasierte Auszeichnungssprache (HTML) eine angepasste Nutzerschnittstelle zum Datenbank Management System MySql programmieren.

ArgoUML Version 0.24

ArgoUML ist ein freies, plattformunabhängiges UML-Werkzeug. Unter anderem können damit Beschreibungen und Modellierungen von Systemen vorgenommen werden.

DBDesigner 4.0.5.6 Beta

Der DBDesigner ist ein Werkzeug, das einen Softwareentwickler beim Design und der Implementierung von Datenbanken unterstützt. Dabei ist der DBDesigner auf die Modellierung von MySql-Datenbanken spezialisiert.

### 3.2. Stand der Technik

Es gibt bereits zahlreiche Anwendungen im Web, die ebenfalls darauf basieren, die vom Google Maps API Dienst gelieferten Daten mit ihren Daten zu kombinieren. Für dieses Vermischen von Daten hat sich bereits der Begriff „Mashup“ etabliert. Die Möglichkeiten für praktische Anwendungen sind schließlich auch nahezu unbegrenzt. Es können auf einer Karte eigene Markierungen gesetzt und gespeichert werden, aber auch Routenplanungen und Einiges mehr ist möglich. Einige Beispiele für Webseiten, in denen Google Maps API Funktionen mit eigenen Anwendungen kombiniert werden sind:

<http://www.brennende-autos.de/> (11.09.2008): Auf dieser Webseite wird mit Hilfe einer Google Map dargestellt wo in Berlin bereits Autos abgebrannt sind.

<http://www.weltweit-kostenlos-geld-abheben.de/> (11.09.2008): Diese Seite wird von der Deutschen Kreditbank AG angeboten. Dort kann Jeder, der im Auslandsurlaub schon einmal Geld abheben musste, seine Erfahrungen dazu berichten. Auf der Google Map wird eingezeichnet, wo der Nutzer Geld abgehoben hat.

<http://www.suchr.de/> (11.09.2008): Wer sich auf dieser Seite registriert und einloggt, kann seine eigenen Wunschplätze hinzufügen und dazu Beschreibungen eintragen.

<http://www.booking.com> (11.09.2008): Diese Webseite bietet den Nutzern, die nach Hotels suchen an, sich sofort auf einer Karte von Google Map einen Überblick über die Lage der Hotels, sowie Sehenswürdigkeiten und Flughäfen zu verschaffen.

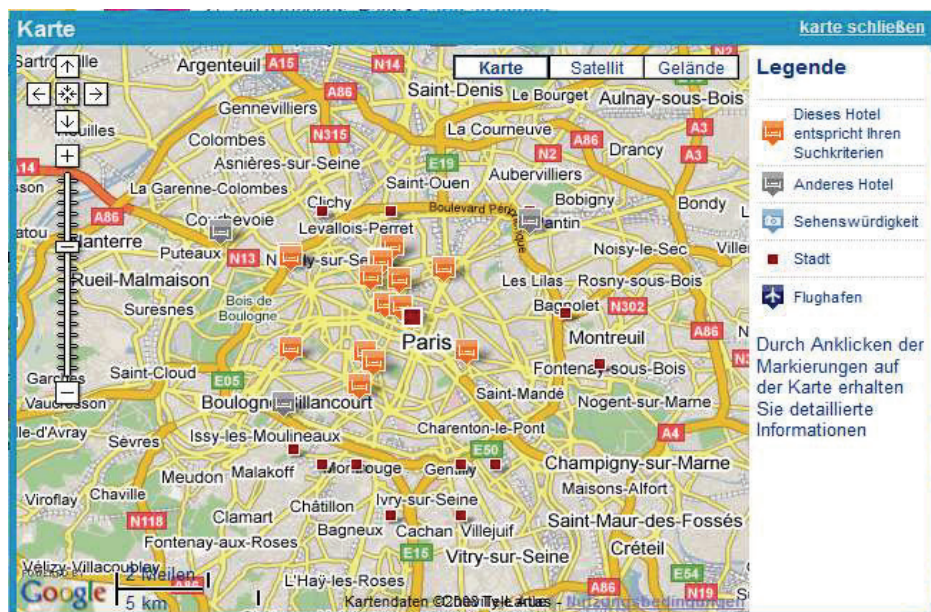


Abbildung 6: [www.booking.com](http://www.booking.com) (11.09.2008)

## 4. Planung und Entwicklung

### 4.1. Funktionalitäten der Praktikumsbörse

Zu Beginn der Entwicklung eines neuen Systems muss zunächst geklärt werden, was das System leisten soll. Im Falle einer Praktikumsbörse ist eine Funktionalität unabdingbar und zwar das Suchen und Bereitstellen von Praktikumsangeboten. Des Weiteren soll diese spezielle Praktikumsbörse nicht nur Angebote in Tabellenform wiedergeben, sondern deren Praktikumsort auf einer Karte visualisieren. Um die Praktikumsbörse stets aktuell zu halten, muss es möglich gemacht werden neue Angebote einzustellen. Dies wird über ein geeignetes Webinterface umgesetzt. Die in der MySQL-Datenbank verwalteten Daten beinhalten Informationen über das Praktikumsangebot, die Ansprechpartner in Betrieb und an der Hochschule sowie den Praktikanten, der in dem Betrieb bereits ein Praktikum angetreten hat. Die Daten des Praktikanten werden für eine weitere Funktionalität genutzt, und zwar um z.B. Praktikumsberichte, mit dem Vermerk anerkannt bzw. nicht anerkannt, zu archivieren. Somit ist die Praktikumsbörse nicht nur für Studenten und Dozenten interessant, sondern auch für den Praktikumsbeauftragten und das Prüfungsamt. Dem Praktikumsbeauftragten wird es möglich gemacht, Praktikumsberichte oder auch Praktikumsverträge der Studenten zu archivieren und das Prüfungsamt könnte Interesse daran haben, zu erfahren welcher Student das Praxissemester mit anerkannt bzw. nicht anerkannt abgelegt hat.

Eine spezielle Funktionalität dieser Praktikumsbörse ist die Visualisierung der Praktikumsorte. Dies soll realisiert werden, indem für jedes Ergebnis einer Suchanfrage ein Marker (eine Markierung) auf einer Google Map zu finden ist. Um es dem Nutzer etwas übersichtlicher zu machen, soll an der Einfärbung der Marker zu erkennen sein, welche Angebote aktuell und welche bereits länger als ein Jahr in der Praktikumsbörse vorhanden sind. Des Weiteren soll die Karte bei jeder Suche soweit herangezoomt werden, dass sich die Markierungen innerhalb eines minimal umschließenden Rechtecks (Bounding Box) befinden. Die über das Webinterface eingegebenen Praktikumsangebote werden geocodiert und die daraus zur Verfügung stehenden Längen- und Breitenangaben in eine Tabelle der MySQL-Datenbank gespeichert. Dabei ist darauf zu achten, dass zuerst die Praktikumsangebote eingefügt und später die Angaben zu den Praktikanten hinzugefügt werden müssen.

Mit Hilfe eines Use-Case-Diagramms (Abb. 7) soll das externe Verhalten eines Systems

```
graph TD
    subgraph System
        direction TB
        UC1([Suche nach Praktikumsangeboten])
        UC2([Anmeldung])
        UC3([Suche nach Praktikumsangeboten])
        UC4([Visualisierung der Praktikumsorte (beschränkte Informationen)])
        UC5([Anmeldung])
        UC6([Suche nach Praktikumsangeboten])
        UC7([Visualisierung der Praktikumsorte])
        UC8([Anmeldung])
        UC9([Suche nach Praktikumsangeboten])
        UC10([Suche nach Praktikanten])
        UC11([Angaben zu Praktikanten anzeigen])
        UC12([Anmeldung])
        UC13([Suche nach Praktikanten])
        UC14([hinzukommende Daten einpflegen])
        UC15([Anmeldung])
        UC16([Stammdatenpflege])
    end

    subgraph Actors
        direction TB
        A1((Gast))
        A2((Student))
        A3((Dozent))
        A4((Prüfungsausschuss))
        A5((Praktikumsbeauftragter))
        A6((Administrator))
    end

    subgraph External_Systems
        direction TB
        ES1[MySQL-Datenbank]
        ES2[LDAP-Server]
    end

    A1 -- 1 --> UC1
    UC1 --> UC4

    A2 -- 1 --> UC2
    UC2 --> UC3
    UC2 -.-> ES1
    UC2 -.-> ES2
    UC3 --> UC7

    A3 -- Login --> UC5
    UC5 -- Zugriff --> UC6
    UC5 -.-> ES1
    UC5 -.-> ES2
    UC6 --> UC7

    A4 -- 1 --> UC8
    UC8 -- Zugriff --> ES1
    UC8 --> UC10
    UC10 --> UC11

    A5 -- Login --> UC12
    UC12 -- Zugriff --> ES1
    UC12 --> UC13
    UC12 --> UC14

    A6 -- 1 --> UC15
    UC15 -- Zugriff --> ES1
    UC15 --> UC16
```

Im Folgenden werden die Akteure des Use-Case-Diagramms (Abb. 7) beschrieben.

Student: Unter dem Benutzertyp Student ist ein Studierender der Hochschule Neubrandenburg zu verstehen. Dieser kann sich mit seinem hochschulweiten Benutzerkennzeichen anmelden. Auch der Student kann nach

Praktikumsangeboten suchen, mit dem Unterschied, dass dieser einige weitere Informationen über den jeweiligen Praktikumsplatz erhält.

Dozent: Der Dozent kann sich ebenfalls mit seiner hochschulweiten Benutzerkennung einloggen. Er hat die Möglichkeit nach Praktikumsangeboten zu recherchieren und anders als ein Gast oder der Student, sieht er zusätzliche Informationen, wie z.B. die Betriebsbewertung, da diese Information nicht für jeden zugänglich sein soll.

Prüfungsamt/Prüfungsausschuss: Im Gegensatz zu den Benutzern Gast, Student und Dozent, wird dem Prüfungsamt die Möglichkeit geboten nach Praktikanten zu suchen, um insbesondere Informationen über anerkannte oder nicht anerkannte Praktika zu erhalten. Aus diesem Grund muss auch bei dieser Benutzergruppe eine Anmeldung erfolgen, da diese Informationen nicht für jeden zugänglich gemacht werden dürfen.

Praktikumsbeauftragter: Der Praktikumsbeauftragte spielt eine besondere Rolle, denn dieser muss für die Aktualität der Praktikumsbörse sorgen und neu hinzukommende Praktikantenstellen über ein Formular eintragen. Aber auch die Zuordnung der Praktikanten zu den Praktika muss er vornehmen. Hierfür bietet ihm das System die Möglichkeit die Praktikumsberichte und die Praktikumsverträge der Studenten zu archivieren. Die Suche nach Praktikanten liefert als Ergebnis Angaben über diesen, den Praktikumsvertrag, sowie den Praktikumsbericht mit der Information anerkannt oder nicht anerkannt.

Administrator: Der Administrator ist für die Pflege des Systems bzw. der Stammdaten und die Datensicherung zuständig. Seine Anmeldung läuft nicht über den LDAP-Server, damit es möglich ist auch bei einem Ausfall des LDAP-Servers auf das System zu zugreifen.

Auf Grund der Integration der Nutzerverwaltung der Hochschule Neubrandenburg durch LDAP ist es den Studenten und Dozenten, dem Prüfungsamt und Praktikumsbeauftragten möglich sich über die hochschulweite Nutzerkennung anzumelden. Es ist jedoch notwendig in einer Benutzertabelle der Praktikumsdatenbank die speziellen Rechte der Dozenten, des Prüfungsamtes, des Praktikumsbeauftragten und des Administrators zu verwalten. Aus diesem Grund wird bei der Anmeldung einer dieser vier Nutzer zunächst auf die MySql-Datenbank zugegriffen, um zum einen die Rechte und zum anderen die Anmeldung über LDAP oder lokal über ein Passwort in der MySql-Datenbank zu klären.

## 4.2. Datenbankentwurf

Bisher wurde geklärt, was das zukünftige System leisten soll und welche Akteure dabei eine Rolle spielen. Im nächsten Schritt ist zu klären, wo und in welcher Form die zum System benötigten Daten gespeichert werden sollen. Das wo wurde bereits angesprochen. Die Daten werden in einer MySql-Datenbank verwaltet.

Der erste Schritt zur Entwicklung einer Datenbank besteht darin, festzustellen, welche Informationen die Nutzer vom späteren System erwarten und somit, welche Informationen gespeichert werden sollen. Dieser Schritt nennt sich Datenmodellierung. In Folge dieser Erhebung wird festgelegt, welche Tabellen benötigt werden. Parallel dazu muss bestimmt werden, welche Datentypen für die einzelnen Tabellenspalten erforderlich sind.

Hierfür wurde zunächst eine einfache Auflistung möglicher Informationen zu einer Praktikumsstelle vorgenommen und diese in Tabellenform umgewandelt.

Beispiel eines Datensatzes zu einer Praktikumsstelle:

Praktikumsbetrieb:	DLR Institut für Planetenforschung, Berlin- Adlershof
Webseite:	<a href="http://www.dlr.de">www.dlr.de</a>
Standort:	Berlin-Adlershof Rutherfordstraße 2 12489 Berlin, Deutschland
Aufgabenstellung:	Von 1997 bis 2005 befand sich der amerikanische Satellit Mars Global Surveyor auf einer Umlaufbahn um den Mars. Von 1997 bis 2001 war das Bordinstrument Mars Orbiter Laser Altimeter (MOLA) aktiv und sammelte über 600 Millionen Entfernungsmessungen zur Marsoberfläche. Die gespeicherten Parameter jedes einzelnen Laser-Schusses enthalten neben der Entfernungsmessung auch Information über die Amplitude und die Verbreiterung des reflektierten Laserstrahls, welche Aufschluss über die Rauigkeit der Marsoberfläche gibt. Die Praktikumsaufgabe besteht darin, einen vorgegebenen Algorithmus programmiertechnisch umzusetzen, so dass Rauigkeitswerte aus dem MOLA- Datensatz berechnet werden können. Anschließend ist in ARCGIS eine Rauigkeitskarte des Mars zu erstellen. Diese soll Aufschluss über die Oberflächenbeschaffenheit von potenziellen Landestellen auf dem Mars geben.

Ansprechpartner Betrieb: M. Wählich,  
Abteilung Planetengeodäsie  
Institut für Planetenforschung  
Deutsches Zentrum für Luft- und Raumfahrt Berlin- Adlershof  
Telefon: 030-67055-348,  
E-mail: marita.waehlich@dlr.de

Ansprechp. Hochschule: Prof. Dr.-Ing. A. Wehrenpfennig  
Phone: (+49) 395 5693 323  
email: wehrenpfennig@fh-nb.de

von-bis: März 2008 – Juli 2008

Studiengang: Geoinformatik

Praktikant: Heinrich Unbekannt

Betriebsbewertung: Empfehlung: Wissenschaft, große Selbständigkeit, Bachelorthema

Bei der Betrachtung einiger solcher Datensätze, die alle für die Datenbank relevanten Attribute enthalten, entwickelt sich über die Normalisierung nach und nach eine relationale Datenbank. Die Vorgehensweise der Normalisierung hat einige Vorteile. Durch eine Normalisierung werden unerwünschte Anomalien (Einfügeanomalie, Löschanomalie, Änderungsanomalie) sowie überflüssige Informationen (Redundanzen) vermieden. Außerdem zwingt die Normalisierung zu einem systematischen Entwurf einer Datenbank, was zu einer besseren Übersichtlichkeit für Benutzer und Entwickler führt.

## 1. Normalform

„Eine Relation befindet sich in der ersten Normalform, wenn alle ihre Attribute nur einfache Attributwerte aufweisen, d.h. wenn jeder Kreuzungspunkt von Zeile und Spalte höchstens einen Wert besitzt. (Bezeichnung: atomar)“ [9]

Im Folgenden ist ein beispielhafter Teil der unnormalisierten Relation (Tab.3) zu sehen, um das Prinzip zu verdeutlichen. Um diese Tabelle in die erste Normalform zu überführen müssen Attribute hinzugefügt und andere dafür gelöscht werden.

Praktikumsbetrieb	Webseite	Standort	Ansprechpartner Hochschule
DLR Institut für Planetenforschung	www.dlr.de	Berlin-Adlershof Rutherfordstraße 2 12489 Berlin Deutschland	Prof. Dr.-Ing. A. Wehrenpfennig phone: (+49) 395 5693 323 email: wehrenpfennig@fh-nb.de
Infoterra GmbH Corporate Development / R2C	www.infoterra-global.com	88039 Friedrichshafen	Prof. Dr.-Ing. A. Wehrenpfennig phone: (+49) 395 5693 323 email: wehrenpfennig@fh-nb.de

Tabelle 3: unnormalisierte Relation

Die Tabelle der 1. Normalform (Tab.4) besitzt anstelle des Attributes Standort nun PLZ, Ort, Straße (nicht dargestellt) und Land. Statt Ansprechpartner Hochschule besitzt diese Relation die zusätzlichen Spalten Ansprechpartner, Email und Telefon.

Prakt.betrieb	Webseite	PLZ	Ort	Land	Ansprechp.	e-mail	Telefon
DLR Institut für Planetenforschung	www.dlr.de	12489	Berlin	Deutschland	Prof. Dr.-Ing. A. Wehrenpfennig	wehrenpfennig@fh-nb.de	(+49) 395 5693 323
Infoterra GmbH Corporate Development / R2C	www.infoterra-global.com	88039	Friedrichshafen	Deutschland	Prof. Dr.-Ing. A. Wehrenpfennig	wehrenpfennig@fh-nb.de	(+49) 395 5693 323

Tabelle 4: 1. Normalform

## 2. Normalform:

„Eine Relation befindet sich in der zweiten Normalform, wenn sie in der ersten Normalform ist und jedes Nicht-Schlüssel-Attribut vom gesamten Identifikationsschlüssel funktional abhängig ist, nicht aber bereits von Teilen des Gesamtschlüssels.“ [9]

Um von der 1. Normalform (Tab.4) zu der 2. Normalform zu gelangen, muss zunächst ein Schlüssel für die Relation festgelegt werden. Falls dieser Schlüssel nur aus einem Attribut besteht, so liegt bereits die 2. Normalform vor. In diesem Fall wird ein zusammengesetzter Primärschlüssel aus Hochschule\_ID und Betrieb\_ID festgelegt. Als nächstes muss nun die



Relation auf Teilschlüsselabhängigkeit untersucht werden, d.h. folgt aus einem Schlüsselattribut bereits weitere Nichtschlüsselattribute. Somit lässt sich feststellen, dass aus Hochschule\_ID Ansprechpartner, Email und Telefon und aus Betrieb\_ID Praktikumsbetrieb, Webseite, PLZ, Straße, Ort und Land folgen. Es wird nun eine neue Relation gebildet, die den Teilschlüssel, der nun Primärschlüssel der neuen Relation wird und die allein von ihm funktional abhängigen Nichtschlüsselattribute enthält. Drei Relationen sind entstanden, zum einen Relation ‚Hochschule‘, Relation ‚Betrieb‘ und Relation ‚Kontakt‘, aus den beiden Primärschlüsseln der Relation die nun Fremdschlüssel sind.

### 3. Normalform:

„Eine Relation befindet sich in der dritten Normalform, wenn sie in der zweiten Normalform ist und die Nichtschlüsselattribute voneinander nicht funktional abhängig sind, d.h. es existieren keine transitiven Abhängigkeiten.“ [9]

Um die Relationen in die 3. Normalform zu überführen, muss jede Relation auf transitive Abhängigkeit untersucht werden, d.h. folgt aus einem Nichtschlüsselattribut ein weiteres Nichtschlüsselattribut eindeutig. In der Relation ‚Betrieb‘ folgt aus Betrieb\_ID Praktikumsbetrieb, Webseite, PLZ, Straße, Ort und Land. Aber aus Straße folgt bereits PLZ, Ort und Land. Aus diesem Grund wird eine neue Relation ‚Standort‘ gebildet, die die transitiv abhängigen Attribute enthält. Der Primärschlüssel dieser Relation heißt Standort\_ID.

Es gibt derzeit 5 Normalformen, doch bei einer Normalisierung bis zur 5. Stufe werden die Tabellen meist zu sehr auseinander gerissen. Außerdem wären für das Zurückziehen der Daten zu viele Schnittstellen nötig, was einen Einfluss auf die Leistungsfähigkeit haben würde. [9]

Im Folgenden ist nun die Datenbank der Praktikumsbörse als ein gesamtes Entity-Relationship-Model (Abb.8) dargestellt. Es dient zum einen zur Überprüfung der Beziehungen und zum anderen zur Analyse und dem Design der Datenstrukturen. Es zeigt einen Ausschnitt der realen Welt und hilft somit in der Implementierungsphase als Grundlage für das Aussehen der Datenbank.

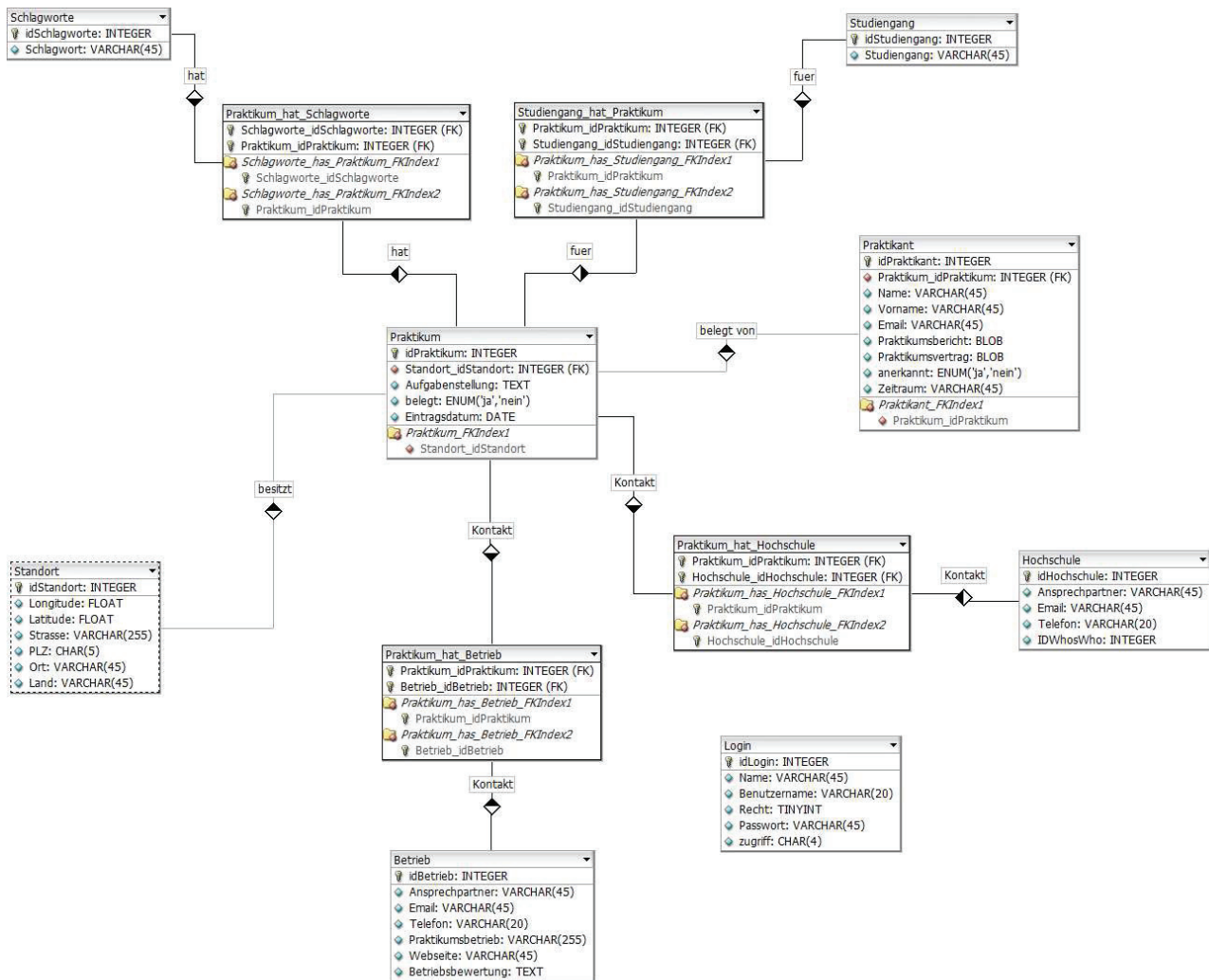


Abbildung 8: Entity-Relationship-Model

Es sind insgesamt 12 Tabellen entstanden, wobei 11 der Relationen in direkten Beziehungen stehen. Die „Login“-Tabelle ist nicht mit den anderen verbunden, da sie lediglich für die Anmeldung notwendig ist. Die Beziehungen wurden durch den Einsatz von Fremdschlüsseln gelöst. Die Tabellen mussten als InnoDB gespeichert werden, da nur diese Speicher-Engine die Überprüfung von Fremdschlüsselbeschränkungen unterstützt. Bei anderen Speicher-Engines verarbeitet der MySQL Server zwar die Syntax für Fremdschlüssel, verwendet oder speichert diese jedoch nicht [10].

Fremdschlüssel werden aus dem Grund eingesetzt, weil sie beim Entwickeln einer Datenbank einige Vorteile haben. So können sie es zum Beispiel erschweren Inkonsistenzen in die Datenbank zu bringen, vorausgesetzt ein guter Entwurf der

Beziehungen ist vorhanden. Der Datenbankserver übernimmt die Überprüfung der Aktionen, die zur Inkonsistenz führen könnten, somit muss dies auf Anwendungsseite nicht mehr getan werden. Außerdem wird damit verhindert, dass in verschiedenen Anwendungen nicht alle Konsistenzbedingungen in gleicher Weise geprüft werden. Dabei ist jedoch zu beachten, dass diese Vorteile die Leistung des Datenbankservers beeinträchtigen, da dieser die Überprüfung der Fremdschlüsselbeschränkungen durchführen muss. [10] Bei größeren Projekten wird aus diesem Grund auf Fremdschlüssel verzichtet. Doch im Falle der Praktikumsbörse ist dieser Nachteil vertretbar, da diese Datenbank keinen übermäßigen Umfang besitzt.

In dem Entity-Relationship-Model (Abb. 8) sind sowohl 1:n Beziehungen (Abb. 9) als auch n:m Beziehungen zu finden. Wobei die n:m Beziehungen nicht direkt umgesetzt werden, da diese zu komplex und in dieser Form nur schwer zu verwalten sind. Es wird stattdessen jeweils eine Verbindungstabelle (Pseudotabelle) eingefügt, die die beiden Primärschlüssel der zu verbindenden Tabellen als Fremdschlüssel enthält. Dabei wird die n:m Beziehung also aufgelöst und es entstehen aus einer n:m Beziehung zwei 1:n Beziehungen.

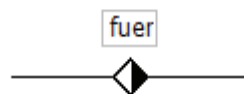


Abbildung 9: 1:n Beziehung

Die Beziehung „Kontakt“ zwischen den Relationen Praktikum und Betrieb ist eine n:m Beziehung. Jedes Praktikum oder vielmehr jede Aufgabenstellung hat einen oder mehrere Ansprechpartner im Betrieb. Andersherum besitzt ein Ansprechpartner im Betrieb eine oder mehrere Aufgabenstellungen, für die er zuständig ist.

Die Beziehung „Kontakt“ zwischen den Relationen Praktikum und Hochschule ist wiederum eine n:m Beziehung, denn jedes Praktikum oder Praktikumsangebot besitzt einen oder mehrere Ansprechpartner an der Hochschule als Kontaktperson. Aber auch ein Ansprechpartner an der Hochschule kann ein oder mehrere Praktika betreuen.

Die Beziehung „belegt von“ zwischen Praktikum und Praktikant ist eine 1:n Beziehung, weil ein Praktikum von einem oder auch mehreren Praktikanten belegt werden kann. Doch ein Praktikant kann zu einer Zeit nur ein Praktikum belegen.

Die Beziehung „fuer“ zwischen Praktikum und Studiengang ist eine n:m Beziehung. Ein

Praktikum wäre für einen oder mehrere Studiengänge denkbar und in einem Studiengang ist es möglich ein oder mehrere Praktika anzubieten.

Die Beziehung „hat“ zwischen Praktikum und Schlagworte ist ebenfalls eine n:m Beziehung, denn ein Praktikum kann ein oder mehrere Schlagworte haben und auch ein Schlagwort kann einem oder mehreren Praktika zugeordnet werden.

Die Beziehung „besitzt“ zwischen den Relationen Standort und Praktikum ist eine 1:n Beziehung, weil ein Standort einen oder mehrere Praktika besitzen kann, doch ein Praktikum genau einen Standort hat.

In dem Entity-Relationship-Model (Abb.8) sind nicht nur die Relationen und ihre Beziehungen untereinander dargestellt, sondern auch die Datentypen der einzelnen Spalten sind darin festgelegt. Bei der Auswahl der Datentypen ist zum einen darauf zu achten, dass speicherplatzsparend gearbeitet wird und zum anderen, dass auf Grund der Rechengeschwindigkeit Zahlen als numerische Typen und nicht als Text gespeichert werden.

Die Primär- und Fremdschlüssel in der Praktikumsbörse-Datenbank werden durch ganze Zahlen (INTEGER) gebildet.

Alles was unter Namen, Eigennamen und Email zu verstehen ist, wird als Zeichenkette gespeichert. Aber auch Telefonnummern, da diese außer den Zahlen auch Zeichen enthalten können und mit Telefonnummern nicht gerechnet werden muss. Da diese Felder in der Länge variieren, aber mit einem Maximalwert auskommen, wird hier der Typ VARCHAR(max) verwendet.

Die Longitude und die Latitude des Standortes sind Gleitkommazahlen, so dass hier nur der Datentyp FLOAT in Frage kommt.

Die Spalte der Postleitzahl wird mit CHAR(5) deklariert, da diese eine feste Länge von 5 Zeichen besitzt und nicht mit der Postleitzahl gerechnet werden muss.

Die Felder Aufgabenstellung und Betriebsbewertung werden mit dem Datentyp TEXT versehen. In den beiden Spalten ist es sehr wahrscheinlich, dass ganze Wortgruppen oder Sätze, also lange Werte, eingetragen werden.

Die Angaben zu belegt und anerkannt können als Typ ENUM gewählt werden, da hier nur ja oder nein von Interesse ist.

Das Eintragsdatum ist mit dem Datentyp DATE zu versehen, denn daraus soll berechnet werden, wie lange ein Angebot bereits in der Praktikumsbörse steht.

Der Praktikumsbericht und der Praktikumsvertrag sollen als PDF-Datei hochgeladen werden und müssen somit als Binärtyp gespeichert werden. Deshalb wird in diesem Fall der Datentyp BLOB gewählt, wobei zu überlegen ist, ob ein MEDIUMBLOB mit größerem Wertebereich nötig wäre.

## 5. Implementierung

In diesem Abschnitt der Arbeit wird die Realisierung des im vorangegangenen Kapitels beschriebenen Systementwurfs erläutert. Wie bereits erwähnt wurde dazu die Programmiersprache PHP verwendet. Das Ergebnis dieser Entwicklungsphase ist der Prototyp der Praktikumsbörse.

Gemäß den vorangegangenen Überlegungen zu den Funktionalitäten der Praktikumsbörse lässt sich das Programm in vier Hauptteile gliedern, und zwar der Anmeldung der verschiedenen Nutzergruppen, der Suche nach Angeboten, der Visualisierung der Praktikumsorte und der Eingabe neuer Praktikumsangebote und Praktikanten mit Geocodierung der Standorte. Außerdem werden die Daten in einer MySql-Datenbank abgelegt. Die Datenbank wird implementiert, bevor mit dem Aufbau des Webinterface begonnen wird.

### 5.1. *MySql-Datenbank*

Für die Implementierung einer Datenbank ist ein gutes Entity-Relationship-Model wichtig und von großem Vorteil, denn dort werden alle für die Realisierung notwendigen Aspekte aufgeführt. Das Model muss lediglich umgesetzt werden. Zudem wurde das ER-Model der Praktikumsbörse mit Hilfe des DBDesigners gefertigt, der auf die Modellierung von MySQL-Datenbanken spezialisiert ist und als Werkzeug die Synchronisation mit einer Datenbank anbietet. Dazu muss eine Datenbank in MySql angelegt sein und eine Verbindung mit dieser aufgenommen werden. Danach werden alle noch nicht bestehenden Tabellen erzeugt und zwar mit den Datentypen und den Beziehungen zwischen den Relationen aus dem ER-Model.

Um einen sicheren Zugriff auf die Datenbank über das Webinterface möglich zu machen, wurde ein Benutzer praktikb mit phpmyadmin angelegt. Der Benutzername und das dazugehörige Passwort wird benötigt, um auf die Datenbank „praktikumsboerse“ zu zugreifen.

Um mit der Datenbank arbeiten zu können, wurden schließlich einige Testdaten eingefügt.

## 5.2. Webinterface

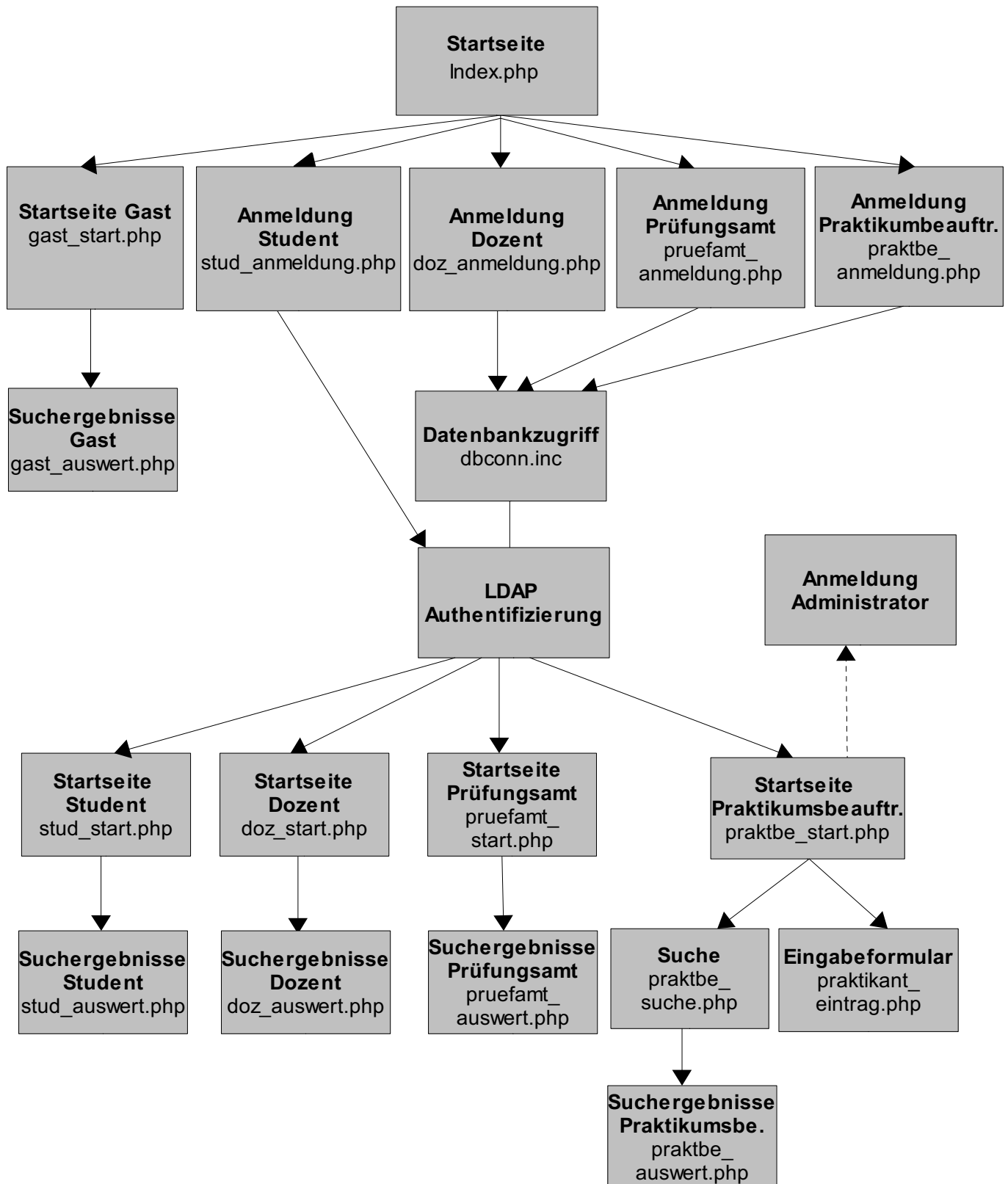


Abbildung 10: Auszug einiger PHP-Seiten der Praktikumsbörse

In der Abbildung 10 sind die für die Praktikumsbörse wichtigsten PHP-Seiten schematisch dargestellt. Eine Skizze dieser Art bietet zum einen eine große Hilfe während der Implementierung und zum anderen ist sie von großem Vorteil, wenn das System erweitert werden soll. Denn damit kann sich der Entwickler stets einen guten Überblick verschaffen. In den folgenden Abschnitten wird die Umsetzung der einzelnen Bereiche des Webinterface der Praktikumsbörse beschrieben. Dazu gehören Inhalte, die Darstellung der Ergebnisse sowie eine Beschreibung einiger aufgetretener Probleme.

### 5.2.1. Layout



Abbildung 11: Index-Seite der Praktikumsbörse

Jede Seite der Praktikumsbörse hat ein einheitliches Standard-Layout, welches sehr einfach gehalten wurde. Die Seiten sind, wie in Abbildung 10 zu sehen ist, in einen Kopfbereich, einen Fußbereich und dem dazwischen liegenden eigentlichen Inhalt eingeteilt.

Der Kopfbereich besteht aus einer eingefärbten Tabelle ohne Rahmen, die den Titel der Seite enthält. Der Inhalt, der zwischen Kopf- und Fußbereich steht, ist beliebig und auf jeder Seite ein anderer. Der Fußbereich dient der Navigation, zum Ausloggen und enthält



eine Information über die aktuelle Version und den Stand der letzten Aktualisierung. Das Datum der letzten Aktualisierung sollte dann verändert werden, wenn sich der Inhalt der Seiten wesentlich geändert hat. Mit dem „zurück“ Button gelangt der Nutzer auf die vorher aufgerufene Seite.

Das Aussehen und der Inhalt des Kopf- sowie des Fußbereichs werden in extra php-Dateien festgelegt, somit können diese in jeder neuen Seite einfach und schnell eingebunden werden. Dies gilt ebenso für die Festlegungen zu Hintergrundfarbe, Schriftfarbe, Schriftart, Schriftgröße, Farbe der besuchten und aktiven Links sowie die Höhe und Breite der Seiten, welche ebenfalls extra als CSS gespeichert werden.

Somit soll ein einheitliches Layout für alle Seiten der Praktikumsbörse erzwungen werden.

### 5.2.2. Anmeldung der Nutzer

Wie in Abbildung 11 bereits zu sehen ist, gibt es vier Benutzergruppen in der Praktikumsbörse, die sich anmelden müssen. Wenn der Nutzer als Student, Dozent, Prüfungsamt oder Praktikumsbeauftragter in die Praktikumsbörse gelangen möchte, kann er dies nur, indem er einen zulässigen Benutzernamen und ein Passwort angibt (Abb.12). In jedem anderen Fall gibt es aber auch die Möglichkeit ohne Anmeldung die Praktikumsbörse zu nutzen, mit dem Unterschied, dass nicht alle Daten freigegeben werden.

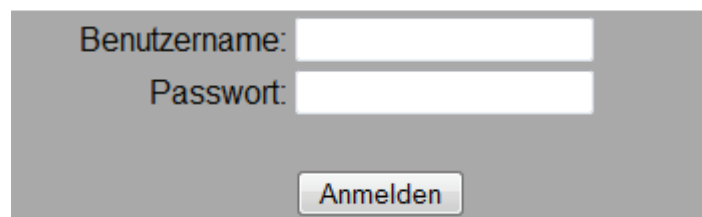


Abbildung 12: Anmeldung Praktikumsbörse

Die Anmeldung für Dozent, Prüfungsamt und Praktikumsbeauftragten verläuft etwa gleich. Diese Anmeldung ist in Abbildung 13 schematisch als Sequenzdiagramm dargestellt. Nach dem Eingeben des Benutzernamens und des Passwortes, wird zunächst in der MySQL-Datenbank eine Abfrage nach der Zugriffsart unter Verwendung des Benutzernamens gestellt. In der Login Tabelle wird dabei in der Spalte Zugriff die Zeile ausgewählt, in der

der eingegebene Benutzername steht. Als Ergebnisse kommen „man“ oder „ldap“ in Frage. In dem Fall, dass der Benutzername nicht in der Tabelle steht, wird ein „false“ (Wert 0) zurückgegeben. Doch ist der Benutzer in der Datenbank vorhanden und als Zugriff „man“ angegeben, so wird in der MySql-Datenbank über eine SQL-Abfrage kontrolliert, ob der Benutzer das richtige Passwort eingegeben hat. Ist hingegen als Zugriff „ldap“ zugewiesen, so werden Benutzername und Passwort über den LDAP-Server kontrolliert. In beiden positiven Fällen wird die Information über das Recht des Benutzers aus der MySql-Datenbank genommen um dies in einer Session-Variable zu speichern und damit zu arbeiten. Anhand des Wertes des Rechts lässt sich erkennen, auf welche Seite der Benutzer zugreifen darf. Die Session-Variable kann für einen einzelnen Benutzer während seines Besuches der Website (=Session) Werte speichern.

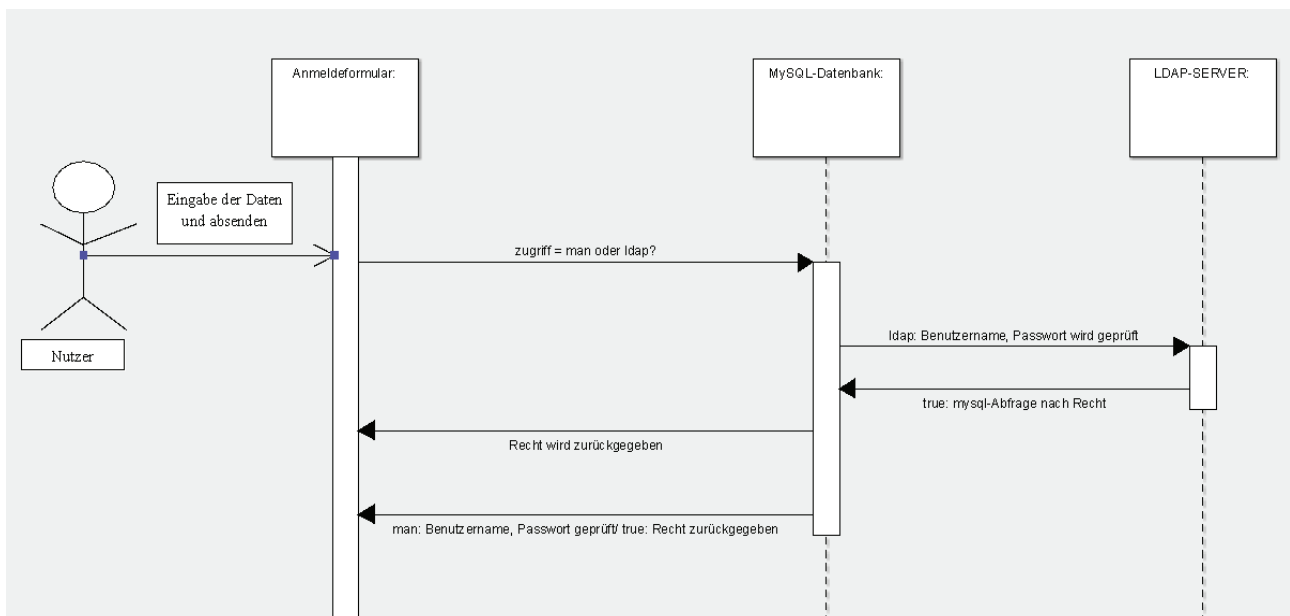


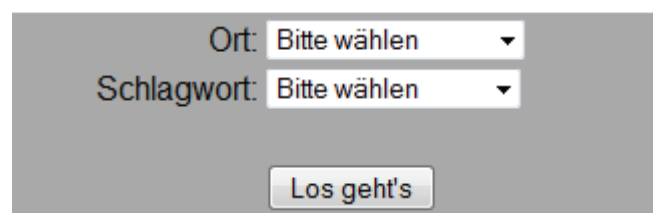
Abbildung 13: Sequenzdiagramm Anmeldung

Die Anmeldung des Studenten verläuft etwas anders. Dieser steht nicht in der MySQL-Datenbank. Aus diesem Grund werden Benutzername und Passwort sofort über den LDAP-Server geprüft. Doch auch ihm wird ein Wert für das Recht zugewiesen, und zwar die 1. Aus diesem Grund ist es wichtig, den in der Tabelle stehenden Nutzern nicht den Wert 1 zu zuweisen.

Die Anmeldung für den Administrator ist in Abbildung 10 bereits vorgesehen, doch wurde noch nicht umgesetzt. Diese Anmeldung sollte nicht auf der Startseite, sondern eine Ebene tiefer durchgeführt werden.

### 5.2.3. Suche nach Praktikumsangeboten

In den Grundlagen wurde sich bereits mit den Methoden der Suche auseinander gesetzt. Insbesondere sollte das eine Hilfe bei der Entscheidung zwischen Volltextsuche und Schlagwortsuche darstellen. Für die Praktikumsbörse wurde eine Schlagwortsuche umgesetzt. Dazu musste bereits im Entwurf berücksichtigt werden, dass eine Tabelle für Schlagwörter benötigt wird. Neben der Schlagwortsuche wurde im Entwurf der Praktikumsbörse auch die Möglichkeit, nach anderen Kriterien zu suchen, gelegt. In der Implementierung des Prototypen der Praktikumsbörse wurde aus Zeitgründen zunächst nur die Suche nach den Orten und die Schlagwortsuche umgesetzt (Abb.14).



The image shows a search interface on a light gray background. It consists of two rows of labels and dropdown menus. The first row has the label 'Ort:' followed by a dropdown menu with the text 'Bitte wählen' and a downward arrow. The second row has the label 'Schlagwort:' followed by a similar dropdown menu. Below these two rows is a rectangular button with the text 'Los geht's'.

Abbildung 14: Suche

Um dem Nutzer die Suche etwas komfortabler zu gestalten und sicher zu gehen, dass in der Datenbank vorhandene Begriffe in die Suche einbezogen werden, werden die Orte und Schlagworte in einer Dropdown-Liste vorgegeben. Dabei sind in dieser Liste die Orte und Schlagworte aufgeführt, die sich aktuell in der Datenbank befinden.

Es ist möglich keines der beiden Suchkriterien, beide oder nur eines der beiden auszuwählen. Wird nach keinem bestimmten Ort oder anhand eines Schlagwortes gesucht, so werden alle in der Datenbank vorhandenen Praktika als Ergebnis ausgegeben. Wenn ein Ort ausgewählt wird, dann werden alle Praktikumsangebote geliefert, die in der Tabelle Standort für den Ort diesen Wert besitzen. Im Falle des Schlagwortes verhält es sich genauso.

Die Ergebnisse werden zum einen als Marker in einer Google Map dargestellt und zum anderen in Tabellenform unmittelbar darunter. Dem Nutzer wird zudem beschrieben, wie viele Ergebnisse gefunden wurden. Dabei ist die Zahl, der auf einer Seite aufgelisteten Ergebnisse, auf 10 reduziert. Sobald die Zahl der Ergebnisse 10 oder ein Vielfaches von 10 übersteigt, werden Buttons wie zum Beispiel „Seite2“ generiert. Nachdem der Nutzer einen dieser Buttons aktiviert, werden die nächsten 1 bis 10 Ergebnisse angezeigt. Dabei bleibt die Karte im oberen Teil der Seite bestehen. Werden keine passenden Ergebnisse

gefunden, so wird das dem Nutzer mitgeteilt.

Außer der Suche nach Praktikumsangeboten über die Kriterien Ort und Schlagwort, wurde auch eine Suche für das Prüfungsamt bzw. den Praktikumsbeauftragten nach Praktikanten implementiert. Diese haben die Möglichkeit einen Studiengang auszuwählen und bzw. oder einen Nachnamen eines Praktikanten einzugeben.

#### **5.2.4. Visualisierung der Praktikumsorte**

In Folge einer Suche nach Praktikumsangeboten sollen die Ergebnisse nicht nur in Textform sondern auch in einer Karte dargestellt werden. Die Karte wird unter Verwendung der Google Maps API erstellt. In diesem Abschnitt wird beschrieben, was zur Erstellung einer eigenen Karte nötig ist und welche Funktionen die Karte der Praktikumsbörse besitzt.

Jeder, der eine erste eigene Karte mit der von Google dafür bereitgestellten Programmierschnittstelle erzeugen möchte, führt zunächst die selben Schritte aus. Denn bevor mit einer eigenen Google Maps Anwendung begonnen werden kann, wird der API-Key benötigt. Dieser Schlüssel ist kostenlos auf der Google Maps API Homepage erhältlich (Abb.15). Um den API-Schlüssel zu bekommen, müssen zunächst die Nutzungsbedingungen gelesen und akzeptiert werden. Darin steht unter anderem, dass keine Google Bildmaterialien entwendet werden dürfen. Außerdem muss das Google-Logo immer sichtbar sein und darf nicht verdeckt oder entstellt werden. [8, S. 27] Eine weitere Nutzungsbedingung besagt zudem, dass der angebotene Dienst für Nutzer frei zugänglich sein muss. Des Weiteren äußert sich der Google-Konzern in dieser Form:

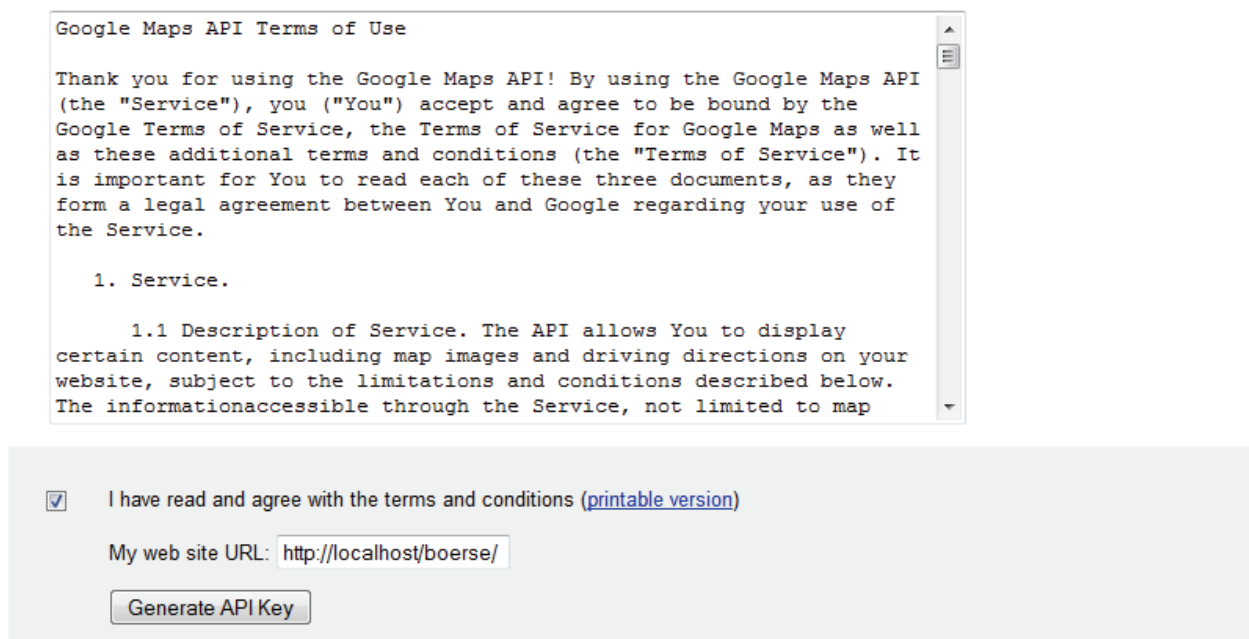
„Bestimmte Arten der API-Nutzung wünschen wir jedoch auf keinen Fall. Dazu gehört beispielsweise, dass die Karten zum Kennzeichnen von Plätzen des städtischen Drogenhandels oder sonstigen illegalen Tätigkeiten verwendet werden. Außerdem halten wir uns an die Datenschutzbestimmungen und möchten deshalb nicht, dass unsere API zum Ausforschen und Verbreiten von vertraulichen Informationen missbraucht wird.“[11]

Der gleiche Absatz ist in englischer Sprache im oberen Teil der Abbildung 15 zu sehen.

Um einen API-Schlüssel zu beantragen muss die Internetadresse, für die der Google Maps Dienst verwendet werden soll, angegeben werden. Dabei ist darauf zu achten, dass der Schlüssel nur für eine Domäne und ein bestimmtes Unterverzeichnis gilt. Aus diesem Grund muss für jede andere Domäne ein neuer Schlüssel generiert werden. Das stellt

weiter kein Problem dar, denn Google weist so viele Schlüssel zu, wie benötigt werden. [8, S. 27] Es kann des Weiteren ein API-Key für Adressen generiert werden, auf die nicht vom Internet aus zugegriffen werden soll, wie zum Beispiel `http://localhost/boerse/` (Abb.16). Der erzeugte Schlüssel kann nun für dieses Verzeichnis und alle seine Unterverzeichnisse genutzt werden. Das ist besonders praktisch für Seiten, die noch in der Entstehung oder Entwicklung sind.

- There are some uses of the API that we just don't want to see. For instance, we do not want to see maps that identify the places to buy illegal drugs in a city, or any similar illegal activity. We also want to respect people's privacy, so the API should not be used to identify private information about private individuals. Remember that we reserve the right to suspend or terminate your use of the service at any time, so please read through the [FAQ](#) and [forum posts](#) to decide whether your site meets the Terms of Use before you begin API integration.



Google Maps API Terms of Use

Thank you for using the Google Maps API! By using the Google Maps API (the "Service"), you ("You") accept and agree to be bound by the Google Terms of Service, the Terms of Service for Google Maps as well as these additional terms and conditions (the "Terms of Service"). It is important for You to read each of these three documents, as they form a legal agreement between You and Google regarding your use of the Service.

1. Service.

1.1 Description of Service. The API allows You to display certain content, including map images and driving directions on your website, subject to the limitations and conditions described below. The information accessible through the Service, not limited to map

☒ I have read and agree with the terms and conditions ([printable version](#))

My web site URL:

Abbildung 15: Anmeldung für einen API-Schlüssel  
(<http://code.google.com/apis/maps/signup.html>)

Nachdem ein API Key generiert wurde, stellt Google dem Nutzer außerdem ein einfaches Startbeispiel zur Verfügung. Anhand dieses Beispiels lassen sich die für eine Karte grundlegenden Elemente analysieren. Es lässt sich einfach in die eigene Webseite einbinden und gut damit experimentieren. Wie viele andere Webapplikationen baut die Google Maps API auf Java Script und AJAX auf. Der eigene API Schlüssel ist bereits in das Java Script des Beispielcodes eingebettet (Anhang A).

Bei dem Codebeispiel von Google handelt es sich um eine Webseite im XHTML-Standard. Im head-Teil des Dokumentes befinden sich zwei script-Elemente. Zum einen ein script-

Element, dass über sein src-Attribut auf die Position der API auf dem Google-Server verweist, wobei der API-Schlüssel als Parameter übergeben wird. Zum anderen ein script-Element, das JavaScript Code enthält. Die Aufgabe des Codes ist die Kommunikation mit der Google-API und schließlich das Erstellen der Karte. Gestartet wird das JavaScript vom onload-Ereignis im body-Teil des Dokumentes sobald der Browser (neu-)geladen wird.

```
function load() {  
    if (GBrowsersCompatible()) {  
        var map = new GMap2(document.getElementById("map"));  
        map.setCenter(new GLatLng(37.4419,-122.1419), 13);  
    }  
}
```

Zunächst wird die Funktion load() aufgerufen und in einer if-Anweisung die Kompatibilität des Browsers getestet. Im nächsten Schritt wird durch eine Funktion ein GMap-Objekt erzeugt, dass sich auf den div-Abschnitt map im body-Teil beziehen soll und es wird der Variablen „map“ zugewiesen. Da sich das GMap-Objekt nun in einer map-Variablen befindet, kann es zum Aufrufen beliebiger Methoden von GMap genutzt werden. In der folgenden Zeile wird die Methode setCenter aufgerufen, in der die Koordinaten von Palo Alto (Kalifornien) als Mittelpunkt der Karte und die Karte auf den Zoomfaktor 13 gesetzt wird.[8, S. 29 f.] In Abbildung 16 ist das Ergebnis des Beispielcodes zu sehen.

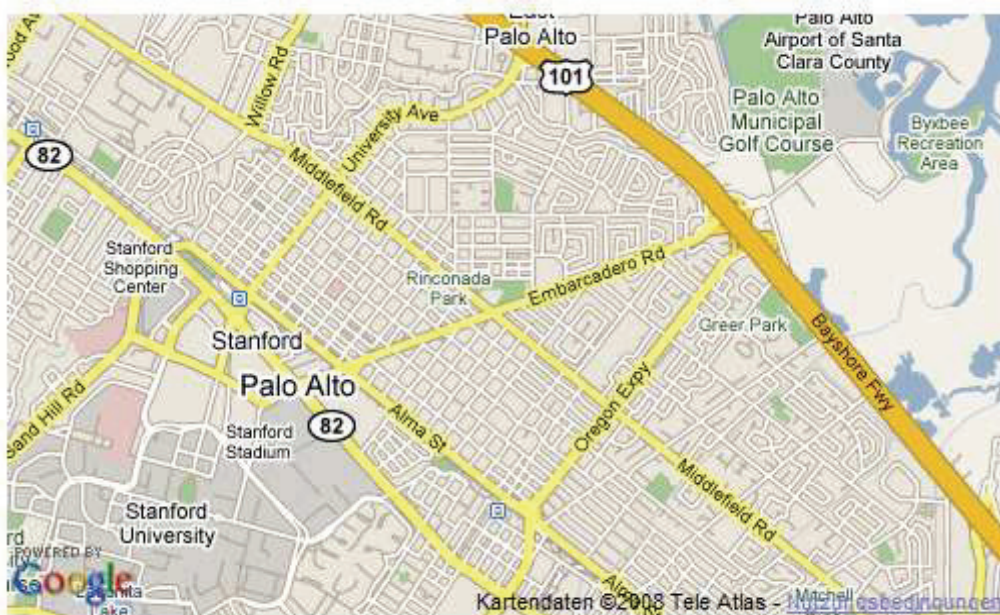


Abbildung 16: Google Startbeispiel

Alles was die Karte jetzt noch benötigt, muss selbstständig durch Erweiterungen des Codes hinzugefügt werden. Dazu gehört eine Navigationsleiste, Marker, die Leiste um den Kartentypen zu wechseln und der Mittelpunkt der Karte muss verändert werden. Dafür muss zunächst etwas recherchiert werden. Dabei kann auf unterschiedliche Weise vorgegangen werden. Zum einen bietet Google eine ausführliche englische Dokumentation zu der Google Maps API an, die online auf der Seite <http://www.google.com/apis/maps/documentation> einzusehen ist. Zum anderen gibt es Literatur, die sich bereits mit dem Thema befasst hat, zum Beispiel „Beginning Google Maps Applications with PHP and AJAX“ oder die deutsche Übersetzung „Google Maps Anwendungen mit PHP und AJAX“. In beiden Fällen werden viele Beispiele gezeigt und beschrieben. Dennoch ist eine Anpassung des Codes oftmals nicht sehr einfach und nimmt viel Zeit in Anspruch. Aus diesem Grund wurde eine weitere Möglichkeit genutzt. Diese Alternative beinhaltet die Nutzung von bestehenden PHP-Bibliotheken für die Google Maps API. Zwei von diesen Bibliotheken sind der Gmapper in der Version 1.2, zu finden unter <http://gmapper.ajax-info.de>, und das Projekt GoogleMapAPI in der Version 2.5 zu finden unter der Seite <http://www.phpinsider.com/php/code/GoogleMapAPI/>. Bevor eine Bibliothek für ein eigenes Projekt genutzt werden kann, muss auf die Lizenz geachtet werden. Die Gmapper-Bibliothek unterliegt der GPL und die GoogleMapAPI-Bibliothek unterliegt der LGPL. Beide Projekte gehören zu freier Software: „Freie Software bedeutet die Freiheit des Benutzers, die Software zu benutzen, zu kopieren, sie zu vertreiben, zu studieren, zu verändern und zu verbessern.“ [12]

### **GoogleMapAPI-Bibliothek**

Zur Umsetzung der Praktikumsbörse wurde schließlich die GoogleMapAPI-Bibliothek genutzt. Diese ist einfach in die eigenen Seiten zu integrieren und bietet eine Vielzahl an nützlichen Funktionen. Die wichtigsten, die für die Google Map in der Praktikumsbörse benötigt wurden (Anhang B), werden im Folgenden kurz beschrieben.

```
GoogleMapAPI($map_id='map', $app_id='MyMapApp')
```

Dies ist eine Klasse, mit der eine neue Karte erzeugt wird. Darin werden alle für eine Karte benötigten Variablen und Funktionen gesetzt. Somit lassen sich mehrere Google Maps auf einer Seite erstellen. Die „map\_id“ muss gesetzt werden, wenn mehrere Karten auf einer Seite vorhanden sind. Die „app\_id“ hingegen wird benötigt, falls die Yahoo GeoCode API für Geocodierungsdienste genutzt werden soll. Es ist dabei ratsam, eine einheitliche „app\_id“ pro Anwendung zu gebrauchen.



`setAPIKey($key)`

An dieser Stelle wird der API-Schlüssel gesetzt. Entweder er wird direkt in die Funktion eingetragen oder steht in einer Variablen, wie zum Beispiel `$key`.

Mit `printHeaderJS()` wird das script-Element ausgegeben, dass den Verweis auf den Google-Server enthält und den Schlüssel-Parameter übergibt. Genau wie auch bei dem Google Startbeispiel muss dies im head-Teil stehen.

Die Funktion `printMapJS()` gibt den JavaScript-Code zurück, der für die Erstellung der Karte zuständig ist. Das beinhaltet nicht nur, wie in dem Codebeispiel von Google, die Überprüfung des Browsers, das Erzeugen eines GMap-Objekts und das Setzen des Kartenmittelpunkts. In der Funktion `printMapJS()` wird einiges mehr festgelegt. So wird z.B. die Navigationsleiste, die Leiste für den Kartentypen sowie ein Maßstab in Meilen und Kilometer mit Defaultwerten gesetzt. Des Weiteren werden der Mittelpunkt der Karte sowie der Zoomfaktor so festgelegt, dass alle Marker im Darstellungsfeld zu sehen sind. Ein weiteres Extra, das bereits voreingestellt ist, sind die Schaltflächen im Info-Fenster, mit denen ein Ort als Ziel- bzw. Startort ausgewählt wird (Abb.18) und dann die Adresse des zweiten Ortes eingegeben werden kann, um die Route berechnen zu lassen (Abb.17).

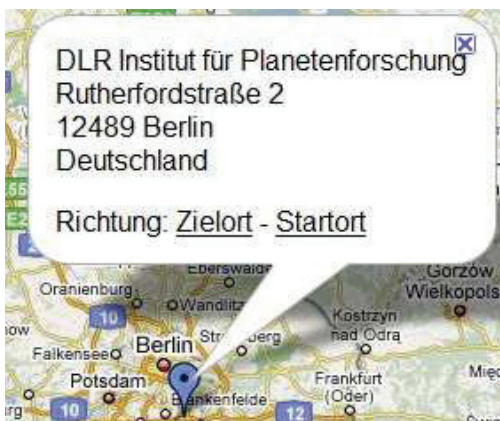


Abbildung 18: Info Window Richtung

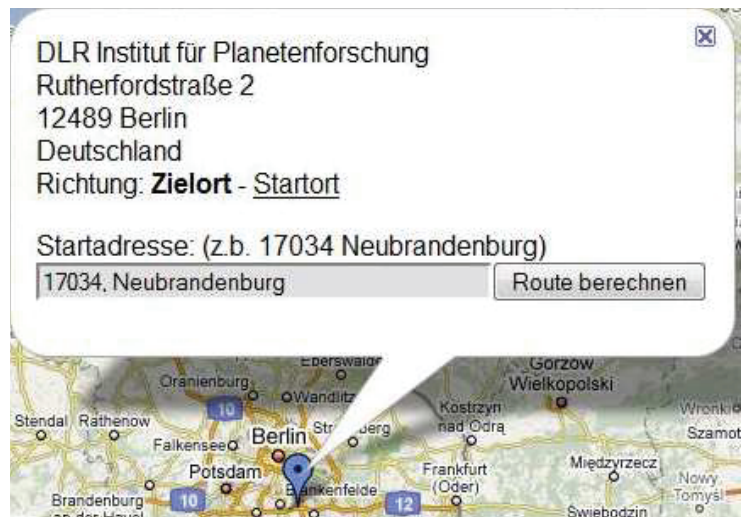


Abbildung 17: Info Window Route berechnen

Beim Einsatz dieses Extras in der Karte der Praktikumsbörse wurde jedoch die Bezeichnungen von Englisch auf Deutsch verändert. Im Original sähe das so aus, wie in Abbildung 19 gezeigt. Es kann aber auch alles, was standardmäßig als Funktionen in die Karte eingefügt wurde, deaktiviert werden.



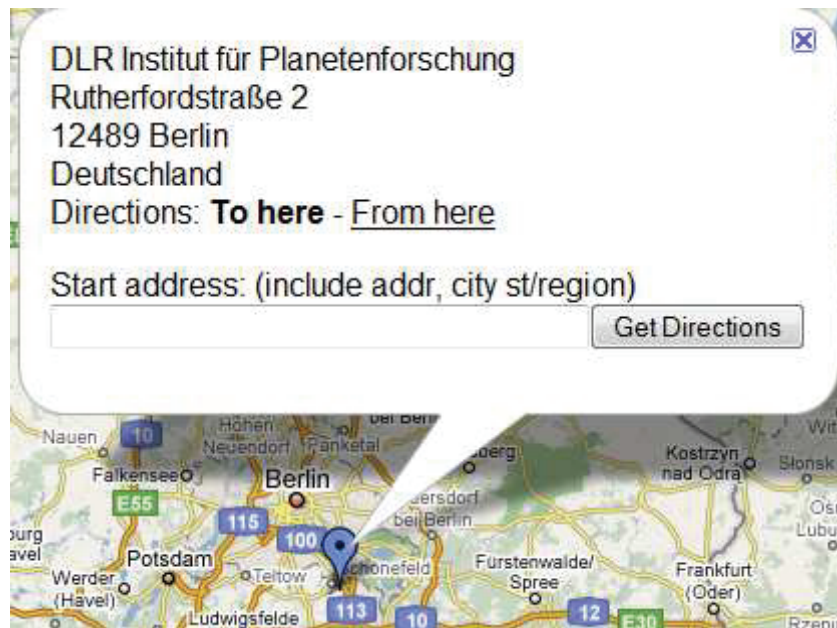


Abbildung 19: Info Window Get Directions

Durch *printMap()* wird der Html-Code für die Karte und mit *printSidebar()* der Html-Code für eine seitliche Leiste mit Links zu den Markierungen ausgegeben.

```
addMarkerByCoords($lon,$lat,$title=",$html=")
```

Dies ist eine weitere wichtige Funktion, da sie einen Marker über die Angabe von Longitude und Latitude an genau diesem Ort auf der Karte erstellt. Der Titel gibt an, was in der seitlichen Leiste als Linkname steht und die Variable *\$html* wird genutzt um das Info Fenster des Markers zu füllen, das sich öffnet, wenn der Marker geklickt wird.

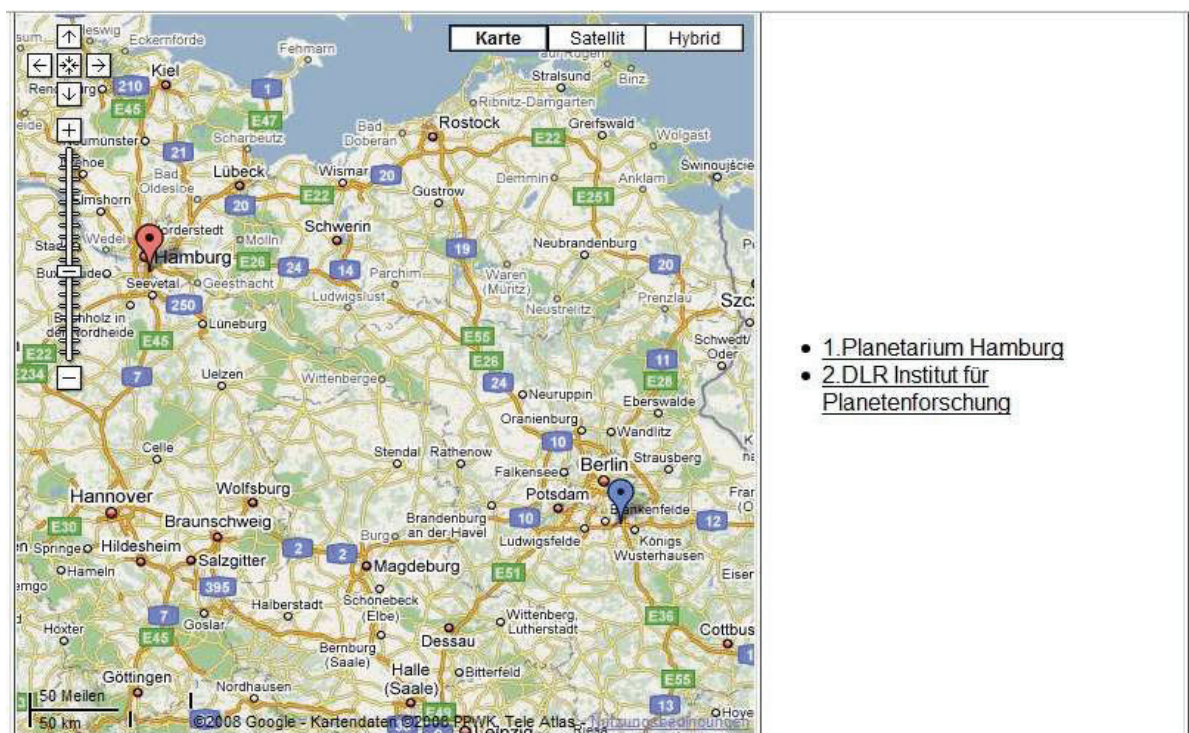
```
addMarkerIcon($iconImage,$iconShadowImage,$iconAnchorX = 'x',$iconAnchorY = 'x',  
$infoWindowAnchorX = 'x',$infoWindowAnchorY = 'x')
```

Mit dieser Funktion lässt sich das Symbol des aktuellen Markers setzen. Für *\$iconImage* und *\$iconShadowImage* wird der Pfad zu einem Bild benötigt, dieser kann auch eine URL sein. Mit *\$iconAnchorX* bzw. *Y* wird die Position des Symbols zum Kartenpunkt festgelegt. Die Parameter *\$infoWindowAnchorX* bzw. *Y* gibt die Position an, an der das Info Fenster mit dem Symbol verbunden ist. Statt *addMarkerIcon* kann auch die Funktion *setMarkerIcon* genutzt werden, diese legt die Symbole von allen Markern in der Karte fest. Im Gegensatz zu *setMarkerIcon* muss *addMarkerIcon* für jeden Marker aufgerufen werden.

Für die Praktikumsbörse wurde *addMarkerIcon()* verwendet, da im Kapitel Funktionalitäten

festgelegt wurde, dass der Nutzer anhand unterschiedlicher Färbungen der Marker erkennen soll, ob ein Angebot aktuell oder schon länger als ein Jahr in der Praktikumsbörse vorliegt. Deshalb muss es zwei unterschiedliche Symbole für die Marker geben. Das Problem was es zu lösen galt, ist zu ermitteln, wie lange ein Praktikumsangebot bereits in der Börse vorhanden ist. Dazu wurde die PHP-Funktion `time()` genutzt, um den aktuellen Unix-Zeitstempel zu erfahren. Weiterhin wurde mit der Funktion `strtotime()` das Eintragsdatum des Angebotes, das als Datum in englischer Textform angegeben ist, ebenfalls in einen Unix-Zeitstempel umgewandelt. Mit Hilfe dieser beiden Zeitstempel lässt sich nun die Differenz zwischen dem aktuellen Datum und dem Eintragsdatum ermitteln. Beträgt die Differenz mehr als ein Jahr, so wird statt einem roten Standardmarker von Google ein blauer eingesetzt.

Nachdem durch `addMarkerIcon()` das Symbol für den Marker feststeht, wird mit `addMarkerByCoords()` der Marker, auf Grund der Koordinaten des Ortes in der Praktikumsbörse-Datenbank, gesetzt. Alle vier Parameter von `addMarkerByCoords()` werden mit Daten aus der Datenbank gefüllt. Somit werden nur die Marker erstellt, die über die Suche abgefragt werden.



Diese  stehen für aktuelle Praktikumsangebote.


Diese  stehen für Praktikumsangebote, die schon 1 Jahr oder länger in der Praktikumsbörse stehen.

Abbildung 20: Ergebnis

In Abbildung 20 ist das Ergebnis einer Suchabfrage zu sehen. Dabei wurden zwei Marker erzeugt, ein blauer und ein roter. In der Seitenleiste sind die beiden dazugehörigen Links zu sehen, mit denen die Info Fenster der Marker geöffnet werden können. Die Links sind nummeriert und zwar in der Reihenfolge der darunter in Tabellenform aufgeführten Praktikumsangebote. Dies soll dem Nutzer die Zuordnung erleichtern. Außerdem wurde unter die Karte eine kleine Legende der Symbole gesetzt, damit der Nutzer die unterschiedlichen Symbole zuordnen kann.

### **5.2.5. Dateneingabe mit Geocodierung**

Bis zu dieser Phase der Implementierung kann der Nutzer sich anmelden, nach Praktikumsangeboten suchen und diese werden ihm in einer Google Maps Karte angezeigt. Als nächster und letzter Teil der Implementierung wird das Eintragen neuer Praktikumsangebote und Praktikanten in die Datenbank umgesetzt.

Bei der Dateneingabe ist zwischen dem Praktikumsangebot und dem Praktikanten, der das Praktikum belegt, zu differenzieren. Aus diesem Grund muss die Eingabe getrennt möglich sein. Zunächst wird also ein Formular erstellt, indem lediglich Daten eingegeben werden, die für ein Angebot notwendig sind. Dabei muss sichergestellt werden, dass bestimmte Felder ausgefüllt werden müssen, da sonst die Beziehungen zwischen den Tabellen nicht richtig zugeordnet werden. Des Weiteren muss mindestens der Ort des Praktikumsangebotes in das Formular eingetragen werden, da andernfalls die Geocodierung nicht funktionieren kann oder viel zu ungenau wird, wenn nur das Land angegeben wird.

Die Praktikanten werden in einem zweiten Formular hinzugefügt, dabei muss die ID für das Praktikum angegeben werden, das durch den Praktikanten belegt wird. Dazu wird vor dem Eingabeformular des Praktikanten eine Suche nach den Praktika gestellt, über die unter anderem die ID des Praktikums ermittelt werden kann. Somit sollte die Zuordnung einfach gelingen. Außerdem ist die Möglichkeit umgesetzt worden, den Praktikumsbericht später hinzu zufügen, als zum Beispiel den Praktikanten und den Praktikumsvertrag, da der Praktikumsbericht erst nach dem Praktikum angefertigt wird.

#### **Geocodierung der Standorte**

Wie bereits in den Grundlagen beschrieben, kann dieser Dienst direkt über eine HTTP-Anfrage oder die Verwendung eines GClientGeocoder Objekts erreicht werden.

Der Java Script-Geocoder richtet Ajax-Aufrufe direkt vom Rechner des Benutzers der Webseite an die Google-Server. Diese Möglichkeit ist schnell und bequem, doch eher für Anwendungen gedacht, bei denen die Adresse über die codiert wird nicht für weitere Verwendungen gespeichert werden soll. [8, S. 87 f.]

Aus diesem Grund wird zur Geocodierung der Standorte der Praktikumsangebote die HTTP-Anfrage genutzt, da die Adresse und die Längen- und Breitenangaben in der Datenbank abgespeichert werden sollen.

Der Zugriff zum Google-Dienst stützt sich auf REST-Anforderungen (Representational State Transfer). Dabei handelt es sich um eine einfache HTTP-Anforderung (<http://maps.google.com/maps/geo?>), die folgende GET Parameter übergeben kann:

- q (*obligatorisch*) – Die Adresse, die geocodiert werden soll.
- key (*obligatorisch*) – Der API-Schlüssel.
- output (*obligatorisch*) – Das Format, in dem die Antwort erzeugt werden soll. Zur Auswahl steht: xml, kml, csv, oder (default) json.
- ll (*optional*)-- Die Breite und Länge der Darstellungsfeldmitte durch ein mit Komma getrennte Zeichenkette (z.B. "ll=40.479581,-117.773438" ). Dieser Parameter hat nur Bedeutung, wenn der spn Parameter auch gesetzt wird.
- spn (*optional*)-- Die Abgrenzung des Darstellungsfeldes durch eine mit Komma getrennte Zeichenkette aus Breite und Länge ausgedrückt ("spn=11.1873,22.5"). Dieser Parameter hat nur Bedeutung, wenn der ll Parameter auch gesetzt wird.
- gl (*optional*) -- Die Landeskennzahl, spezifiziert als ccTLD (" Spitzendomain") two-character Wert. [7]

Im Fall der Praktikumsbörse werden für q die in dem Eingabeformular angegebenen Adressdaten in die URL geschrieben. Dabei werden die Adressteile mit einem + verbunden. Dem Parameter key wird der API-Schlüssel übergeben. Als output wird csv gewählt.

Die Antwort des Google Servers als csv-Datei hat den Vorteil, dass diese kurz und damit einfach zu analysieren ist. Diese Antwort besteht aus vier Zahlen, die durch ein Komma getrennt sind, wie in folgendem Beispiel:

200, 7 ,42.730210,-73.691800  
1. 2. 3. 4.

1. der HTTP Statuscode
2. die Genauigkeit (Genauigkeitskonstante von 0 bis 9)
3. Breite (Latitude)
4. Länge (Longitude)

Die folgenden beiden Tabellen zeigen zum einen die verschiedenen Bedeutungen des HTTP Statuscodes (Tab. 5) und zum anderen die Bedeutung der Genauigkeitsangaben (Tab. 6).

Code	Konstantenname	Beschreibung
200	G_GEO_SUCCESS	Keine Fehler aufgetreten. Der Parser konnte die Adresse erfolgreich verarbeiten und deren Geocode zurückgeben.
500	G_GEO_SERVER_ERROR	Eine Geocoding-Anforderung konnte nicht erfolgreich verarbeitet werden, der genaue Grund für den Fehler ist aber unbekannt.
601	G_GEO_MISSING_ADDRESS	Der HTTP-Parameter q fehlte entweder oder besaß keinen Wert.
602	G_GEO_UNKNOWN_ADDRESS	Für die angegebene Adresse konnte kein entsprechender geographischer Ort gefunden werden. Das kann daran liegen, dass die Adresse noch relativ neu oder falsch ist.
603	G_UNAVAILABLE_ADDRESS	Der Geocode für die angegebene Adresse konnte aus rechtlichen oder vertraglichen Gründen nicht zurückgegeben werden.
610	G_GEO_BAD_KEY	Der angegebene Schlüssel ist entweder ungültig oder entspricht nicht der Domäne, für die er vergeben wurde.
620	G_TOO_MANY_QUERIES	Sie haben zu häufig auf den Dienst zugegriffen und wurden entweder vorübergehend oder dauerhaft vom weiteren Einsatz ausgeschlossen.

Tabelle 5: HTTP-Statuscode [8, S. 84]

Konstante	Beschreibung
0	Unbekannte Position.
1	Landniveau-Genauigkeit.
2	Gebiet (Staat, Provinz, Präfektur, usw.) Niveau-Genauigkeit.
3	Subgebiet (Grafschaft, Stadtbezirk, usw.) Niveau-Genauigkeit.
4	Stadt (Stadt, Dorf) Niveau-Genauigkeit.
5	Postcode (Postleitzahl) Niveau-Genauigkeit.
6	Straßenniveau-Genauigkeit.
7	Kreuzungsniveau-Genauigkeit.
8	Adressniveau-Genauigkeit
9	Proposition (Namen, Eigennamen, Einkaufszentrum usw.), Niveau-Genauigkeit.

Tabelle 6: Genauigkeitskonstanten [7]

Auf Grund des einfachen Aufbaus der csv-Datei ist es leicht diese zu parsen und die vier Elemente in entsprechende Variablen zu speichern um damit weiterarbeiten zu können. So werden die Längen- und Breitenangaben in \$lon und \$lat übergeben und die Variablen in der SQL-Anweisung zum Speichern der Daten in der Datenbank genutzt.

Im Folgenden ist der Vorgang des Geocodierens als Sequenzdiagramm dargestellt (Abb.21).

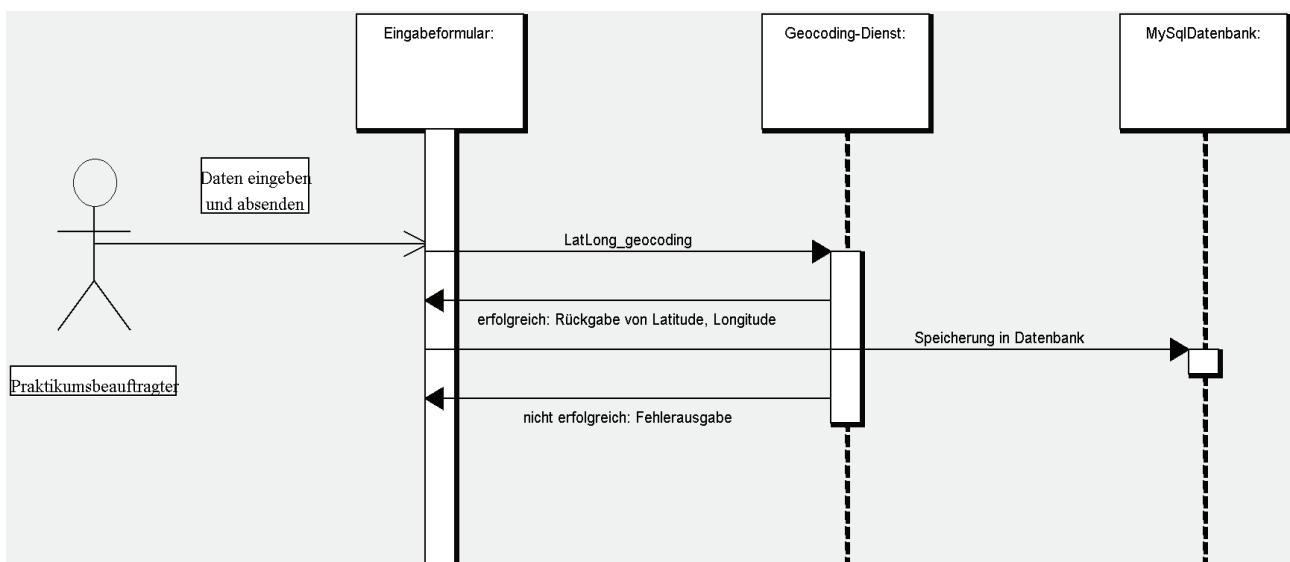


Abbildung 21: Sequenzdiagramm Geocoding

Eine HTTP-Anfrage wird an den Google-Server abgeschickt, daraufhin gibt dieser die

Antwort in Form einer csv-Datei zurück. Die eingelesenen Zeilen werden auf kommaseparate Felder hin überprüft und der Inhalt der Felder wird in einem Array gespeichert. Die Länge und Breite wird in Variablen umgespeichert und diese werden zur Speicherung in der MySQL-Datenbank genutzt.



## 6. Testphase

Zum Ende jeder Implementierung ist es wichtig einige Tests mit dem erstellten Prototypen durchzuführen. Dabei ist darauf zu achten, dass alle Seiten das gewünschte Verhalten zeigen. Die Testphase soll mögliche Fehler und Probleme der Anwendung aufdecken.

Für die Tests am System wurden einige Beispieldaten in die MySQL-Datenbank eingefügt. Um die Anwendung offline testen zu können, wurde mit Hilfe von XAMPP eine lokale Serverumgebung geschaffen und mit dem Mozilla Firefox Browser in der Version 3.0.3 getestet.

### Test Anmeldung

Um zu testen, ob die Anmeldung funktionsfähig ist, wurden in der Tabelle Login die Benutzer dozent, pruefamt und praktbetreuer mit den Passwörtern 123, 456, 789 eingefügt. Als Zugriff wurde „man“ festgelegt, da die Tests für die Anmeldung über die MySQL-Datenbank ohne LDAP-Freischaltung möglich sind.

Im folgenden Beispiel wurde der Dozent getestet. Als erstes wurde der richtige Benutzername und das richtige Passwort in die Anmeldung eingetragen. Daraufhin folgt die Webseite in Abbildung 22. Die Anmeldung war also erfolgreich und richtig.

Praktikumsbörse-Dozent

Ort:

Schlagwort:

Version 1.1 - letzte Aktualisierung: September 2008

Abbildung 22: erfolgreiche Anmeldung

Als nächstes wurde das Verhalten des Systems getestet, wenn eine falsche Eingabe in das Anmeldeformular getätigt wird. Als Ergebnis erscheint die Anmeldemaske des Dozenten mit der Fehlermeldung in Abbildung 23.



Ihre Eingaben waren nicht korrekt!

Anmeldung Dozent

Abbildung 23: Falsche Eingabe

Der dritte Test am Anmeldeformular des Dozenten simuliert die Eingabe eines Benutzers aus der Datenbank mit richtigem Passwort, dieser gehört jedoch zu einer anderen Nutzergruppe. Daraufhin wird die Index-Seite angezeigt, mit der Fehlermeldung: „Falsche Anmeldemaske!“ (Abb.24). Der Nutzer ist jedoch eingeloggt und kann ohne Anmeldung auf seine Startseite.

Falsche Anmeldemaske!

Praktikumsbörse

**Anmeldung für die Praktikumsbörse**

[Student](#)

[Dozent](#)

[Prüfungsamt](#)

[Praktikumsbeauftragter](#)

[ohne Anmeldung](#)

[Zurück](#)

[logout](#)

Version 1.1 - letzte Aktualisierung: September 2008

Abbildung 24: In der falschen Anmeldemaske angemeldet

Außerdem wurde getestet, ob die Möglichkeit besteht, dass jemand durch Eingeben des Pfades in der Adresszeile des Browsers ohne Anmeldung auf eine Seite zugreifen kann, für die er nicht das Recht besitzt. In dem Fall wird der Nutzer, bei richtigem Verhalten des Systems, auf die Index-Seite gebracht und darauf hingewiesen, dass er nicht als Student, Dozent, Prüfungsamtsmitglied oder Praktikumsbeauftragter angemeldet ist. Die Tests zur Anmeldung für die oben genannten drei Nutzer waren positiv. Der Student konnte nicht getestet werden, da dieser auf den LDAP-Server zugreifen muss.

Des Weiteren wurden abschließende Tests zur Suche durchgeführt. Im Großen und Ganzen sind dort keine Probleme aufgetaucht. Doch bei der Darstellung der Ergebnisse in den Tabellen könnte noch etwas verändert werden. Im Moment werden insbesondere dem Dozenten alle Daten zum Praktikum in einer Tabelle dargestellt. Es erscheinen die Zeilen mit den Angaben zum Praktikanten, auch wenn noch kein Praktikant dem Praktikum zugeteilt ist. Dies führt zu leeren Feldern und sieht nicht gut aus.

Während den Tests zur Eingabe neuer Praktikumsangebote ist besonders eines aufgefallen, und zwar dass für die Adresse zur Geocodierung zum Beispiel für die Straße nur ein Wort eingefügt werden darf, ansonsten schlägt das Geocodieren fehl.

Des Weiteren wurde das Verhalten des Systems insbesondere bei der Darstellung mit einem weiteren Browser getestet. Das System der Praktikumsbörse zeigt bei der Ausführung mit dem Internet Explorer 7 ein anderes Verhalten (Abb.25).

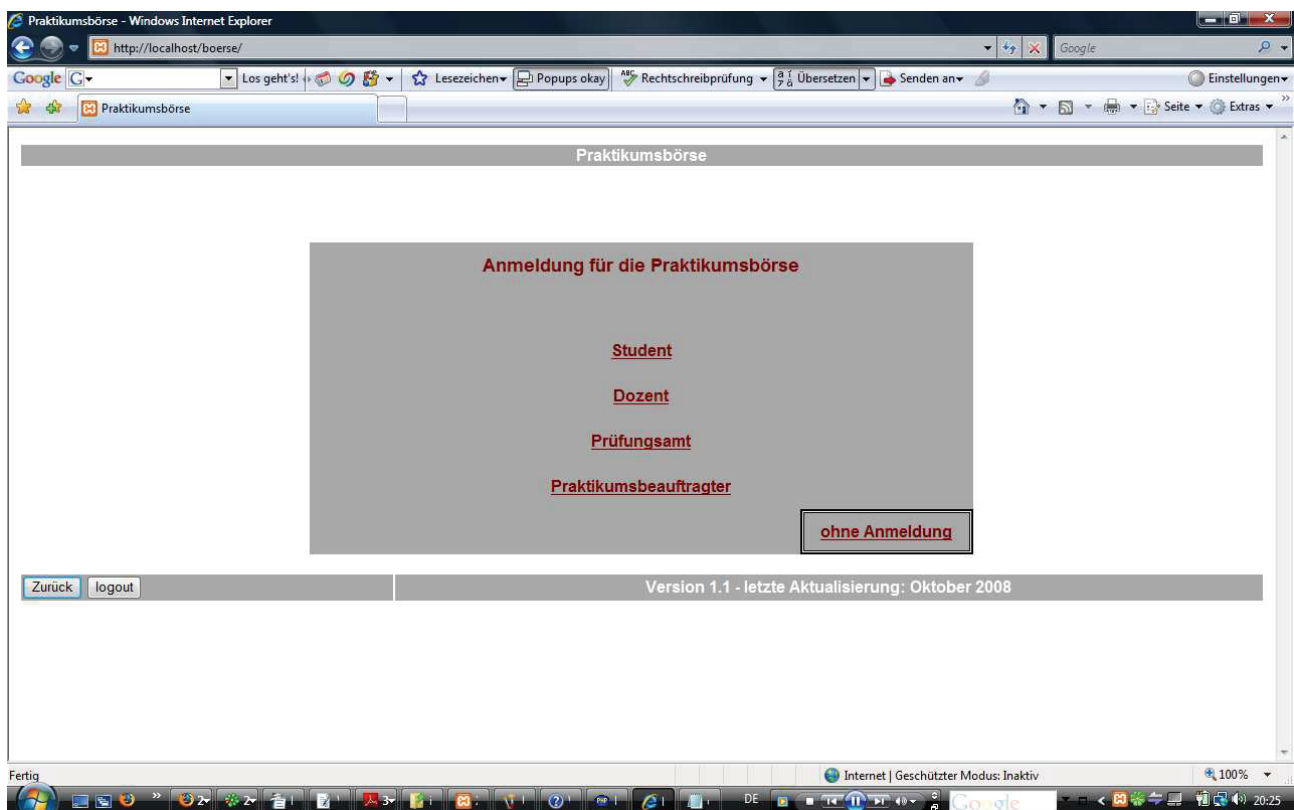


Abbildung 25: Praktikumsbörse mit Internet Explorer 7

In Abbildung 25 ist zu sehen, dass die Seite über die gesamte Breite des Browserfensters dargestellt wird und nicht wie es sein sollte eine Breite von 800 Pixeln aufweist. Alle weiteren Darstellungen werden richtig ausgeführt. Die Anmeldung der Nutzer weist keine Probleme auf. Die Suche und Darstellung der Ergebnisse funktioniert ebenfalls. Auch die

Eingabe neuer Praktikumsangebote wird ohne Fehler durchgeführt. Dennoch reagieren die Buttons in der Fußzeile auf keiner Seite, die mit dem Internet Explorer 7 ausgeführt wird. Somit kann sich der Nutzer nicht ausloggen oder auf die vorherige Seite wechseln.

Aufgrund der knapp bemessenen Zeit für die Bachelorarbeit, konnte die Testphase nicht gründlicher durchgeführt werden. Wenn das Projekt fortgeführt wird, müsste dieses nachgeholt werden und nochmals auf die Probleme mit dem Internet Explorer 7 eingegangen werden.

## **7. Schlussbetrachtung und weiterführende Arbeiten**

Ausgangspunkt dieser Arbeit war die Aufgabe eine Praktikumsbörse mit Visualisierung der Praktikumsorte unter Nutzung der Google Maps Schnittstelle zu entwerfen und zu implementieren. In den vorangegangenen Kapiteln wurde zunächst grundlegendes Wissen geklärt, der Entwurf des Systems beschrieben und die Implementierung dokumentiert.

### **Zusammenfassung der Ergebnisse**

Das Ergebnis der Implementierung ergab einen Prototypen einer Praktikumsbörse. Mit diesem System ist es möglich, dass verschiedene Nutzergruppen eine Suche nach Praktikumsangeboten bzw. Praktikanten durchführen können. Dabei sind die Ergebnisse auf die jeweiligen Bedürfnisse und Berechtigungen abgestimmt. Die Praktikumsorte werden in einer Google Maps Karte angezeigt und, abhängig von dem Datum des Eintrags in der Praktikumsbörse, durch ein anderes Symbol dargestellt. Über eine Leiste mit Links zu den einzelnen Markierungen ist es möglich direkt auf einen Praktikumsort zu „springen“, auch wenn dieser außerhalb des Darstellungsfeldes liegt. In der Leiste sind die Praktikumsbetriebe aufgelistet und genauso nummeriert, wie in den Tabellen unterhalb der Karte, dies soll die Zuordnung und das Finden von Praktikumsorten auf der Karte erleichtern. Mit Hilfe von Info Fenstern zu den Markierungen lassen sich die Adressen der Praktikumsstellen erfahren. Alle weiteren Informationen stehen in Tabellenform unterhalb der Karte. Des Weiteren ist es möglich eine Route von oder zu einem Praktikumsort berechnen zu lassen, wenn die Adresse des zweiten Ortes angegeben wird. Um die Praktikumsbörse aktuell zu halten, ist es notwendig und möglich neue Praktikumsangebote über ein Formular einzugeben. Dabei werden die Praktikumsorte auf Grund ihrer Adressen mit Hilfe eines Google-Dienstes geocodiert. Insgesamt wurden fünf Nutzergruppen berücksichtigt, vier von ihnen gelangen nur über eine Anmeldung in die Praktikumsbörse. Dadurch sollen bestimmte Daten vor Unbefugten gesichert werden.

Abschließend ist zu sagen, dass ein recht umfangreicher Prototyp entstanden ist, der dennoch viel Platz für Erweiterungen und Optimierungen lässt.

### **Weiterführende Arbeiten**

Aufgrund der knapp bemessenen Zeit von zwei Monaten konnte einiges in der Praktikumsbörse zunächst nur exemplarisch umgesetzt werden.

So wurde zum Beispiel die Anmeldung des Administrators noch nicht umgesetzt. Diese ist

jedoch in dem Entwurf enthalten und müsste als nächstes realisiert werden.

Weiterhin wurde bei der Suche nach Praktikumsangeboten bisher nur die Kriterien Ort und Schlagwort zur Auswahl gestellt. Hier wäre es durchaus denkbar noch weitere Kriterien in die Suche mit einzubeziehen, denn das wurde bereits in dem Datenbankentwurf bedacht. Somit wären weitere mögliche Suchkriterien das Land, die Postleitzahl, der Studiengang oder eventuell der Ansprechpartner an der Hochschule. Weiterhin könnte die Suche optimiert werden, indem mehrere Schlagworte auswählbar gemacht werden. Ein weiterer Gedanke, der beim Thema Schlagworte aufkommt, ist das Einfügen der Schlagworte zu optimieren. Im Moment können zu jedem Praktikumsangebot bis zu drei neue Schlagworte hinzugefügt werden. Damit wird die Auswahlliste in der Suche jedoch länger und unübersichtlicher, insbesondere wenn die Praktikumsbörse um weitere Studiengänge erweitert wird. Ein Vorschlag wäre eine bestimmte Zahl an Schlagworten in die Datenbank einzufügen und in dem Datenerfassungsformular nur die Zuordnung zu Schlagworten zu übernehmen. Des Weiteren könnte die Zuordnung der Praktikanten zum Praktikum verbessert werden. Im Prototypen muss dies noch über das Eintragen der PraktikumsID vollzogen werden. In dem Formular zur Erfassung der Praktikumsangebote wird die Adresse eingetragen, über die die Längen- und Breitenangaben des Praktikumsortes durch einen Google Dienst ermittelt werden. Dieser Dienst gibt nicht nur die Länge und die Breite zurück, sondern wie in Kapitel 5.2.5. beschrieben auch einen Statuscode und die Genauigkeit. Diese können sehr nützlich sein und zum Beispiel für Fehlermeldungen verwendet werden, um dem Nutzer mitzuteilen, aus welchem Grund die Geocodierung nicht durchgeführt werden konnte.

Eine weitere Überlegung wäre, die Markierungen auf der Karte über einen Link mit den Ergebnissen in den Tabellen zu verknüpfen. Damit würde es für den Nutzer noch einfacher sein die Symbole auf der Karte den Informationen in den Tabellen zu zuordnen. Weiterhin könnten die Symbole der Markierungen nummeriert werden. Dazu müssten jedoch eigene Symbole erzeugt werden, denn jeder Marker ist eine kleine Grafik, die über die Karte gelegt wird. Bei einer großen Anzahl an Markern wird es schnell unübersichtlich und verschlechtert zudem sicher die Performance der gesamten Anwendung. Deshalb könnte darüber nachgedacht werden, ab einer bestimmten Anzahl von Ergebnissen, einige von ihnen zusammenzufassen (clustering).

# Literaturverzeichnis

- [1] Robert Chromow: Google Maps ganz einfach in die eigene Website einbinden: Der Google-Baukasten für Landkarten, Anfahrtsrouten und Stadtpläne, <http://www.akademie.de/programmierung-administration/perl/tipps/skripte-formulare/google-baukasten-fuer-karten-und-plaene.html> (eingestellt am 11.09.2007, abgerufen am 07.09.2008)
- [2] PHP BLOGGER: Suchen (und finden) mit PHP – Teil 3: Die Basics der Schlagwortsuche, <http://www.phpblogger.net/2007/07/14/suchen-und-finden-mit-php-teil-3-die-basics-der-schlagwortsuche/> (eingestellt am 14.07.2007, abgerufen am 09.09.2008)
- [3] T3COM: Glossar: CMS, <http://t3com.de/glossar.html> (abgerufen am 12.09.2008)
- [4] admin-wissen.de: Typo3 Workshop: Was ist ein CMS?: Was ist Typo3, <http://www.admin-wissen.de/eigene-tutorials/webentwicklung/typo3-workshop/das-typo3-backend/was-ist-ein-cms/was-ist-typo3/> (abgerufen am 12.09.2008)
- [5] kgblog: Google Maps Beispiele, <http://kgblog.de/googlemaps/> (abgerufen am 12.09.2008)
- [6] Hans-Christian Dirscherl : Ratgeber: Google-Maps in den eigenen Internetauftritt einbauen, [http://www.pcwelt.de/start/dsl\\_voip/online/praxis/177420/google\\_maps\\_in\\_den\\_eigenen\\_internetauftritt\\_einbauen/](http://www.pcwelt.de/start/dsl_voip/online/praxis/177420/google_maps_in_den_eigenen_internetauftritt_einbauen/) (abgerufen am 13.09.2008)
- [7] Google: Google Maps API Services: Geocoding, <http://code.google.com/apis/maps/documentation/services.html#Geocoding> (abgerufen am 29.09.2008)
- [8] Purvis, Michael / Sambells, Jeffrey / Turner, Cameron (2007): Google Maps Anwendungen mit PHP und Ajax, Mitp-Verlag, Auflage: 1 (Juli 2007)
- [9] Tino Hempel: Normalisierung von Datenbanken, <http://www.tinohempel.de/info/info/datenbank/normalisierung.htm> (abgerufen am 14.09.2008)
- [10] MySQL 5.1 Referenzhandbuch: 1.9.5.5. Fremdschlüssel, <http://dev.mysql.com/doc/refman/5.1/de/ansi-diff-foreign-keys.html> (abgerufen 25.09.2008)

[11] Google Maps Hilfe: Entspricht meine Website den Nutzungsbedingungen für das Google Maps-API?, <http://maps.google.com/support/bin/answer.py?hl=de&answer=29435> (abgerufen am 28.09.2008)

[12] GNU Project: Die Definition Freier Software, <http://www.gnu.org/philosophy/free-sw.de.html>, (abgerufen am 03.10.2008)

## Abbildungsverzeichnis

Abbildung 1: <a href="http://www.uni-rostock.de/transfer/">http://www.uni-rostock.de/transfer/</a> (08.09.2008).....	7
Abbildung 2: <a href="http://praktikum.wifo.uni-mannheim.de/">http://praktikum.wifo.uni-mannheim.de/</a> (08.09.2008).....	8
Abbildung 3: <a href="http://www.praktikant24.de/praktikum/praktikumsboerse/angebote/studenten/">http://www.praktikant24.de/praktikum/praktikumsboerse/angebote/studenten/</a> (08.09.2008).....	9
Abbildung 4: <a href="http://www.ingenieurkarriere.de/bewerberservice/unternehmenskontakte/praktikumsboerse">http://www.ingenieurkarriere.de/bewerberservice/unternehmenskontakte/praktikumsboerse</a> (08.09.2008).....	10
Abbildung 5: <a href="http://www.hs-nb.de/geoinf/lehre/praxis/betriebe/index.html">http://www.hs-nb.de/geoinf/lehre/praxis/betriebe/index.html</a> (09.09.2008)....	11
Abbildung 6: <a href="http://www.booking.com">www.booking.com</a> (11.09.2008).....	18
Abbildung 7: Use-Case-Diagramm.....	20
Abbildung 8: Entity-Relationship-Model.....	26
Abbildung 9: 1:n Beziehung.....	27
Abbildung 10: Auszug einiger PHP-Seiten der Praktikumsbörse.....	31
Abbildung 11: Index-Seite der Praktikumsbörse .....	32
Abbildung 12: Anmeldung Praktikumsbörse.....	33
Abbildung 13: Sequenzdiagramm Anmeldung.....	34
Abbildung 14: Suche .....	35
Abbildung 15: Anmeldung für einen API-Schlüssel ( <a href="http://code.google.com/apis/maps/signup.html">http://code.google.com/apis/maps/signup.html</a> ).....	37
Abbildung 16: Google Startbeispiel.....	38
Abbildung 17: Info Window Route berechnen.....	40
Abbildung 18: Info Window Richtung.....	40
Abbildung 19: Info Window Get Directions.....	41
Abbildung 20: Ergebnis.....	42
Abbildung 21: Sequenzdiagramm Geocoding.....	46
Abbildung 22: erfolgreiche Anmeldung.....	48
Abbildung 23: Falsche Eingabe.....	49
Abbildung 24: In der falschen Anmeldemaske angemeldet.....	49
Abbildung 25: Praktikumsbörse mit Internet Explorer 7.....	50



## Tabellenverzeichnis

Tabelle 1: Pro und Contra Volltextsuche.....	12
Tabelle 2: Pro und Contra Schlagwortsuche.....	13
Tabelle 3: unnormalisierte Relation.....	24
Tabelle 4: 1. Normalform.....	24
Tabelle 5: HTTP-Statuscode [8, S. 84].....	45
Tabelle 6: Genauigkeitskonstanten [7].....	46

# Anhang

## A: Startbeispiel zur Google Maps API

```
1 <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
2   "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
3 <html xmlns="http://www.w3.org/1999/xhtml">
4 <head>
5   <meta http-equiv="content-type"
6     content="text/html; charset=utf-8"/>
7   <title>Google Maps API: JavaScript-Beispiel</title>
8   <script
9     src="http://maps.google.com/maps?file=api&
10     v=2&
11     key=ABQIAAAAIvGYk25Yn18xe9NKkAOfkBQA4D2A0d0z5H4PD0vravQLN_adGRRkArBFaU8t-W0iB83H9-PQvxCNvw"
12     type="text/javascript"></script>
13   <script type="text/javascript">
14     //
15     function load() {
16       if (GBrowserIsCompatible()) {
17         var map = new GMap2(
18           document.getElementById("map"));
19         map.setCenter(new GLatLng(37.4419,
20           -122.1419), 13);
21       }
22     }
23     //]]&gt;
24   &lt;/script&gt;
25 &lt;/head&gt;
26 &lt;body onload="load()" onunload="GUnload()"&gt;
27   &lt;div id="map" style="width: 500px;
28     height: 300px"&gt;&lt;/div&gt;
29 &lt;/body&gt;
30 &lt;/html&gt;</pre></div><div data-bbox="90 911 686 931" data-label="Page-Footer"><p>Anja Diekhoff, Entwurf und Implementation einer Praktikumsbörse</p></div><div data-bbox="878 911 915 930" data-label="Page-Footer"><p>58</p></div>
```

## B: index\_api.php

```
1 <?php
2 require_once($_SERVER['DOCUMENT_ROOT']. '\boerse\GoogleMapAPI.class.php');
3
4 $map = new GoogleMapAPI('map');
5 $map->setAPIKey($api_key);
6
7 include ('google/map_data.php');
8 //$map->disableDirections();
9
10 ?>
11
12 <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
13 <html xmlns="http://www.w3.org/1999/xhtml" xmlns:v="urn:schemas-microsoft-com:vml">
14 <head>
15 <?php $map->printHeaderJS(); ?>
16 <?php $map->printMapJS(); ?>
17 <!-- necessary for google maps polyline drawing in IE -->
18 <style type="text/css">
19     v\:* {
20         behavior:url(#default#VML);
21     }
22 </style>
23 </head>
24 <body onload="onLoad()">
25 <table border=1>
26 <tr><td>
27 <?php $map->printMap(); ?>
28 </td><td>
29 <?php $map->printSidebar(); ?>
30 </td></tr>
31 </table>
32 </body>
33 </html>
34
```